

INDEX

A

A record, DNS, 28–30
 Accuracy *versus* ambiguity, 9
 Alcatraz, isolated execution with, 131
 Analyzing forensic data
 See also capturing forensic data
 See also malware analysis
 See also timeline reconstruction
 See also virtual memory analysis
 from existing files, 70–73
 honeypots, 82–84
 identifying unusual activity, 4–5, 32
 OOV (order of volatility), 5–8, 20
 preparing for, 60–61
 process creation rate, 14
 replaying an incident, 120
 Anonymous memory, 163
 Anonymous memory pages, 165
 Archaeology *versus* geology, 13–15
 Architecture of computer systems, 88–89
 Argus system, 21–25
 Articles. *See* books and publications.
 atime attribute
 description, 18–20
 disabling update, 20–21
 example, 150
 Autonomous processes *versus* user control,
 13–15
 Avoiding intrusions. *See* evading intru-
 sions.

B

Backing up suspect files, 20, 61
 Barney intrusion
 DNS, and time, 28–31
 first signs, 17–18
 timeline reconstruction, 23–25, 28–31

Bind (Berkeley Internet Name Daemon),
 28–31
 Birth time, 50
 Bitmaps, file system, 54, 76, 147, 157
 “Black-box” dynamic analysis, 117
 Block device files, 47
 bmap command, 57
 Books and publications
 The Cuckoo’s Egg, 83
 “An Evening With Berferd,” 83
 Software Tools, 29
 Brute-force persistence, 149–151
 Buffer memory, 163
 Bypassing the file system, 55–56

C

Capturing forensic data
 See also analyzing forensic data
 accuracy *versus* ambiguity, 9
 archaeology *versus* geology, 13–15
 file system information, 61–63
 gaps in process IDs, 14
 honeypots, 82–84
 layers and illusions, 8–9
 perceptions of data, 9
 recovering encrypted file contents,
 172–173
 timelines. *See* timeline reconstruction.
 traps set by intruders, 10–11
 trustworthiness of information, 10–11
 user control *versus* autonomous
 processes, 13–15
 virtual memory, 165–171
 Case studies and examples
 atime attribute example, 150
 Barney intrusion
 DNS, and time, 28–31

Case studies and examples (*continued*)

- Barney intrusion (*continued*)
 - first signs, 17–18
 - timeline reconstruction, 23–25, 28–31
- ctime attribute example, 150
- malware sample code, 140
- mtime attribute example, 150
- persistence of deleted information
 - example, 146–147
- rpc.statd service
 - analyzing existing files, 70–73
 - capturing file system information, 61–63
 - copying entire disk, 61
 - copying files by inode number, 75
 - copying individual files and partitions, 61
 - creating disk images, 63–65
 - data blocks, 76
 - deleted file analysis, 77–78
 - disk images on analysis machine, 65–67
 - disk imaging over a network, 63–65
 - distributed denial-of-service attacks, 82–84
 - file deletion, effects of, 73–76
 - files, out of place, 78–79
 - files, tracing by inode, 81–82
 - files, tracing by location, 80–81
 - first contact, 59–60
 - honeypots, 82–84
 - inode blocks, 75–76
 - listing directory entries, 75
 - listing files by inode number, 76
 - MACtime, 68–69, 76–77
 - making backups, 61
 - parent directory attributes, 75
 - parent directory entries, 75
 - preparing file system for analysis, 60–61
 - timeline reconstruction, 68–69
 - system start-up, 90–92
 - UNIX file access example, 51–52
- ChangeTime, 18

Character device files, 47

Cheswick, Bill, 83

chroot() system call, 122–123

Clock skews, 23

Clocks. *See* timeline reconstruction.

Command-level rootkits, 102

Computer system architecture, 88–89

Concealing file system information, 42

Containers. *See* jails.Containing malware. *See* program confinement.

