

INDEX

Note: Pages numbers followed by f, n, or t indicate figures, notes, and tables, respectively.

Symbols

(hash mark), 13, 15

! (logical NOT) operator, 42

A

Acar, Can Erkin, 173

ACK (acknowledgment) packets
class-based bandwidth allocation,
139–140
HFSC algorithm, 124, 126, 142
priority queues, 132, 137–138
two-priority configuration,
120–121, 120n1

adaptive.end value, 188

adaptive firewalls, 97–99

adaptive.start value, 188

advbase parameter, 153–154

advskew parameter, 153–154, 158–159

aggressive value, 192

ALTQ (alternate queuing) framework,
9, 133–145, 133n2

basic concepts, 134

class-based bandwidth allocation,
139–140

overview, 135

queue definition, 139–140

tying queues into rule set, 140

handling unwanted traffic, 144–145

operating system-based queue
assignments, 145

overloading to tiny queues,
144–145

HFSC algorithm, 140–142

overview, 135

queue definition, 140–141

tying queues into rule set,
141–142

priority-based queues, 136–145

match rule for queue assignment,
137–138

overview, 134–135

performance improvement,
136–137

queuing for servers in DMZ,
142–144

setting up, 135–136

on FreeBSD, 135–136

on NetBSD, 136

on OpenBSD, 135

transitioning to priority and
queuing system, 131–133

anchors, 35–36

authpf program, 61, 63

listing current contents of, 92

loading rules into, 92

manipulating contents, 92

relayd daemon, 74

restructuring rule set with, 91–94

tagging to help policy routing, 93

ancontrol command, 46n1

antispoof tool, 27, 193–195, 194f

ARP balancing, 151, 157–158

atomic rule set load, 21

authpf program, 59–63, 60

basic authenticating gateways,
60–62

public networks, 62–63

B

bandwidth

actual available, 142–143

class-based allocation of, 139–140
overview, 135

queue definition, 139–140

tying queues into rule set, 140

queues for allocation of, 121–122

DMZ network with traffic
shaping, 128–130

fixed, 123–125

- bandwidth, queues for allocation of
 - (*continued*)
 - flexible, 125–128
 - HFSC algorithm, 123
 - total usable, 122
 - Beck, Bob, 115
 - Berkeley Software Distributions. *See*
 - BSDs (Berkeley Software Distributions); FreeBSD; NetBSD; OpenBSD
 - blacklisting, 101–103, 115
 - block all rule, 19, 24, 61, 69
 - block in all rule, 16–17
 - blocknonip option, 87–88
 - block-policy option, 186–187
 - block rule, 13
 - Brauer, Henning, 5, 133, 177
 - brconfig command, 87, 89
 - bridges, 86–91, 86n5, 90f
 - defined, 86
 - pros and cons of, 86
 - rule set, 90–91
 - setting up
 - on FreeBSD, 88–89
 - on NetBSD, 89–90
 - on OpenBSD, 87–88
 - brute-force attacks, 96–99
 - defined, 96
 - expiring tables using pfctl, 99
 - overview, 96
 - setting up adaptive firewalls, 97–99
 - BSDs (Berkeley Software Distributions), 3–4, 3n3.
 - See also* FreeBSD; NetBSD; OpenBSD
 - configuration files, 7
 - Linux versus, 6–7
 - network interface naming
 - conventions, 6
 - online resources, 201–203
 - print resources, 204–205
 - Bytes In/Out statistics, 23
- C**
- CARP (Common Address Redundancy Protocol), 79
 - failover, 150–154
 - kernel options, 150
 - network interface setup with
 - ifconfig, 151–154
 - sysctl values, 151
 - load balancing, 157
 - load-balancing mode, 158
 - setting up, 158–160
 - overview, 147–148
 - carpdev option, 150, 152
 - cbq (class-based) queues, 132–135
 - definition, 139–140
 - tying into rule set, 140
 - cloneable interfaces, 55n4, 167
 - command succeeded message, 77
 - Common Address Redundancy Protocol. *See* CARP
 - complicated networks, 65–94
 - bridges, 86–91
 - FreeBSD setup, 88–89
 - NetBSD setup, 89–90
 - OpenBSD setup, 87–88
 - rule set, 90–91
 - interface groups, 84–85
 - NAT, 79–84
 - DMZ, 80–81
 - load balancing with
 - redirection, 81
 - single NATed network, 81–84
 - nonroutable IPv4 addresses, 91–94
 - establishing global rules, 91
 - restructuring rule set with
 - anchors, 91–94
 - packet tagging, 85–86
 - routable IPv4 addresses, 66–79, 67f
 - DMZ, 70–71, 70f
 - load balancing with redirection,
 - 72–73
 - load balancing with relayd,
 - 73–79
 - macros, 66–67
 - configuration files
 - FreeBSD, 7, 14–15
 - NetBSD, 15–16
 - OpenBSD, 7, 13
 - tools for managing, 7–8, 11
 - connection refused message, 18
 - content filtering, 100, 105, 107
 - Core Force project, 5n7
 - Core Security, 5n7
- D**
- DDoS (distributed denial-of-service)
 - attacks, 187, 187n1

