Symbols

@m (computer virus naming conventions), 42
@mm (computer virus naming conventions), 42
@mm worms (mass-mailer worms), 29
3 Tunes (virus), 92
3APA3A (virus), 116
16-bit Windows
  EPO (entry-point obscuring) viruses, 147-150
  NE viruses, 60
32-bit address spaces. See virtual memory systems (Windows NT)
32-bit polymorphic viruses, 264-268
32-bit Windows. See Win32
64-bit platforms, kernel mode scanning on, 530-531
64-bit Windows, PE viruses, 61
911 attacks, 308
1260 virus, self-protection technique, 261-262

A
ABAP viruses, 89
ABAP/Rivpas (virus), 89

access
counterattacks, 596
Dumaru (worm), 640
early warning systems, 598
firewalls, 588-589
honeypot systems, 593-594
network intrusion detection system (NIDS), 591-592
router access lists, 585-587
worm behavior patterns, 598-608
accidentally destructive payload viruses, 297
ACG (Amazing Code Generator) virus, 270, 277
code emulation, 463-464
disassembling, 463
heuristics, 465
self-protection technique, 253
Acrobat, PDF viruses, 90
ActionScript viruses, 91
activation methods. See payload activation
active instructions, tracking, 454
active pages, patching code in, 522
ActiveX controls
  rights verification, 388
  safe-for-scripting, 388-389, 417-419
Address Resolution Protocol (ARP) requests, 595
address-book worms, 319
address spaces
  process randomization, 570
  return-to-LIBC attacks, 569-573
  upper 2G of address space (memory scanning), 527
  user address space of processes (scanning), 523
  virtual address spaces (Windows NT), 501-505
addresses
  GOT/IAT page attributes, 574
  virtual, translation of, 500
AddressOffEntryPoint field (PE header), 164
Adleman, Leonard, 18
ADM (worm), avoiding buffer overflow attacks, 413
administration
  memory, 498-499
  Virtual Memory Manager, 503
Admiral Bailey (virus writer), IVP (Instant Virus Production Kit), 292
Adobe Acrobat, PDF viruses, 90
Adore (rootkit), 36
adware, definition of, 38
AIDS Information Diskette (Trojan horse), 31, 305
Alcopaul (virus writer), W32/Sand.12300 virus, 140
alerts, DeepSight, 598
algorithmic detection, metamorphic viruses, 271
algorithmic scanning methods, 441-443
  filtering, 443-444
  static decryptor detection, 444-446
  X-RAY method, 446-451
algorithms, Boyer-Moore, 431
Aliz (worm), 644
ALS/Burstead (virus), 92
altering module, 592
Amazing Code Generator (ACG) virus. See ACG virus
AmiPro viruses, 94-95
Amoeba (infection technique), 140
analysis, malicious code analysis techniques, 612
  architecture guides, 615
  collection maintenance, 661
  dedicated system installation, 612, 615
  Digital Immune System, 661-664
disassemblers, 626-632
dynamic analysis techniques, 634-655
  knowledge bases, 615-616
  process of, 618-625
  unpacking, 625
  Virus Analysis Toolkit (VAT), 656-659
  VMWARE, 616-617
Anarchy.6093 (virus), 112
ANIMAL (game), 17
Anna Kournikova virus, 35, 292
ANSLSYS drivers, reconfiguring key functions, 90-91
Anthrax (virus), 210
Anti-AVP (virus), self-protection technique, 248
ANTI-VIR.DAT file (antivirus program), 248
AntiCMOS (virus), 306
antidebugging techniques (armored viruses), 226-234
antidisassembly techniques (armored viruses), 220-226
antiemulation techniques (armored viruses), 242-247
antiheuristics techniques (armored viruses), 234-242
AntiPascal (virus), 302
INDEX

antivirus defense techniques, 426-427
antivirus programs. See also disinfection methods
“Are you there?” calls, 199
behavior-blocking programs, 19
disabling with retroviruses, 247-249
half-cooked repairs, 136
history of, 27-28
integrity checker programs, 19
modeling virus infections, 11-12
scanning, 252
testers, 672
vendor contact information, 670
versus computer security companies, 366-367
antivirus viruses, 357
API hooking (infection technique), 150-151
API strings, 241-242
APIs, control transfer, 246
ApIS/Simpsons@mm (worm), 90
APM/Greenstripe (virus), 95
appending viruses (infection technique), 132-133, 174-175, 240-241
AppleScript viruses, 90
applications
algorithmic scanning methods. See algorithmic scanning methods
antivirus defense techniques, 426-427
code emulation. See code emulation disinfection methods, 474-477
first-generation antivirus scanners. See first-generation antivirus scanners
heuristic analysis, 467-474
metamorphic virus detection. See metamorphic virus detection
rights verification, 388
second-generation antivirus scanners. See second-generation antivirus scanners
architecture dependency. See computer architecture dependency
architecture guides, malicious code analysis techniques, 615
archive format dependency, 100
“Are you there?” calls (self-detection technique), 198
arenas (sections of memory), 498
armored viruses, 220
antidDebugging techniques, 226-234
antidisassembly techniques, 220-226
antiemulation techniques, 242-247
antigoat techniques, 247
antiheuristics techniques, 234-242
ARP (Address Resolution Protocol) requests, 595
“Art of the Fugue” (Bach), 5
art versus science, 4
ASPACK (run-time packer), 625
Atkinson, Bill, 91
attachment inserters (worm infections), 334
attacks. See also blended attacks; buffer overflow attacks; viruses; worm blocking techniques
against memory scanning, 532-533
algorithmic scanning methods. See algorithmic scanning methods
antivirus defense techniques, 426-427
code emulation. See code emulation
code injection attacks, 341-342, 543
dictionary attacks, 324
DoS (denial of service) attacks, 306-308, 539
e-mail worm attacks, 333-334
executable code-based attacks, 339
file parsing attacks, 319-320
first-generation antivirus scanners. See first-generation antivirus scanners
future, 575-578
heuristic analysis, 467-474
injected code detection, 557-562
instant messaging attacks, 333
Linux/Slapper, 647
metamorphic virus detection. See metamorphic virus detection
network share enumeration, 324-326
network-level defense strategies. See network-level defense strategies
NNTP attacks, 338
password-capturing attacks, 325
peer-to-peer network attacks, 332-333
phishing attacks, 308-309
remote login-based attacks, 341
return-to-LIBC, 543, 569-573
second-generation antivirus scanners.
    See second-generation antivirus scanners
shell code-based attacks, 342-344
SMTP proxy-based attacks, 334-335
SMTP-based attacks, 335-338
stack smashing, 546
vampire attacks, 358
attributes, GOT/IAT page, 574
authenticated updates (worm infections), 346-351
auto-rooters, definition of, 34
AutoLisp viruses, 92-93
automata. See cellular automata;
    self-replicating systems
automated analysis, Digital Immune System, 661-664
automated exploit discovery, 578
AUTORUN.INF file viruses, 97
AV-Test.org, 672
AVP (antivirus software), 248
Azusa (virus), infection technique, 125