Copying

disk images, 63, 66

entire disk, 61

files by inode number, 75

individual files, 61

individual partitions, 61

Coroner's Toolkit. *See also* tools and utilities.

categorizing data content, 27

copying files by inode, 75

dynamic state, examining, 65

file length, examining, 77–78

file recovery, 49

file system images, examining, 68–69

grave-robber command, 65, 68–69

icat command

copying files by inode, 75

examining file length, 77–78

reading inode data block references, 51

recovering deleted files, 49

saving journal contents, 33

icat utility, 75

ils command, 49, 51, 77–78

ils utility, 76

lazarus tool, 27

listing files by inode, 76

mactime tool, 19, 68–69. *See also* timeline reconstruction.

pcat command, 47

probing memory locations, 47

processing deleted file information, 77–78

reading inode contents, 51

Coroner's Toolkit (*continued*)
 saving journal contents, 33
 sorting by time of day, 68–69
 ctime attribute
 changing, 21
 description, 18–20
 example, 150
The Cuckoo's Egg, 83

D

Data

analyzing. *See* analyzing forensic data.
 capturing. *See* capturing forensic data.
 categorizing by content. *See* lazarus program.
 frequency of change, 4–5
 life expectancy, 6. *See also* persistence.

Data block addresses, 50

Data blocks, effects of file deletion, 76

Date and time. *See* timeline reconstruction.

dd command, 53, 58, 62–63, 65, 67, 81, 168–169, 195

debugfs command, 33–34

Decompiling programs. *See* reverse engineering.

Deleted file analysis, 77–78

Deleted file attributes, half-life, 153

Deleted file contents, half-life, 148

Deleted file persistence, 12. *See also* undeleting files.

Deleted files, fossilization of information, 12

Deletion time, 50

Demand paging, 163

Destructive software. *See* malware; rootkits.

Detecting intrusions. *See also* analyzing forensic data; capturing forensic data.

kernel-level rootkits, 111–115

malware, 152

rootkits, 102–106

Device files, 47

/dev/kmem device file, 168–169

/dev/mem device file, 168–169

Directories

accessing over networks, 52–53

description, 45–46

names, listing, 51

Disassembling programs. *See* reverse engineering.

Disk images

on analysis machine, 65–67

copying, 63, 66

creating, 63–65

sending over a network, 63–65

Disk label, 54

Disk partitions. *See* partitions.

Disks

displaying partitions, 43

wasted space, 57

Distributed denial-of-service attacks, 82–84

dmesg, 42–43

DNS (Domain Name Service)

A record, 28–30

MX record, 28

PTR record, 28–31

TTL (Time to Live), 28–29, 31

DNS and time, 28–31

Doubly indirect blocks, 50

Downstream liability, 84

Drift, time, 34–35

dtime attribute, 18

Dumping memory, 167–171

Dumping swap space, 169

Dynamic analysis. *See also* analyzing forensic data.

“black-box,” 117

dangers of, 118

definition, 117

with library-call monitors, 132–133

machine-instruction level, 136

with system-call monitors, 123–126

virtual memory, 177–179

E

Encrypted file contents, recovering, 172–173

Encryption, 56

210 Index

Evading intrusions
 kernel-level rootkits, 111–115
 rootkits, 102–106
 “An Evening with Berferd,” 83
 Evidence of intrusion. *See* analyzing forensic data; capturing forensic data.
 Examining forensic data. *See* analyzing forensic data; capturing forensic data.
 Examples. *See* case studies and examples.
 Executable program layer, 88–89
 Ext3fs, 32–34, 40

F

False evidence, 84
 fdisk command, 43
 FIFO processes, 46–47
 File system blocks, 173–175
 File systems. *See also* UNIX file system.
 Ext3fs, 32–34, 40
 journaling, 31–34
 layout, 54
 restricting access to, 122–123. *See also* program confinement.
 Steganographic File System, 56
 File types, 45–47, 49
 Files
 access, examples, 51–52
 activity, frequency of use, 4–5
 deletion, effects of, 73–76
 headers, timeline reconstruction, 26–27
 names
 description, 44
 listing, 51
 open, listing, 97–98
 organization, 40–43
 out of place, 78–79
 as pseudo-disk partitions, 66–67
 recognizing from memory, 175–177
 recovering encrypted contents, 172–173
 size, 49
 tracing by inode, 81–82
 tracing by location, 80–81

First contact, 59–60
 First signs of trouble, 17–18
 fls command, 51, 75
 Forensic data. *See* data.
 analyzing. *See* analyzing forensic data.
 capturing. *See* capturing forensic data.
 protecting. *See* protecting forensic data.
 Forging MACTimes, 21
 Fossilization of information, 12, 84
 Fragmentation, 57
 Frequency of data change, 4–5
 fuser command, 42

G

Gaps in process IDs, 14
 Garner, George, 169
 gdb command, 133
 Geology *versus* archaeology, 13–15
 grave-robber command, 65, 68–69
 Gutmann, Peter, 146