- debugging, 197–199. *See also* logging
 - debug option, 190–191
 - troubleshooting-friendly networks, 37–38
 - debug option, 52, 190–191
 - deep packet inspection, 2
 - demilitarized zone (DMZ). *See* DMZ
 - demotion counter, 79, 153
 - denial-of-service (DoS) attacks, 91, 168, 193n2
 - de Raadt, Theo, 4n4
 - dhclient command, 56–57, 59
 - dhcpcd program, 54
 - distributed denial-of-service (DDoS) attacks, 187, 187n1
 - divert(4) sockets, 2
 - divert-to component, 36
 - Dixon, Jason, 10
 - dmesg command, 48–49, 209
 - DMZ (demilitarized zone)
 - NAT, 80–81
 - queuing for servers in, 142–144
 - routable IPv4 addresses, 70–71, 70f
 - testing rule set, 195–196, 195f
 - with traffic shaping, 128–130, 128f
 - DNS, 22, 34n4, 66, 68
 - documentation, 8
 - domain name lookups, 163–164, 166, 169
 - domain name resolution, 18, 20
 - domain names, 34
 - DoS (denial-of-service) attacks, 91, 168, 193n2
 - DragonFly BSD, 3n3, 5–6, 12
 - dropped packets, 128
 - drop value, 186
- E**
- echo requests/replies, 38–41, 53, 69, 82, 90, 92
 - Engen, Vegard, 62n5
 - expiretable tool, 99n4
- F**
- failover, 148–156
 - CARP, 79, 150
 - kernel options, 150
 - network interface setup with ifconfig, 151–154
 - sysctl values, 151
 - load balancing versus, 158
 - pfsync protocol, 154–155
 - rule set, 155–156
 - false positives, 102, 106, 110, 115
 - FIFO (first in, first out), 120, 132–134, 137
 - file servers
 - NAT, 79
 - routable IPv4 addresses, 66–67
 - file transfer protocol. *See* FTP
 - firewalls, 3. *See also* bridges
 - adaptive, 97–99
 - simple gateways, 25–27
 - first in, first out (FIFO), 120, 132–134, 137
 - flags S/SA keep state rule, 21
 - floating state policy, 187
 - Floeter, Reyk, 183
 - flowd collector daemon, 177–182
 - flowd-reader program, 178–181
 - flow-tools program, 177
 - flush global state-tracking option, 97
 - fragment reassembly options, 192–193
 - frag value, 188
 - FreeBSD, 3n3, 5
 - configuration files, 7
 - online resources, 204
 - pfSense, 8
 - setting up ALTQ framework on, 135–136
 - setting up bridges, 88–89
 - setting up PF on, 13–15
 - spamd spam-deferral daemon, 101, 105
 - wireless interface configuration, 50
 - wireless network setup, 58–59
 - WPA access points, 52–53
 - FreeBSD Handbook, 14
 - from keyword, 33
 - FTP (file transfer protocol), 35–37, 53–54
 - fetching list data via, 102
 - ftp-proxy with diversion or redirection, 36–37
 - history of, 35, 35n5
 - security challenges, 35
 - variations on ftp-proxy setup, 37
 - ftp-proxy command, 13
 - enabling, 36
 - redirection, 36–37
 - reverse mode, 36–37