Badboy (virus), self-protection technique, 260, 271
Badtrans (worm), 366
BAT/Batalia (virus), 82
BAT/Hexvirus (virus), 82
BAT/Mumu (virus), 83
    weak passwords, 324
BAT/Polymba (virus), 82
BAT/Ramble (virus dropper), 96
BAT/Zipbat (virus), 82
BATCH viruses, 82
BATVIR (virus), 82
Beast (virus), 112
behavior blockers, definition of, 19, 209
behavior patterns (worms), 598-608
Belcebu, Billy (virus writer), 233
beneficial viruses, 357
Benny (virus writer)
    W2K/Installer virus, 137
    W32/Donut virus, 99
    W32/HIV virus, 59
    W32/Press virus, 78
Bergroth, Ismo, 496
BHP (virus), 57-58
binary viruses
    computer architecture dependency, 52
    CPU dependency, 53-54
    operating system dependency, 55
BIND (Berkeley Internet name domain) servers, Linux/ADM worm, 397
BioWall project Web site, 12
Bizatch (virus), 61
Black Baron (virus writer), 448
black boxing, 624
black-box testing, 634
BlackIce firewall, 646
blank passwords, danger of, 324
Blaster (worm). See W32/Blaster (worm)
blended attacks. See also buffer overflow attacks
    danger of, 366-367
    defined, 366
INDEX

history of, 367-368
blocking
buffer overflow attacks (worms). See buffer overflow attacks (worms)
Microsoft SQL Server exploits, 559-560
scripts, 539-541
self-sending code blocking, 563-565
shellcode, 558-562
SMTP, 539-541
W32/Blaster (worm) exploits, 561
W32/CodeRed (worm), 542, 560-561, 564-565
W32/Slammer (worm), 542-564
W32/Welchia (worm) exploits, 562
blocking mode, 592
Bluetooth and wireless mobile worms, 359-361
Bochtch, Vesselin, 39, 61, 74-75, 349, 447, 633, 661
bookmarks, first-generation antivirus scanners, 433-434
boot sector viruses. See boot viruses
boot strap loader, 122
replacement of, 124-125
boot viruses, 122-124
computer architecture dependency, 52
DBR (DOS BOOT record) infection techniques, 126-129
encryption, 303-304
hooking INT 13h (interrupt handler), 191-193
installation, 197
interrupt hooking, 188
MBR (master boot record) infection techniques, 124-126
over networks, 129
in Windows 95, 129
Borland Quattro spreadsheet program, 187
Brain (virus), 52, 122, 197, 200, 497
attack by Denzuko virus, 127
competition between viruses, 357
read stealth viruses, 203
break points
detecting, 227
removing, 233
stopping, 454
broadcast segmentation technique, 353
Brown, Ralf, 615
Brunner, John (Shockwave Rider), 29
brute-force decryption, RDA viruses, 245, 256
BSD/Scalper (worm), 327, 353, 401, 406, 543
.bss section (PE files), 167
buffer overflow attacks (worms), 538-542
avoiding, 413-414
blocking, 543-544
code reviews, 544
CodeRed worm, 398-401
compiler-level solutions, 545-552
definition of, 368-369
first-generation buffer overflows, 369-371
kernel-mode extensions, 554-556
Linux/ADM worm, 397-398
Linux/Slapper worm, 401-407
Morris worm, 367, 395-397
operating system-level solutions, 552-554
program shepherding, 556
script/SMTP blocking, 539-542
second-generation buffer overflows, 371-378
subsystem extensions, 554
third-generation buffer overflows, 378-394
W32/Blaster worm, 410-413
W32/Slammer worm, 407-410
Buffer Security Check feature, 552
BugTraq databases, 598
Bumblebee (virus writer), W32/RainSong virus, 152
Burger, Ralf (virus writer), Virdem virus, 135
Burglar.1150.A (virus), system modification attacks, 391
Burks, Arthur, 6
Butler, Max, 397