H

Half-life
 deleted file contents, 148
 deleted file attributes, 153
 Hard link counts, 49
 Hard virtual machines, 119
 Hardware access, 47
 Hardware layer, 88–89
 Heisenberg, Werner, 7
 Heisenberg uncertainty principle, 7
 Hobbit’s Netcat. *See* Netcat.
 Holes (in files), 53, 61
 HoneyNet Project, 83–84
 HoneyPots, 82–84
 Hypervisor. *See* Virtual machine monitor

I

icat command
 copying files by inode, 75
 examining file length, 77–78

`icat` command (*continued*)
 reading inode data block references,
 51
 recovering deleted files, 49
 saving journal contents, 33

Illusions about data, 8–9

`ils` command
 examining file length, 77–78
 listing files by inode, 76
 recovering deleted files, 49, 51

Information leaks, 84

inode blocks, 75–76

inode information, 50–51

Installation, kernel-level rootkits,
 107–108

Interactive mode, 129

Intrusions

See analyzing forensic data

See capturing forensic data

See detecting intrusions

See evading intrusions

IPC endpoints, 46–47

J

Jails, 122–123

Janus system, 127–128

Journaling file systems, 31–34

K

Kernel

 configuration mechanisms, 92–94

 loadable kernel modules, 93–94

Kernel memory, 163

Kernel security levels, 95–96

Kernel-level rootkits

 detection and evasion, 111–115

 installation, 107–108

 operation, 108–111

 subversion with, 107

`klogd` daemon, 42–43

Known Goods database, 70

L

Last access, determining, 18–20

Last access (read) time, 50

Last change to contents, determining, 18–20

Last change to meta-information, deter-
 mining, 18–20

Last modification time, 50

Last status change, 50

`LastAccessTime`, 18

`LastWriteTime`, 18

Layers of data, 8–9

`lazarus` program, 27

Library calls

 dangers of, 133–135

 description, 133–135

Library layer, 88–89

Library of Alexandria, 21–22

Library-call monitoring, 132–133

Library-level rootkits, 106–107

Life cycle of UNIX systems, 89–90

Life expectancy of data, 6

Listing

 active ports, 98–99

 directory entries, 75

 file and directory names, 51

 files by inode number, 76

 processes, 97–99

Loadable kernel modules, 93–94

`lofiadm` command, 67

Log files, timeline reconstruction, 26–27

Long-term persistence, 153–154

`lookup()` operation, 110

Loopback file system mount, 41, 67–68

`ls` command, timeline reconstruction, 19–20

`lsOf` command, 42

`lstat()` system call, 19–21, 50, 128

`ltrace` command, 132–133

M

Machine-instruction level, dynamic analy-
 sis, 136

`mactime` tool, 19, 68–69

212 Index

- MACtimes**
degrading over time, 21
deleted files, 76–77
forging, 21
historical activity, 21
introduction, 18–20
journaling file systems, 31–34
limitations, 20–21
malware detection, 152
persistence of deleted information
 brute-force persistence, 149–151
 impact of user activity, 154–156
 long-term persistence, 153–154
 malware analysis, 152
 measuring, 149
 mechanism of persistence, 157–159
 trustworthiness of deleted information, 156–157
 sample report, 68–69
- Magritte, René**, 8
- Malware analysis.** *See also* analyzing forensic data; program confinement; rootkits.
countermeasures, 141
detection, with MACtimes, 152
dynamic analysis
 “black-box,” 117
 dangers of, 118
 definition, 117
 with library-call monitors, 132–133
 machine-instruction level, 136
 with system-call monitors, 123–126
MACtimes, 152
sample code, 140
static analysis
 definition, 117
 reverse engineering, 136–140
- MD5 hash**, 65, 70, 78, 103, 106–107, 174–176, 178–180, 183, 188
- Measuring persistence of deleted information**, 149
- memdump** program, 169
- Memory.** *See* virtual memory.
- Memory device files**, 168–169
- Memory manager**, 161–164
- Memory page technique**, 173–175
- Memory pages**
anonymous, 165
definition, 162
description, 164
and files, 164–165
- Monitoring**
networks, 22–25
process status, 96–97
system calls, 124–126
- mtime** attribute
description, 18–20
example, 150
MX record, DNS, 28
- N**
- Named pipes**, 46–47
- Netcat**, 63, 66
- Network monitoring**, 22–25
- newfs** command, 57
- nm** command, 134
- NTP (Network Time Protocol)**, 35
- O**
- objdump** command, 134
- OOV (order of volatility)**, 5–8, 20
- Ownership**, 48
- opendir()** library function, 46
- OpenSSH.** *See* ssh
- P**
- Pages.** *See* memory pages.
- Papers.** *See* books and publications.
- Parent directory attributes**, 75
- Parent directory entries**, 75
- Partitions**
copying, 61
displaying, 43
pseudo, files as, 66–67
- Pathnames**, 44–45
- pcat** command, 47
- Perceptions of data**, 9