ftpproxy_flags variable, 36–37
FTPS, 35n6
fw_update script, 48

G

grep program, 113, 178
greyxp value, 107
greylisting, 104–108

- compensating for unusual situations, 113–114
- defined, 104
- keeping lists in sync, 112–113
- online resources, 205–206
- in practice, 107–108
- setting up, 104–105, 107

greytrapping, 109–111, 115

- adding to list, 111–112
- deleting from list, 112

H

Hail Mary Cloud sequence of brute-force attempts, 98, 98n2
hardware, 5, 207–210

- helping hardware support efforts, 210
- issues facing hardware support developers, 209
- pool memory, 190
- selecting, 208–209
- selecting for wireless networks, 48

Harris, Evan, 104
Hartmeier, Daniel, 4–5, 132, 136
hash mark (#), 13, 15
HFSC (Hierarchical Fair Service Curve) algorithm, 123, 125–126, 134–135, 140–142

- queue definition, 140–141
- transitioning from ALTQ to priority and queuing system, 132–133
- tying queues into rule set, 141–142

high-latency value, 192
hostapd command, 52–53
host command, 18, 22, 34
hostnames, 34
HTTP, 68, 75, 77–79, 99

- fetching list data via, 102
- NetFlow data collection, 181

HTTPS, 77, 79

I

IBM Christmas Tree EXEC worm, 2n1
ICMP, 37–41, 41n7, 124, 140

- bandwidth allocation, 124
- letting pass unconditionally, 38
- letting pass while stopping probes from elsewhere, 39
- path MTU discovery, 40–41

ICMP6, 38

- letting pass unconditionally, 38
- letting pass while stopping probes from elsewhere, 39
- path MTU discovery, 41

if-bound policy, 187–188
if_bridge module, 88
ifconfig command, 46n1, 59, 109, 148

- bridge setup, 87–89
- interface groups, 84–85
- logging, 167, 176
- MTU, 40
- redundancy and resource availability, 150–155, 158–160
- running status of interfaces, 30
- wireless networks, 49–53, 56–59

ifstated interface state daemon, 157
ILOVEYOU worm, 2n1
inserts statistic, 23
interface groups, 84–85
Interface Stats statistics, 23
interval value, 188
IP-based load balancing, 157–158
IPFilter subsystem, 4–5, 4n4, 4n5, 8–9
IPsec

- filtering on encapsulation interfaces, 55, 55n4
- state synchronization, 155
- with UDP key exchange, 55

IPv4, 23–24

- network address translation, 28–29, 54
- nonroutable addresses, 91–94
 - establishing global rules, 91
 - restructuring rule set with anchors, 91–94
- packet forwarding, 30
- routable addresses, 31–32, 66–79
 - DMZ, 70–71
 - load balancing with reLayd, 73–79

- load balancing with redirection, 72–73
 - wireless networks, 49–50, 54, 58
- IPv6, 24, 30, 37–38, 41, 67, 71, 73, 75, 81
- NAT versus, 28–29
- release of, 28
- wireless networks, 49–50, 54, 56–59

K

- KAME project, 28, 28n3
- keep state flags *S/SA* rule, 17n3
- keep state rules, 16–17, 17n3, 21, 26, 26n1, 41, 68, 188
- kernel memory, 189–190
- Knight, Joel, 183