679
INDEX

C

Cabanitas. See W32/Cabanas (virus)
cache bypass vulnerability, W32/Blebla
worm, 419
cache viruses. See disk cache viruses
calc.exe, 619
CALL-to-POP trick, 240-241
calls, system tracing, 647-648
canonicalization, 385-386
captures
Linux/Slapper (worm), 600-602
network traffic, 643
W32/Blaster (worm), 598-600
W32/Sasser.D (worm), 603
W32/Slammer (worm), 607-608
W32/Welchia (worm), 605
CARO (Computer Antivirus Researchers Organization), 38
Cascade (virus), 24-26, 53, 59
nondestructive payload viruses, 298
self-protection technique, 230, 253
X-RAY scanning, 447
cavity viruses (infection technique), 136-137
CBAC (context-based access control), 586
CC hack, 104
CEF file format, 111
cell phones, worms on, 359-361
cellular automata (CA) computer architecture, 6. See also self-replicating systems
Edward Fredkin structures, 7-8
game of Life (Conway), 8-12
chain letters, definition of, 37
Characteristics field (PE header), 164
check bytes. See bookmarks
checksum
API strings, 242
CRC checksum, 248
detecting break points, 227
recalculation, 239
as self-protection technique, 224-225
Checksum field (PE header), 165
Cheeba (virus), self-protection technique, 257
Cheese (worm), 315, 318
Chess, Dave, 26, 277
Cheswick, Bill, 593
Chi, Darren, 75
CHRISTMA EXEC worm, 78-79
Cisco routers. See routers
classic parasitic viruses (infection technique), 135-136
cleaning goat files, 639
Clementi, Andreas, 673
cluster preponent infection method, 57
cluster viruses, file system dependency, 56-58
cluster-level stealth viruses, 207-208
CMOS viruses, 306
Codd, E.F., 6
code
in active pages, patching, 522
injected code detection, 557-562
malicious code analysis techniques. See malicious code analysis techniques
quick examination during computer virus analysis, 621
self-sending code blocking, 563-565
versus data in von Neumann machines, 5
code builders (infection technique), 155-156
code confusion. See obfuscated code
code emulation, 451-454
antiemulation techniques (armored viruses), 242-247
dynamic decryptor detection, 459-461
encrypted/polymorphic virus detection, 455-458
metamorphic virus detection, 463-466
code emulation-based tunneling, 219
code evolution, 252-253
code injection attacks, 341-342, 398-401, 543
code integration viruses (infection technique), 155, 278-281
code propagation techniques (worms), 338
code injection attacks, 341-342
executable code-based attacks, 339
HTML-based mail, 340
INDEX

links to Web sites or proxies, 339-340
remote login-based attacks, 341
shell code-based attacks, 342-344
code redirection, 469
code reviews, buffer overflow attacks
(worms), 544
code sections
  naming, 469
  packing, 237
  PE entry points, 468
  random entry point, 237-238
  sizes in header, 241
  writeable flag, 238
CodeGreen (antiworm). See W32/CodeGreen
  (antiworm)
CodeRed (worm). See W32/CodeRed (worm)
CodeRed_II (worm), 310, 520
Cohen, Frederick, 18, 302
definition of computer viruses, 18-20
history of antivirus programs, 27
Coke. See W32/Coke (virus)
collection (viruses) maintenance, 661
COM viruses, 59
combined attacks. See blended attacks
Commander_Bomber (virus), infection
technique, 142-143
companion viruses (infection technique),
  18, 176
competition between viruses, 357-358
compiler alignment areas, recycling, 238
compiler dependency, 108-109
compiler-level solutions, buffer overflow
  attacks (worms), 545-546
  Microsoft Visual .NET, 2003 (7.0 & 7.1),
    549-552
  ProPolice, 548-549
  StackGuard, 546-548
compressing viruses (infection technique),
  139-140
  file system dependency, 59
  compression
    PE file-infection techniques, 235
    run-time packers, 625
    as self-protection technique, 225-226
Computer Antivirus Researchers
  Organization (CARO), 38
computer architecture dependency, 52-53
computer security companies versus
  antivirus programs, 366-367
computer simulations of nature. See
  nature-simulation games
computer virus analysis, process of, 618-624
computer virus research. See virus research
computer viruses. See viruses
computer worms. See worms
compilers, modeling virus infections, 11-12
connections, worm blocking techniques,
  574-575. See also network-level defense
  strategies
construction kits. See virus construction kits
contagion worms, 576
context-based access control (CBAC), 586
control transfer with APIs, 246
Conway, John Horton (game of Life), 8-12
cookies, security_cookie values, 550
cooperaion between viruses, 354-357
coprocessor instructions, 242-243
copy-protection software, extra disk
  sectors, 127
copycat worms. See worm blocking
  techniques
Core War (game), 12-16, 534
Core Wars instructions (1994 revision), 14
Corel Script viruses, 95
corruption of macro viruses, 69-71
counterattacks, 596
CPU dependency, 53-54
CPU instructions, undocumented, 245
CPUs, Win32 platform support, 159
CR0 control registers, 529
CRC checksums, 248
CreateFile() API, 232-233
CreateProcess() API, 559
Creeper (virus), 17
cross-platform binary viruses, 52
Cruncher (virus), infection technique, 139
Crypto API, 257
cryptographic detection, 446
cryptography, AIDS TROJAN DISK Trojan horse, 31
Cryptor (virus), 232
Csakany, Antal, 11
CSC/CSV (virus), 95
CSC/PVT (virus), 95