- Permissions, 49
 - Persistence, virtual memory
 - anonymous data, 180–182
 - data, 177–179
 - files, 179–180
 - nonfile data, 180–182
 - swap space, 182
 - through the boot process, 182
 - trustworthiness, 182–185
 - Persistence of deleted information
 - deleted files, 12
 - examples, 146–147
 - MACTimes
 - brute-force persistence, 149–151
 - impact of user activity, 154–156
 - long-term persistence, 153–154
 - malware analysis, 152
 - measuring, 149
 - mechanism of persistence, 157–159
 - trustworthiness of deleted information, 156–157
 - measuring
 - file contents, 147–148
 - file MACTimes, 149–151
 - Policy-enforcing mode, 128–129
 - Policy-generating mode, 128–129
 - Ports, listing active, 98–99
 - POSIX, 44, 48–49, 112, 114
 - Post-mortem analysis, 22, 58–60, 68, 83, 85, 117–118, 160
 - Preserving forensic data. *See* protecting forensic data.
 - Preventing intrusions. *See* evading intrusions.
 - Process creation rate, 14
 - Process IDs, identifying gaps, 14
 - Process memory, 163
 - Processes
 - active ports, listing, 98–99
 - busy, identifying, 42
 - interprocess communication, 46–47
 - listing, 97–99
 - open files, listing, 97–98
 - restricting actions of, 126–129
 - status monitoring, 96–97
 - Program confinement. *See also* malware analysis.
 - chroot () system call, 122–123
 - hard virtual machines, 119
 - jails, 122–123
 - Janus system, 127–128
 - library calls, 133–135
 - ReVirt system, 120
 - sandboxes, 118
 - soft virtual machines, 119–122
 - system-call censors, 126–129
 - system-call monitors, 129–132
 - system-call spoofing, 129–131
 - Systrace policy, 128–129
 - Program disassembly / decompilation. *See* reverse engineering.
 - Protecting forensic data
 - backing up suspect files, 20, 61
 - clock skews, 23
 - disabling atime update, 20–21
 - journaling file systems, 31–34
 - kernel security levels, 95–96
 - last access, 18–20
 - last change to contents, 18–20
 - last change to meta-information, 18–20
 - OOV (order of volatility), 5–8, 20
 - time drift, 34–35
 - traps set by intruders, 10–11
 - trustworthiness of information, 10–11
 - Protection from intrusion. *See* evading intrusions.
 - Ptolemy III, 21–22
 - PTR record, DNS, 28–31
 - Publications. *See* books and publications.
- ## R
- RAM (random-access memory). *See* virtual memory.
 - Regular files
 - accessing as block devices, 67
 - definition, 45
 - Replaying an incident, 120
 - Resident operating system kernel, 88–89, 92–94

214 Index

Resident set size, virtual memory, 163
 Restoring files. *See* undeleting files.
 Restricting file system access, 122–123
 Reverse engineering, 136–140
 ReVirt system, 120
 Ring buffer, kernel message, 42
 Roach motels, 83
 Rootkits. *See also* malware; tools and utilities.
 activity signatures, 151
 command level, 102
 description, 101
 evasion and detection, 102–106
 kernel level
 evasion and detection, 111–115
 installation, 107–108
 operation, 108–111
 subversion with, 107
 library level, 106–107
 rpc.statd service case study. *See* case studies and examples, rpc.statd service.
 Run levels, 90

S

Sandboxes, 118
 savecore command, 165, 167–169, 183
 Scripting languages, timeline reconstruction, 29–31
 Secure booting, 92
 securelevel feature, 56, 93, 95
 Security, kernel security levels, 95–96
 Segal's Law, 34
 setuid() system call, 112, 133–134, 139–140
 SHA-1 hash, 65, 70, 103, 105
 Singly indirect blocks, 50
 Slash (/), in UNIX file system, 44
 Sleuth Kit, 51, 75, 80
 Soft virtual machines, 119–122
 Software, malicious. *See* malware; rootkits.
 Software disassembly/decompilation. *See* reverse engineering.