L

- labels, 169–171
- leaf queues, 126–127
- limit option, 189
- linkshare value, 140–141
- Linux
 - BSD versus, 6–7
 - network interface naming conventions, 6
 - porting PF to Linux machines, 7
- lists
 - defined, 18
 - usefulness of, 20
- load balancing
 - CARP for, 157
 - load-balancing mode, 158
 - setting up, 158–160
 - redirection for
 - NAT, 81
 - routable IPv4 addresses, 72–73
 - with *relayd* daemon, 73–79
 - synproxy* state option, 68
- log (all) clause, 165–166
- logger option, 169
- logging, 161
 - all packets, 165–166
 - basic concepts, 162–164
 - graphing traffic with *pfstat*, 173–175
 - legal implications of, 166
 - monitoring with *pftop*, 173
 - monitoring with *sysstat*, 171–173

- NetFlow data collection, 176–182
 - flow collector daemon, 177–182
 - pfflowd* tool, 182
 - setting up sensor, 176–177
- packet path through rule set, 164–165
- to several *pflog* interfaces, 167
- SNMP tools and MIBs, 182–183
- to *syslog*, 167–169
- tracking statistics for each rule with labels, 169–171
- logical NOT (!) operator, 42
- log keyword, 162, 167
- log (matches) clause, 164–165

M

- MAC addresses
 - bridges, 87
 - filtering, 46–47, 46n2, 60
 - IP-based load balancing, 157–158
- Mac OS X, 3n3
- macros
 - defined, 18–19
 - defining, 18–19
 - defining local network, 29
 - expanding into separate rules, 20–21
 - usefulness of, 19–20
- mail servers
 - NAT, 79
 - routable IPv4 addresses, 66–69
- mail-in/mail-out labels, 170
- management information bases (MIBs), 182–183
- man pages, 9
- match rules, 31–32
 - debugging, 198
 - load balancing, 73–74, 79, 83
 - logging, 164–165
 - packet normalization, 193–194
 - spam, 103
 - tags, 85
 - traffic shaping, 119, 121–122, 124–126, 130, 132, 134, 137–138, 141–142
 - wireless networks, 54
- max-src-conn-rate* state-tracking option, 97
- max-src-conn* state-tracking option, 97

- max state-tracking option, 98
- McBride, Ryan, 5
- mekmitasdigoat passphrase, 154, 154n2
- MIBs (management information bases), 182–183
- Miller, Damien, 178, 182
- Morris worm, 2n1

N

- NAT (network address translation), 31, 71, 73, 79–84, 165
 - IPv6 versus, 28–29
 - release of, 28
 - wireless networks, 54–55, 61
- nat rule, 32
- nat-to keyword, 31–32, 54, 81, 83–84, 138, 164–165
- neighboradv (neighbor advertisements), 41
- neighbrsol (neighbor solicitations), 41
- NetBSD, 3n3, 5
 - bridge setup, 89–90
 - configuring wireless interface, 50
 - online resources, 204
 - setting up ALTQ
 - framework on, 136
 - setting up PF on, 15–16
 - spamd spam-deferral daemon, 101
- NetFlow, 176–182
 - collectors
 - choosing, 178
 - defined, 176
 - data collection with pfflowd, 182
 - flowd collector daemon, 177–182
 - flow-tools program, 177
 - nfdump program, 177
 - sensors
 - defined, 176
 - setting up, 176–177
- net-snmp package, 183
- network address translation (NAT), 31, 71, 73, 79–84, 165
 - IPv6 versus, 28–29
 - release of, 28
 - wireless networks, 54–55, 61
- nfdump tool, 177
- nixspam blacklist, 115
- nohup command, 168
- no-sync option, 156
- NTP, 33
- nwid parameter, 49, 56
- nwkey parameter, 50, 56