DBR (DOS BOOT record), infection techniques, 126-129
DCL viruses, 79-80
DDoS (distributed denial of service) attacks, 36
de Wit, Jan, 35
dead virus code, reviving, 127
debug interfaces, tracing with, 219
debug registers, clearing, 232
 .debug section (PE files), 168
debugger dependency, 106-108
debugging, 648-651, 655
antidebugging techniques (armored viruses), 226-234
DEC/VMS systems, DCL viruses, 79-80
deception, e-mail worm attacks, 333-334
decoders, packets, 591
decryption. See also encryption
backward decryption, 230
disassemblers, 626-632
nonlinear decryption, 256
RDA viruses, 245
with stack pointer (SP), 230
decryptors
dynamic detection, 459-461
static detection, 444-446
tracking, 454
dedicated virus analysis systems
installation of, 612-615
VMWARE, 616-617
DeepSight alerts, 598
Demon Emperor (virus writer), Hare virus, 129, 255
denial of service (DoS) attacks, 35, 306-308, 539
against Windows Update Web site, 413
Denzuko (virus)
competition between viruses, 357
infection technique, 127
dependencies
  archive format dependency, 100
  compiler and linker dependency, 108-109
  computer architecture dependency, 52-53
  CPU dependency, 53-54
  date and time dependency, 98
  debugger dependency, 106-108
  device translator layer dependency,
    109-112
  embedded object insertion dependency,
    112-113
  extension dependency, 101-102
  file format dependency, 59-66
  file system dependency, 56-59
  host size dependency, 105-106
  interpreted environment dependency,
    66-98
  JIT dependency, 99-100
  language dependency of macro viruses,
    71-72
  multipartite viruses, 115-116
  network protocol dependency, 102
  operating system dependency, 55
  operating system version dependency,
    55-56
  platform dependency of macro viruses,
    73-74
  Registry-dependent viruses, 93-94
  resource dependency, 104-105
  self-contained environment dependency,
    113-115
  source code dependency, 102-104
  vulnerability dependency, 98
destructive payload viruses
  highly destructive payloads, 301-306
  somewhat destructive payloads, 300-301
detection. See also first-generation antivirus scanners; second-generation antivirus scanners
  active viruses in memory, 497
  cryptographic, 446
  direct library function invocations,
    571-573
  dynamic decryptor, 459-461
  engines, 592
  geometric, 461-462
  injected code, 557
    shellcode blocking, 558-562
  network intrusion detection system
    (NIDS), 584, 591-592
  static decryptor, 444-446
  threads, 518-521
device driver viruses, 65
device translator layer dependency, 109-112
[<devolution>] (computer virus naming
conventions), 41
devolution of macro viruses, 74-75
Dewdney, A.K., 13
dialers, definition of, 33
dictionary attacks, 324
Digital Immune System, 661-664
Digital Millennium Copyright Act
  (DMCA), 596
DIR-II (virus), 56
direct library function invocations, detection
  of, 571-573
direct-action viruses, 186
directories, page (memory), 500
directory stealth viruses, 200-203
dirty memory pages, 455
disassemblers, 624
  antidisassembly techniques (armored viruses), 220-226
  malicious code analysis techniques,
    626-632
  metamorphic virus detection, 462-463
discovery of automated exploits, 578
disinfection methods, 474-475. See also
  antivirus programs; memory scanning
  generic decryptors, 477
  standard, 475-477
disk access with port I/O, 219
disk cache viruses, 209-210
Disk Killer (virus), 128, 303
Dispatch routine of DeactivatorDrivers, 529
INDEX

distributed denial of service (DDoS) attacks, 36
divide-by-zero exceptions, 229
DLL viruses, 62-63
DLLs
  disinfecting, 523
  linking to executables, 168-171
DMCA (Digital Millennium Copyright Act), 596
Donut (virus). See W32/Donut (virus)
Doomed (game), 113
Doomjuice (worm). See W32/Doomjuice (worm)
DOS
  cluster and sector-level stealth viruses, 207-208
  COM viruses, 59
  EPO (entry-point obscuring) viruses, 145-147
  EXE viruses, 60
  full-stealth viruses, 205-206
  interrupt hooking, 188-196
  memory-resident viruses, 196-199
  metamorphic viruses, 270
  system buffer viruses, 209
  TSR (Terminate-and-Stay-Resident) programs, 187
  undocumented interrupt (Int, 21h/52h function), 498
DoS (denial of service) attacks, 35, 306-308, 539
  against Windows Update Web site, 413
DOS BOOT record (DBR), infection techniques, 126-129
DOS stub in PE header, 162
“double extensions,” 81
down-conversion of macro viruses, 71
downloaders, definition of, 33
Doxtor L (virus writer), W32/Idle virus, 153
DR. DR. STROBE & PAPA HACKER (virus writers), 57
Dream (virus), 89
driver-list scanning, detecting debuggers, 230
drivers
  filter, 427, 527-529
  kernel-mode, 503
  lists of, 527
droppers, definition of, 33-34
Dukakis (virus), 91-92
Dumaru (worm), 635, 640
dumps
  PEDUMP, 645
  strings, 623-624
Dustbin, 619
Dwarf (Core War warrior program), 14-15
dynamic analysis techniques, 634-655
dynamic decryptor detection, 459-461
dynamic heuristics, 234
dynamic link library viruses, 62-63
dynamically allocated memory. See heaps