Software Tools, 29
 Solaris fingerprint database, 70
 sotruss command, 132–133
 Sparse files, 53
 ssh (secure shell)
 command, 64–65, 169
 server, 17, 20, 22, 23, 30, 97, 102, 105, 125–126
 tunnel, 65
 stat() system call, 50, 106, 112, 128
 Start-up, case study, 90–92
 Static analysis. *See also* analyzing forensic data.
 definition, 117
 reverse engineering, 136–140
 virtual memory, 171
 Steganographic File System, 56
 Stoll, Clifford, 83
 strace command, 124–126
 strings command, 27, 52, 70–71, 75, 103–104, 111, 117, 134, 137, 171, 173
 Studying program behavior. *See* analyzing forensic data.
 Superblock, 54
 Swap devices, 168–169
 Swap space
 determining, 165–167
 dumping, 169
 persistence, 182
 Symbolic links, 46
 Synchronization, timeline reconstruction, 34–35
 sysctl command, 93, 95, 166
 syslogd daemon, 42–43
 System start-up, case study, 90–92
 System-call censors, 126–129
 System-call monitors
 dangers of, 131–132
 description, 129–131
 dynamic analysis, 123–126
 System-call spoofing, 129–131
 Systrace policy, 128–129

T

- Time attributes, 18–20. *See also* timeline reconstruction.
- Time stamps, 50
- Timeline reconstruction. *See also* analyzing forensic data; MACtime.
 - accuracy, 34–35
 - Argus system, 21–25
 - atime attribute
 - description, 18–20
 - disabling update, 20–21
 - example, 150
 - backing up suspect files, 20
 - ChangeTime, 18
 - clock skews, 23
 - ctime attribute
 - changing, 21
 - description, 18–20
 - example, 150
 - debugfs command, 33–34
 - DNS and time, 28–31
 - drift, 34–35
 - dtime attribute, 18
 - examining parts of systems, 27
 - Ext3fs file system, 32–34
 - file headers, 26–27
 - first signs of trouble, 17–18
 - icat command, 33
 - journaling file systems, 31–34
 - last access, 18–20
 - last change to contents, 18–20
 - last change to meta-information, 18–20
 - LastAccessTime, 18
 - LastWriteTime, 18
 - lazarus program, 27
 - log files, 26–27
 - mtime attribute
 - description, 18–20
 - example, 150
 - network monitoring, 22–25
 - rpc.statd case study, 68–69
 - scripting languages, 29–31
 - Segal's Law, 34
 - synchronization, 34–35
 - time attributes, 18–20
 - time data in unusual places, 25–27
 - uncertainty, 34–35
- Tools and utilities. *See also* Coroner's Toolkit; rootkits.
 - capturing memory, 165–171
 - copying disk images, 63, 66
 - dumping memory, 169
 - Findrootkit, 114
 - library-call monitoring, 132–133
 - listing open files, 97–98
 - listing open ports, 98
 - lofiadm command, 67
 - ltrace command, 132–133
 - memdump program, 169
 - memory, determining, 165–167
 - monitoring system calls, 124–126
 - mounting file system images, 67
 - Netcat, 63, 66
 - process and system status
 - functional description, 99–100
 - limitations, 100–101
 - lsdf (list open files) command, 97–99
 - /proc pseudo-file system, 99–100
 - ps command, 96
 - process status monitoring, 96, 99–100
 - regular files, accessing as block devices, 67
 - replaying incidents, 120
 - ReVirt system, 120
 - rootkit detection, 114
 - savecore command, 167–171
 - Sleuth Kit, 51, 75, 80
 - sotruss command, 132–133
 - strace command, 124–126
 - swap space, determining, 165–167
 - top command, 165–167
 - truss command, 124–126
 - vnode pseudo-disk devices, 67
- top command, 165–167
- Tracking intruders. *See* analyzing forensic data; capturing forensic data.