O

- oldqueue keyword, 133
- OpenBSD
 - approach to security, 2, 2n2
 - bridge setup, 87–88
 - configuration files, 7
 - configuring wireless interface, 50
 - history of, 3–5
 - purchasing, 205–206
 - setting up ALTQ framework on, 135
 - setting up PF on, 12–13, 12n1
 - wireless network setup, 56–57
 - WPA access points, 51–52
- operating system-based queue assignments
 - ALTQ framework, 145
 - priority and queuing system, 131
- optimization option, 192
- overload option, 97–99
 - ALTQ framework, 144–145
 - priority and queuing system, 130–131

P

- packet-filtering gateways, 25
 - FTP, 35–37
 - ftp-proxy with diversion or redirection, 36–37
 - variations on ftp-proxy setup, 37
 - simple, 25–34, 26f
 - defining local network, 29
 - in/out rules, 26–27
 - NAT versus IPv6, 28–29
 - setting up, 29–33
 - testing rule set, 34
 - tables, 42–43
 - troubleshooting-friendly networks, 37–41
 - letting ICMP pass, 38–39
 - path MTU discovery, 40–41
 - ping command, 39
 - traceroute command, 40
- Packet Filter subsystem. *See* PF (Packet Filter) subsystem
- packet forwarding, 30
- Packets In/Out statistics, 23
- packet tagging, 85–86
- pass all rule, 15, 22
- pass in rule, 26, 33
- pass out rule, 16–17, 27

- passtime value, 107
- path MTU (maximum transmission unit) discovery, 38, 40–41
- pf_rules= setting, 13
- PF (Packet Filter) subsystem, 1–2
 - displaying system information, 22–24
 - history of, 4–5
 - IPFilter configuration
 - compatibility, 4n5, 8–9
 - migrating from other systems, 6–9
 - copying across IPFilter configuration to OpenBSD, 8–9
 - Linux versus BSD, 6–7
 - porting to Linux machines, 7
 - rule syntax changes, 9
 - tools for configuration file management, 7–8
 - tools for converting network setups, 8
 - performance improvements, 5
 - purpose and function of, 3
 - rule set configuration
 - simple, 16–18
 - stricter, 18–22
 - setting up, 12–16
 - on FreeBSD, 13–15
 - on NetBSD, 15–16
 - on OpenBSD, 12–13
 - wireless access point rule set, 53–54
- pfctl command-line administration
 - tool, 11–12
 - debug level, 191
 - disabling PF, 12, 197
 - displaying system information, 22–23, 189
 - displaying verbose output, 20–21
 - enabling PF, 12, 13
 - expiring table entries, 99
 - fetching periodic data, 170
 - flushing existing rules, 22
 - list current contents of anchors, 92
 - load rules into anchors, 92
 - manipulating anchor contents, 92
 - memory pool information, 190
 - parsing rules without loading, 21
 - traffic tracking totals on per-rule basis, 169–170
 - viewing rule numbers and debug information, 197–198
- pfflowd tool, 182
- pflogd logging daemon, 162
 - logging to several interfaces, 167
 - logging to syslog, 168
- pflog(4) interface, 176–182
 - data collecting, reporting, and analysis, 177–182
 - setting up sensor, 176–177
- pfSense, 8
- pfstat command, 173–175, 175f
- pfsync protocol, 154–155
- pftop command
 - traffic monitoring, 173, 173n1
- ping6 command, 39
- ping command, 39
- ping of death bug, 38
- PPP, 31
- PPP over Ethernet (PPPoE), 31
- prio keyword, 119–121
- priority and queuing system, 118–131
 - handling unwanted traffic, 130–131
 - operating system-based queue assignments, 131
 - overloading to tiny queues, 130–131
 - queues for bandwidth allocation, 121–130
 - DMZ network with traffic shaping, 128–130
 - fixed, 123–125
 - flexible, 125–128
 - HFSC algorithm, 123
 - setting traffic priorities, 119–121
 - assigning two priorities, 120–121
 - prio priority scheme, 119–120
 - transitioning from ALTQ to, 131–133
- priq (priority) queues, 131–132, 134–138
 - match rule for queue assignment, 137–138
 - performance improvement, 136–137
- proactive defense, 95–115
 - spam, 100–114
 - blacklisting, 100–103
 - compensating for unusual situations, 113–114
 - content filtering, 100
 - detecting out-of-order MX use, 113