E
e-mail
  executable code-based attacks, 339
  HTML-based mail, 340
  worm infections, 333-334
e-mail addresses
  harvesting, 319-324
  parsing files for, 320
e-mail attachment inserters (worm infections), 334
eyear early warning systems, 598, 669
Easter eggs, definition of, 30
ecophagy, 7
.edata section (PE files), 167
Eddie (virus), 218, 303
Eddie-2 (virus), 200
EICAR (European Institute for Computer Antivirus Research), 672
ELF viruses, 64-65
Elk Cloner (virus), 17, 52
EMACS viruses, 87
embedded decryptor (infection technique), 141-142
embedded decryptor and virus body
(infection technique), 142-143
embedded object insertion dependency,
112-113
emulation. See code emulation
encoding URLs, 385-386
encrypted viruses, 253-258
encryption, 221-222, 303-305. See also
decryption
of host file headers, 236
Linux/Slapper worm, 406
virus detection, 455-458
W95/Marburg virus, 632
X-RAY algorithmic scanning method,
446-451
entry points
obfuscation, 233
random entry points in code section,
237-238
entry-point obscuring viruses (infection
technique), 143-155, 237, 443, 459
W32/Simile virus, 282
entry-point scanning, first-generation
antivirus scanners, 435-436
enumeration
network enumeration attacks, 393-394
of network shares, 324-326
processes, 517
environments of malicious code, 50-52
archive format dependency, 100
compiler and linker dependency, 108-109
computer architecture dependency, 52-53
CPU dependency, 53-54
date and time dependency, 98
debugger dependency, 106-108
device translator layer dependency,
109-112
embedded object insertion dependency,
112-113
extension dependency, 101-102
file format dependency, 59-66
file system dependency, 56-59
host size dependency, 105-106
interpreted environment dependency,
66-98
JIT dependency, 99-100
multipartite viruses, 115-116
network protocol dependency, 102
operating system dependency, 55
operating system version dependency,
55-56
resource dependency, 104-105
self-contained environment dependency,
113-115
source code dependency, 102-104
vulnerability dependency, 98
EPO viruses. See entry-point obscuring
viruses (infection technique)
error detection and correction with
Hamming code, 233
ESC sequences, reconfiguring, 90-91
Etap.D (virus), 53, 64
ETG (executable trash generator) engine, 280
Ethereal
Linux/Slapper (worm), 601
W32/Aliz@mm (worm) captures, 644
W32/Blaster worm, 599
W32/Sasser.D (worm), 603
ethics of using virus construction kits, 293
Etoh, Hiroaki, 548
European Institute for Computer Antivirus
Research (EICAR), 672
Evol (virus). See W32/Evol (virus)
evolution
macro viruses, 74-75
virus code, 252-253
exact identification, 439-441
Excel viruses. See macro viruses
exception handlers, 232
CodeRed worm, 400-401
exception-handler validation, 565-569
exceptions
generating, 229
structured exception handling, 243-244
EXE viruses, 60
INDEX

Exebug (virus), 123
execute macro viruses, 75-76
executables code-based attacks, 339
executables trash generator (ETG) engine, 280
executables linking DLLs to, 168-171
executed images (Win32 viruses), 512-514
ExecuteOnly attribute (Novell NetWare),
attacks via, 389-393
execution, random execution logic, 244-245
execution environments. See environments of
malicious code
execv() function, 647
exploits. See also blended attacks;
 vulnerability dependency
 automated discovery, 578
definition of, 33
W32/Slimmer (worm), 607-608
export table (PE files), 171-172
exporting functions, 171-172
extended access lists, 586
Extended Memory Specification (XMS), 198
extended tiny encryption algorithm
(XTEA), 346
extension dependency, 101-102
extensions
 kernel-mode, 554-556
subsystems, 554
extra disk sectors, formatting, 126-128

F
F-PROT (antivirus program), 195, 438,
441, 451
F1 key, Help file viruses, 89
false positives, signatures, 608
family_name> (computer virus naming
conventions), 40
FAT file systems, cluster viruses, 56-58
Father Christmas (worm), 79-80, 102
FC (File Compare), 622
Ferenc, Leitold, 673
Ferrie, Peter, 75, 154
File Compare tool, 645
file extension dependency, 101-102
file format dependency, 59-66
file formats, obfuscation, 233
file infection techniques. See infection
 techniques
File Monitor log, 635
file parsing attacks, 319-320
file stealth viruses, 207-208
file structure infection, Win32, 239
file system dependency, 56-59
file systems, filter drivers, 427
file viruses, hooking INT 21h (interrupt
handler), 193-196
FileAlignment field (PE header), 165
files
 goat (natural infection testing), 637-638
IDA command script (IDC), 631
images, scanning, 517
monitoring, 635-637
Filler (virus), 127, 198, 302
filter driver virus deactivation (memory
scanning), 527-529
filtering
algorithmic scanning methods, 443-444
drivers, 427
as process of computer virus analysis,
619-621
fingerd program, Morris worm attack
against, 395
fingerprinting worm targets, 326-330
Finnpoly (virus), 53
firewalls, 588-589, 646
first-generation antivirus scanners, 428
bookmarks, 433-434
equipment scanning, 435-436
fixed-point scanning, 435-436
generic detection, 432
hashing, 432-433
hyperfast disk access, 436
mismatches, 432
string scanning, 428-430
INDEX

top-and-tail scanning, 435
wildcards, 430-431
first-generation buffer overflows, 369-371
first-generation Windows 95 viruses, 172-173
FitzGerald, Nick, 39
fixed-point scanning, first-generation
antivirus scanners, 435-436
flags, suspicious combinations of, 471
Flash ActionScript viruses, 91
Flash BIOS viruses, 305-306
Flip (virus), somewhat destructive payload
viruses, 300
flirt signatures, 628
flooders, definition of, 35
Ford, Richard, 74
Form (virus), infection technique, 128
format specifiers, 379
format string attacks, 378-384
formatting extra sectors, 126-128
formula macros, 77
FPU instructions, 242-243
definition of, 32-33
generics, definition of, 99
Geometer, Michael, 88
generics, definition of, 99
geometric detection, 461-462
germs, definition of, 32-33
GetProcAddress() function, 522, 645
GetProcAddress() function, 522, 645
Geombot (virus), 88
Gigabyte (virus writer)
Darkness virus, 88
JIT-dependent viruses, 99
Logic worm, 83-85