216 Index

- Traps set by intruders, 10–11
- `truss` command, 124–126
- Trustworthiness
 - of deleted information, 156–157
 - limitations of, 10–11
 - virtual memory persistence, 182–185
- TTL (Time to Live) of DNS records, 28–29, 31
- Turing, Alan, 10

- U**
- Uncertainty
 - Heisenberg uncertainty principle, 7
 - perceptions of data, 9
 - time, 34–35
- Undeleting files, 3. *See also* deleted file persistence.
 - `icat` command, 49
 - `ils` command, 51
 - recovering encrypted file contents, 172–173
- UNIX file system analysis
 - analyzing existing files, 70–73
 - capturing file system information, 61–63
 - copying
 - disk images, 63, 66
 - entire disk, 61
 - files by inode number, 75
 - individual files and partitions, 61
 - creating disk images, 63–65
 - data blocks, 76
 - deleted file analysis, 77–78
 - disk images on analysis machine, 65–67
 - disk imaging over a network, 63–65
 - distributed denial-of-service attacks, 82–84
 - files
 - deletion, effects of, 73–76
 - out of place, 78–79
 - tracing by inode, 81–82
 - tracing by location, 80–81
 - first contact, 59–60
 - honeypots, 82–84
 - inode blocks, 75–76
 - listing directory entries, 75
 - listing files by inode number, 76
 - MACtime, 68–69, 76–77
 - making backups, 61
 - parent directory attributes, 75
 - parent directory entries, 75
 - preparing file system for analysis, 60–61
 - timeline reconstruction, 68–69
- UNIX file system basics
 - aliases. *See* symbolic links.
 - below the file system interface, 56–57
 - birth time, 50
 - block device files, 47
 - bypassing the file system, 55–56
 - character device files, 47
 - concealing information, 42
 - data block addresses, 50
 - deletion time, 50
 - device files, 47
 - directories, 45–46
 - directory access over networks, 52–53
 - disk label, 54
 - displaying disk partitions, 43
 - doubly indirect blocks, 50
 - encryption, 56
 - FIFO processes, 46–47
 - file access, examples, 51–52
 - file names, 44
 - file organization, 40–43
 - file size, 49
 - file system layout, 54
 - file types, 45–47, 49
 - fragmentation, 57
 - hard link counts, 49
 - hardware access, 47
 - identifying busy processes, 42
 - inode information, 50–51
 - internals, 48–53
 - interprocess communication, 46–47
 - IPC endpoints, 46–47
 - last access (read) time, 50
 - last modification time, 50
 - last status change, 50
 - listing file and directory names, 51
 - named pipes, 46–47

UNIX file system basics (*continued*)

- ownership, 48
 - pathnames, 44–45
 - permissions, 49
 - regular files, 45
 - shortcuts. *See* symbolic links.
 - singly indirect blocks, 50
 - sparse files, 53
 - superblock, 54
 - symbolic links, 46
 - time stamps, 50
 - unmounting, 42
 - wasted space, 57
 - zones, 54, 81, 157–159
- UNIX systems, life cycle, 89–90
- Unmounting file systems, 42
- Unusual activity, identifying, 4–5
- User activity, impact on persistence, 154–156
- User control *versus* autonomous processes, 13–15

V

- Virtual machine, monitor, 116, 119–121
- Virtual machines
- hard, 119
 - soft
 - dangers of, 121–122
 - definition, 119–120
 - VMware virtual hardware environment, 121–122
- Virtual memory analysis
- anonymous memory, 163
 - buffer memory, 163
 - cap, 165
 - capturing, 165–171
 - classes of, 163
 - `/dev/kmem` device file, 168–169

- `/dev/mem` device file, 168–169
 - dumping, 167–171
 - dynamic analysis, 177–179
 - file recognition, 175–177
 - file system blocks, 173–175
 - kernel memory, 163
 - memory device files, 168–169
 - memory page technique, 173–175
 - overview, 162–163
 - pages
 - anonymous, 165
 - definition, 162
 - description, 164
 - and files, 164–165
 - persistence
 - anonymous data, 180–182
 - data, 177–179
 - files, 179–180
 - nonfile data, 180–182
 - swap space, 182
 - through the boot process, 182
 - trustworthiness, 182–185
 - process memory, 163
 - recognizing from files, 171
 - recovering encrypted file contents, 172–173
 - static analysis, 171
 - swap devices, 168–169
 - swap space, 169
- VMware virtual hardware environment, 121–122
- vnode pseudo-disk devices, 67

Z

- Zero, Darryl, 3
- The Zero Effect*, 3
- Zones. *See* jails.
- Zones, file system, 54, 81, 157–159