- proactive defense, spam (*continued*)
 - greylisting, 104–108
 - greytrapping, 109–111
 - list management with `spamdb`, 111–113
 - tips for fighting, 115
 - updating whitelists, 108–109
- SSH brute-force attacks, 96–99
 - defined, 96
 - expiring tables using `pfctl`, 99
 - overview, 96
 - setting up adaptive firewalls, 97–99

Q

- `qlimit` value, 125–126, 141
- queues. *See also* priority and queuing system
 - for bandwidth allocation, 121–122
 - DMZ network with traffic
 - shaping, 128–130
 - fixed, 123–125
 - flexible, 125–128
 - HFSC algorithm, 123
 - handling unwanted traffic
 - overloading to tiny queues, 130–131
 - queue assignments based
 - on operating system fingerprint, 131
- queue-scheduler algorithms (disciplines), 134–135
 - class-based bandwidth allocation, 132–133, 135
 - queue definition, 139–140
 - tying queues into rule set, 140
 - HFSC algorithm, 123, 125–126, 132–135
 - queue definition, 140–141
 - tying queues into rule set, 141–142
- priority-based queues, 131–132, 134–138
 - match rule for queue assignment, 137–138
 - performance improvement, 136–137
- quick rules, 33, 192, 198

R

- random early detection (RED), 137
- random option, 72–73
- `rc` script, 13–15, 30
- `rdr-anchor` anchor, 74
- `rdr-to` keyword, 36, 75, 80, 83, 103, 164
- realtime value, 141
- reassemble option, 192–193
- RED (random early detection), 137
- redirection
 - FTP, 36
 - for load balancing
 - NAT, 81
 - routable IPv4 addresses, 72–73
 - public networks, 62–63
 - with `relayd` daemon, 73–75
- redundancy and resource availability, 147–160
 - failover
 - CARP, 150–154
 - `pfsync` protocol, 154–155
 - rule set, 155–156
 - load balancing, 157–160
 - CARP in load-balancing mode, 158
 - setting up CARP, 158–160
 - redundant pair of gateways, 148–150, 149f
- Reed, Darren, 4
- `relayctl` administration program, 76–77
- `relayd` daemon, 73–79, 73n2
 - CARP, 79
 - checking configuration before starting, 76
 - checking interval, 75
 - HTTP, 77–78
 - SSL, 78
- relays, 73–75
- removals statistic, 23
- return value, 186
- `round-robin` option, 72
- `routeradv` (router advertisements), 41
- `routersol` (router solicitations), 41
- `rtadvd` daemon, 54
- `rtsol` command, 56, 58
- `ruleset-optimization` option, 191
- rule sets
 - atomic rule set load, 21
 - bridges, 90–91
 - defined, 11
 - evaluation of, 17