INDEX

top-and-tail scanning, 435
wildcards, 430-431
first-generation buffer overflows, 369-371
first-generation Windows 95 viruses, 172-173
FitzGerald, Nick, 39
fixed-point scanning, first-generation
antivirus scanners, 435-436
flags, suspicious combinations of, 471
Flash ActionScript viruses, 91
Flash BIOS viruses, 305-306
Flip (virus), somewhat destructive payload
viruses, 300
flirt signatures, 628
flooders, definition of, 35
Ford, Richard, 74
Form (virus), infection technique, 128
format specifiers, 379
format string attacks, 378-384
formatting extra sectors, 126-128
formula macros, 77
FPU instructions, 242-243
definition of, 32-33
generics, definition of, 99
Geometer, Michael, 88
generics, definition of, 99
geometric detection, 461-462
germs, definition of, 32-33
GetProcAddress() function, 522, 645
GetProcAddress() function, 522, 645
Geombot (virus), 88
Gigabyte (virus writer)
Darkness virus, 88
JIT-dependent viruses, 99
Logic worm, 83-85
INDEX

Ginger (virus), 198
  infection technique, 126
  self-protection technique, 248
“glider” starting structure (game of Life), 10
  global offset table (GOT), 570
  page attributes, 574
GML (Game Maker Language), 113-114
  goat files
    antigoat techniques (armored viruses), 247
    natural infection testing, 637-638
GoldBug (virus), 198
Good Times hoax, 37
GOT (global offset table), 570
  page attributes, 574
Gömb (virus), nondestructive payload
  viruses, 299
Green, Andy, 347
GriYo (virus writer), 27
  symbiosis project, 356
  W32/CTX and W32/Dengue viruses, 150
  W32/Parvo worm, 321
  W95/HPS and W95/Parvo viruses, 264
.<group_name> (computer virus naming
  conventions), 41
Gryaznov, Dmitry, 257, 619

H
hackers, 12
  half-cooked repairs, definition of, 136
Hamming, Richard, 233
  Hamming code, error detection and
correction, 233
Happy99 (worm), 29, 62, 314, 350
  e-mail address harvesting, 322-323
  NNTP attacks, 338
  nondestructive payload viruses, 299
  hard-coded API addresses, 172-173
  hardware destroying viruses, 305-306
  hardware-level stealth viruses, 208-209
Hare (virus)
  infection technique, 129
  self-protection technique, 255
  harvesting e-mail addresses (worms), 319-324
  hashing, first-generation antivirus scanners,
  432-433
  header, PE files, 162-165
  header infection viruses (infection
technique), 173
  heap management, 384-385
  heap overflows, 373-374
    compiler-level solutions, 546
    exploiting, 375-376
    Linux/Slapper worm, 401-407
  heaps
    definition of, 373
    exception-handler validation, 568
Heyne, Frank, 637
  hidden window procedure (Win32
viruses), 512
HIEW tool, 621, 633, 639
  High Memory Area (HMA), 198
  high-interaction honeypot systems, 593
  highly destructive payload viruses, 301-306
history
  antivirus programs, 27-28
  blended attacks, 367-368
  computer viruses, 17-18
  self-replicating systems, 4-16
  Win32 viruses, 157
hit list method. See IP addresses, scanning
hive, definition of, 93
HLP/Demo (virus), 89
HMA (High Memory Area), 198
hoaxes, definition of, 37
holes in memory, 197
Honeyd, 595
INDEX

honeypot systems, 593-594
hook table for Frodo virus, 205-206
hooking
  API hooking (infection technique), 150-151
  function call-hooking (infection technique), 151-152
  IAT (import address table), 201-203
  interrupts, 188-196, 226
host application mutation (metamorphic viruses), 276-277
host file headers, encryption, 236
host size dependency, 105-106
host-based intrusion prevention techniques, 538-542
  buffer overflow attacks
    blocking, 543-544
    code reviews, 544
    compiler-level solutions, 545-552
    kernel-mode extensions, 554-556
    operating system-level solutions, 552-554
    program shepherding, 556
    subsystem extensions, 554
    script/SMTP blocking, 539-542
HTML files, WebTV worms, 86-87
HTML viruses, 97-98
HTML-based mail, 340
HybrisF (virus). See W32/HybrisF (virus)
HyperCard, HyperTalk viruses, 91-92
hyperfast disk access, first-generation antivirus scanners, 436
HyperTalk viruses, 91-92
Hyphervisor (virus), 310
Hypponen, Mikko, 326, 349, 496

IAT (import address table), 161, 522
  hooking, 201-203
  page attributes, 574
  patches, 469
IBM Antivirus, mismatches, 432
IBM systems, REXX viruses, 78-79
ICA, harvesting e-mail addresses using, 322
ICMP (Internet control message protocol), 643
ICSA Labs, 672
IDA command script (IDC) files, 631
IDA disassemblers, 221, 428, 626-632
.idata section (PE files), 167
IDC (IDA command script) files, 631
IDEA (virus)
  nondestructive payload viruses, 299
  self-protection technique, 256
IDEA.6155 (virus), self-protection technique, 248
IDT, entering kernel mode on Windows 9x, 228-229
“Igor’s problem,” 74
IIS Web servers, W32/Nimda.A@mm worm, 414-415
ImageBase field (PE header), 164
images, scanning, 517
IMP (Core War warrior program), 14
Implant (virus), 264
import address table (IAT), 161, 522
  hooking, 201-203
  page attributes, 574
  patches, 469
import table (PE files), 168-171
import table-replacing (infection technique), 153
imports by ordinal, 240, 469
“in the wild” viruses, 26
in-memory injectors over networks, 215
in-memory residency strategies. See memory residency strategies
InCtrl tool, 637
indirection, layers of, 501
INETINFO.EXE process, 520
INF/Vxer (virus), 96
INF/Zox (virus), 102
infection propagator of worms, 315-316, 331
  backdoor-compromised systems, 331-332
  e-mail attachment inserters, 334
  e-mail attacks, 333-334
INDEX