- queues for bandwidth allocation
 - fixed, 124–125
 - flexible, 126–128
 - restructuring with anchors, 91–94
 - simple, 16–18
 - overview, 16–18
 - testing, 18
 - stricter, 18–22
 - checking rules, 21–22
 - overview, 19–20
 - reloading and looking for errors, 20–21
 - testing, 22
 - using domain names and hostnames in, 34
 - wireless access point, 53–54
 - writing to default deny, 18n4
- S**
- sample configurations, 203–204
 - satellite value, 192
 - SCP, 35, 124, 139–140
 - scrub keyword
 - fragment reassembly options, 192–193
 - packet normalization, 193
 - Secure Shell. *See* SSH
 - self keyword, 32
 - Sender Policy Framework (SPF)
 - records, 114, 114n7
 - set skip on lo rule, 13, 15–16
 - SFTP, 35
 - Simple Network Management Protocol (SNMP), 182–183, 182n5
 - skip option, 187
 - SMTP, 22, 68–69, 95, 100–106, 108–110, 113–114, 164
 - SNMP (Simple Network Management Protocol), 182–183, 182n5
 - Solaris, 8–9
 - spam, 100–114
 - blacklisting, 100–103, 101–103
 - content filtering, 100
 - detecting out-of-order MX use, 113
 - greylisting
 - compensating for unusual situations, 113–114
 - defined, 104
 - function of, 106
 - in practice, 107–108
 - setting up, 104–105, 107
 - greytrapping, 109–111
 - list management, 111–113
 - keeping greylists in sync, 112–113
 - updating lists, 111–112
 - logging, 103
 - stuttering, 100–101
 - tarpitting, 100–101
 - tips for fighting, 115
 - updating whitelists, 108–109
 - SpamAssassin, 100
 - spamdb tool
 - adding/deleting whitelist entries, 111
 - greylisting, 104, 111–113
 - keeping lists in sync, 112–113
 - updating lists, 111–112
 - greytrapping, 110–112
 - adding to list, 111–112
 - deleting from list, 112
 - spamd spam-deferral daemon, 13, 100–114
 - blacklisting, setting up, 101–103
 - detecting out-of-order MX use, 113
 - greylisting, 104–108
 - compensating for unusual situations, 113–114
 - defined, 104
 - function of, 106
 - in practice, 107–108
 - setting up, 104–105, 107
 - greytrapping, 109–111
 - list management with spamdb, 111–113
 - keeping greylists in sync, 112–113
 - updating lists, 111–112
 - logging, 103
 - online resources, 205–206
 - updating whitelist, 108–109
 - spamlogd whitelist updater, 108–109, 167
 - SPF (Sender Policy Framework)
 - records, 114, 114n7
 - spoofing, 194–195, 194f
 - SSH (Secure Shell), 33, 48, 156
 - authpf program, 60
 - bandwidth allocation, 124, 139
 - brute-force attacks, 96–99
 - defined, 96
 - expiring tables using pfctl, 99
 - overview, 96
 - setting up adaptive firewalls, 97–99

- SSH (Secure Shell) (*continued*)
 - traffic prioritizing, 119
 - VPNs, 55
 - SSL encryption, 48, 78
 - state defaults, 177, 188
 - state-defaults option, 188
 - state-policy option, 187–188
 - state tables, 22–23, 182, 187–189
 - defined, 17
 - logging, 171, 174, 175f, 176
 - synchronizing, 154–155
 - State Table statistics, 23
 - state-tracking options, 97
 - sticky-address option, 72–73, 75
 - stuttering, 100–101
 - sudo command, 12, 14–16
 - symon utility, 175
 - sync listeners, 112
 - sync targets, 112
 - SYN-flood attacks, 68
 - synproxy state option, 68
 - sysctl command, 88, 158
 - setting up CARP, 151
 - turning on packet forwarding, 30
 - syslogd logging daemon, 167–169
 - sysstat command
 - redundancy and resource
 - availability, 155, 160
 - traffic monitoring views, 171–173, 173n1
 - traffic shaping, 127, 138, 142
 - system information, displaying, 22–24
- T**
- tables. *See also* state tables
 - brute-force attacks, 97, 99
 - expiring table entries, 99
 - loading, 42
 - manipulating contents of, 42–43
 - naming, 42
 - “probation”, 99
 - tagged keyword, 85, 87
 - tags, 85–86
 - tarpitting, 100–101
 - TCP
 - ALTQ priority queues, 137
 - NetFlow data collection, 176, 179, 181
 - ports, 35
 - protocol handler definitions, 78
 - strict rule sets, 21–22
 - tcpdump program, 198
 - two-priority configuration, 120
 - UDP versus, 20
 - tcpdump program, 162–163, 166, 168, 198–199
 - TCP/IP, 3
 - ATLQ, 134
 - bridges, 86
 - FTP, 35n5
 - NetFlow data collection, 176
 - network interface configuration, 24
 - packet filtering, 31
 - redundancy and resource
 - availability, 154
 - total usable bandwidth, 122
 - troubleshooting-friendly networks, 37, 40
 - wireless networks, 46, 49, 56–57, 62
 - testing, 195–196, 196t
 - timeout option, 188–189
 - to keyword, 26–27
 - traceroute6 command, 39
 - traceroute command, 39
 - traffic shaping, 117–145
 - ALTQ framework, 117–118, 133–145
 - basic ALTQ concepts, 134
 - class-based bandwidth
 - allocation, 139–140
 - handling unwanted traffic, 144–145
 - HFSC algorithm, 140–142
 - priority-based queues, 136–145
 - queue-scheduler algorithms, 134–135
 - queuing for servers in DMZ, 142–144
 - setting up, 135–136
 - priority and queuing system, 118–131
 - handling unwanted traffic, 130–131
 - queues for bandwidth
 - allocation, 121–130
 - setting traffic priorities, 119–121
 - transitioning from ALTQ to, 131–133
 - trojans (trojan horses), 2

- troubleshooting-friendly networks, 37–41
 - letting ICMP pass
 - unconditionally, 38
 - while stopping probes from elsewhere, 39
 - path MTU discovery, 40–41
 - ping command, 39
 - traceroute command, 40
- two-priority configuration, 120–121, 132

U

- UDP, 21, 33, 40, 61, 168
 - IPsec with UDP key exchange, 55
 - NetFlow data collection, 176–177, 179
 - TCP versus, 20
- up parameter, 49, 56
- upperlimit value, 141
- user_ip macro, 62

V

- verbose output
 - flowd-reader program, 178–179, 181
 - pfctl administration tool, 20–21
 - spamd spam-deferral daemon, 102, 107
- vhid (virtual host ID) parameter, 152
- virtual local area networks (VLANs), 70f
- virtual private networks (VPNs), 55
- Virtual Router Redundancy Protocol (VRRP), 148, 152
- viruses, defined, 2
- VLANs (virtual local area networks), 70f
- VoIP (Voice over Internet Protocol), 119–120
- VPNs (virtual private networks), 55
- VRRP (Virtual Router Redundancy Protocol), 148, 152

W

- web servers
 - NAT, 79
 - routable IPv4 addresses, 66–67, 72, 74–75, 77
- WEP (Wired Equivalent Privacy), 47, 59
- whiteexp value, 107
- whitelists, 101–102, 105
 - adding/deleting entries, 111
 - keeping updated, 108–109
- wicontrol command, 46n1
- Wi-Fi Protected Access. *See* WPA
- Wired Equivalent Privacy (WEP), 47, 59
- wireless networks, 45–63, 205
 - guarding with authpf, 59–63
 - basic authenticating gateways, 60–62
 - public networks, 62–63
 - privacy mechanisms
 - MAC address filtering, 46–47
 - WEP, 47
 - WPA, 47–48
 - selecting hardware for, 48
 - setting up, 48–59
 - access point PF rule set, 53–54
 - access points with three or more interfaces, 54–55
 - client side, 55
 - configuring interface, 49–51
 - FreeBSD setup, 58–59
 - FreeBSD WPA access points, 52–53
 - initializing card, 48–49
 - OpenBSD setup, 56–57
 - OpenBSD WPA access points, 51–52
 - VPNs, 55
- worms, 2, 2n1
- WPA (Wi-Fi Protected Access), 47–48, 59
 - FreeBSD access points, 52–53
 - OpenBSD access points, 51–52
- wpakey parameter, 56