instant messaging attacks, 333
NNTP attacks, 338
peer-to-peer network attacks, 332-333
SMTP proxy-based attacks, 334-335
SMTP-based attacks, 335-338
infection techniques
Amoeba, 140
appending viruses, 132-133, 174-175
boot viruses, 122-129
cavity viruses, 136-137
classic parasitic viruses, 135-136
code builders, 155-156
companion viruses, 176
compressing viruses, 139-140
embedded decryptor, 141-142
embedded decryptor and virus body, 142-143
every-point obscuring viruses, 145-155
first-generation Windows 95 viruses, 172-173
fractionated cavity viruses, 137-139, 177
header infection viruses, 173
KERNEL32.DLL infection, 175-176
Ifanew field modification, 178
obfuscated tricky jump, 143-144
overwriting viruses, 130-131
PE (portable executable) file format, 160-172, 235
prepending viruses, 133-135, 174
random overwriting viruses, 131-132
system loader comparison between Windows 95 and Windows NT, 181-183
VxD-based viruses, 178-180
W32/Simile virus, 284-285
W95/Zmist virus, 278-280
Win32 viruses, growth of, 181
infections
goat files, 639
natural testing, 637-638
<infective_length> (computer virus naming conventions), 41
Infis (virus). See [W2K, WNT]/Infis (virus)
information query class, 11, 527
INI file viruses, 97
initialization, W95/Zmist virus, 278
injected code detection, 557
shellcode blocking, 558-562
injectors
definition of, 34
in-memory injectors over networks, 215
input validation attacks, 385
MIME types, 387-388, 414-415
URL encoding, 385-386
installation script viruses, 96
installing
dedicated virus analysis systems, 612-615
memory-resident viruses under DOS, 196-198
instant messaging viruses, 83, 333
Instant Virus Production Kit (IVP), 292
instruction tracing (infection technique), 153
INT 13h (interrupt handler), hooking, 188, 191-193
INT 21h (interrupt handler), hooking with file viruses, 193-196
integrity checker programs, 19
Intel, sysenter, 525
Intel Architecture Software Manuals, 615
intended debugger-dependent viruses, 108
intended viruses, 20
interactions between viruses, 354
competition, 357-358
cooperation, 354-357
sexual reproduction, 359
SWCP (simple worm communication protocol), 359
interactive disassembler (IDA), 428
intercept mode, 587
Internal (virus writer)
HTML viruses, 98
installation script viruses, 96
Internet control message protocol (ICMP), 643
Internet Explorer, MIME types, 387-388
Internet Relay Chat (IRC) worms, 83, 333
interpreted environment dependency, 66-98
interrupt handlers, memory scanning for, 218
Interrupt Request Packets (IRPs), 529
Interrupt Spy tool, 392, 647
interrupt vector table (IVT), 188-189, 227
interrupts
calling with INT 1 and INT 3 228
divide-by-zero exceptions, 229
entering kernel mode on Windows 9x, 228-229
generating exceptions, 229
hooking, 188-196, 226
in polymorphic decryptors, 246
undocumented DOS interrupts
(Int 21h/52h), 498
intrusion. See NIDS
Invader (virus), 26
invalidation, exception frame pointers, 568
IP addresses, scanning, 326-330
IRC (Internet Relay Chat) worms, 83, 333
IRPs (Interrupt Request Packets), 529
IsDebuggerPresent( ) API, 229
ISO images, infecting, 59
IVP (Instant Virus Production Kit), 292
IVT (interrupt vector table), 188-189, 227
J
jacky (virus writer), 85
Jacky Qwerty (virus writer), 27
W32/Cabanas virus, 157
W32/Redemption virus, 139
JellyScript, WebTV worms, 86-87
Jerusalem (virus), 136, 197, 497
Jiskefet. See OS2/Jiskefet (virus)
JIT dependency, 99-100
joke programs, definition of, 37
JPEG files, W32/Perrun virus, 116
JS/Kak (virus), 417
JS/Spida (worm), remote login-based attacks, 341
JScript viruses, 85
Junkie (virus), 115
K
Kaspersky, Eugene, 242, 349, 437-438, 447-448, 451
KAV (antivirus program), 438, 442
Kefi (virus writer), PHP/Feast virus, 88
Kelsey, John, 347
kernel mode
driver debuggers, 648
drivers, 503
entering on Windows 9x, 228-229
extensions, buffer overflow attacks (worms), 554-556
viruses in, 212-215
kernel modification, W32/Bolzano virus, 415-417
KERNEL32.DLL
checksum recalculation, 239
hard-coded pointers to, 470
imports, 469-470
inconsistency, 471
infection of, 175-176
kernels, memory scanning, 523
64-bit platforms, 530-531
classes of context, 526
filter driver virus deactivation, 527-529
read-only memory, 529
upper 2G of address space, 527
user address space of processes, 523
Windows NT functions, 525
Windows NT service API entry points, 524
key functions, reconfiguring, 90-91
keyboard, disabling, 231-232
keyloggers, definition of, 36
Khafer, Masouf, 264
Kinematic Self-Replicating Machines (Freitas and Merkle), 7
kits, definition of, 34
KiUserExceptionDispatcher( ) function, 566
knowledge bases, malicious code analysis techniques, 615-616
known plain-text attacks, 449
INDEX

KOH (virus), 304
Krishna (virus), infection technique, 129
Krukov, Andrew, 75

L
L0phtCrack (password cracking program), 326
LADS (tool), 637
Langton, Christopher G., 6
language dependency of macro viruses, 71-72
large scale damage due to worms, 577
layers of indirection, 501
LE (linear executable) file format, 160
Leapfrog (virus), infection technique, 144
Lehigh (virus), 137, 198
Leitold, Ferenc, 662
Lexotan engine, 463
lfanew field modification (infection technique), 178
LFM (virus), 91
LIB viruses, 66
libraries
direct function invocation detection, 571-573
return-toLIBC attacks, 569-573
Libsafe (subsystem extension), 554
Life (game), 8-12
life-cycle manager of worms, 316-317
linear executable (LE) file format, 160
linker dependency, 108-109
linking DLLs to executables, 168-171
links to Web sites or proxies, 339-340
Linux, ELF viruses, 64
Linux/ADM (worm)
detailed description of, 397-398
shellcode blocking, 558
Linux/Cheese (worm), 315, 318
Linux/Jac.8759 (virus), 64
Linux/Lion (antivirus), 318
Linux/Slapper (worm), 64, 98, 108, 315, 538, 543, 647
blocking buffer overflow attacks, 548-549
capturing, 600-602
detailed description of, 401-407
DoS attack, 308
e-mail address harvesting, 323
GOT and IAT page attributes, 574
heap overflows, 376
peer-to-peer network control, 352-354
predefined class table for network scanning, 326-329
shellcode blocking, 558
shellcode-based attacks, 344
worm blocking techniques, 557
Liston, Tom, 596
lists, router access, 585-587
Litchfield, David, 408, 559
LMF (lunar manufacturing facility), 7
LNK viruses, 94
loaded DLLs, disinfecting, 523
LoadLibrary() function, 645
<typename specification> (computer virus naming conventions), 42
logging module, 592
logic bombs, definition of, 30
Logic worm, 83-85
Logo language, Super Logo viruses, 83-85
logs, File Monitor, 635
long loops, 247
Lorez. See W95/Lorez (virus)
Lotus 1-2-3 macro viruses, 96
Lotus Word Pro viruses, 94
LoveLetter. See VBS/LoveLetter.A@mm (worm)
low-interaction honeypot systems, 593
Lucifer (virus), infection technique, 128
Ludwig, Mark, 304
lunar manufacturing facility (LMF), 7
LWP/Spenty (virus), 94
LX viruses, 60-61

M
Ma, Albert, 13
MAC OS X shell scripts, 81
Machine field (PE header), 163
Macintosh platform
  MAC OS X shell scripts, 81
  resource-dependent viruses, 104-105
Macro Identification and Resemblance
  Analyzer (MIRA), 620
macro viruses, 66-69, 157
  corruption, 69-71
  evolution and devolution, 74-75
  formula macros, 77
  infecting user macros, 77
  language dependency, 71-72
  Lotus 1-2-3, 96
  Lotus Word Pro, 94
multipartite infection strategy, 76
  naming conventions, 41
  platform dependency, 73-74
  source code, p-code, execode, 75-76
  up-conversion and down-conversion, 71
  XML, 77
Magic field (PE header), 164
Magistr (virus). See W32/Magistr (virus)
mailers
  definition of, 29
  naming conventions, 42
maintenance, virus collection, 661
malicious code analysis techniques, 612. See also computer viruses
  architecture guides, 615
  collection maintenance, 661
  dedicated system installation, 612-615
  Digital Immune System, 661-664
  disassemblers, 626-632
  dynamic analysis techniques, 634-655
  knowledge bases, 615-616
  process of, 618-624
  unpacking, 625
  Virus Analysis Toolkit (VAT), 656, 659
  VMware, 616-617
malloc() function, 647
malware. See computer viruses
<malware_type>:// (computer virus naming conventions), 40
management
  memory, 498-499
  Virtual Memory Manager, 503
MapInfo viruses, 88-89
MARS (Memory Array Redcode Simulator), 12
Martin, Edwin, 9
Marx, Andreas, 672
mass-mailer worms (@mm worms)
  definition of, 29
  naming conventions, 42
matching patterns, 628
mathematical model for computer viruses, 18
MBR (master boot record), 122, 301
  infection techniques, 124-126
McAfee SCAN (antivirus program), 248
MCB (memory control block), 197-198
MDEF viruses, 105
Memorial. See W95/Memorial (virus)
memory
  buffer overflow attacks. See buffer overflow attacks
  dirty memory pages, 455
  dynamically allocated memory. See heaps
  management, 499
  read-only kernel, 529
  video memory, checking, 232
  VMM memory area, 471
Memory Array Redcode Simulator (MARS), 12
memory control block (MCB), 197-198
Memory Manager, paging, 515-517
memory residency strategies. See also memory-resident viruses
  direct-action viruses, 186
  in-memory injectors over networks, 215
  kernel mode, viruses in, 212-215
  processes, viruses in, 211-212
  swapping viruses, 211
  temporary memory-resident viruses, 210-211
memory scanning, 497-498
  attacks, 532-533
  detecting debuggers, 230
disinfection, 517-523
for interrupt handler, 218
in kernel mode. See kernels, memory
scanning
paging, 515-517
in user mode. See user mode, memory
scanning
Windows NT virtual memory system,
499-505
memory-resident viruses, 186-187
disk cache and system buffer viruses,
209-210
installation under DOS, 196-198
interrupt hooking, 188-196
self-detection techniques, 198-199
stealth viruses, 199-209
Mental Driller (virus writer), 27
W32/Simile virus, 281
W32/Simile.D virus, 53
W95/Drill virus, 224
Merkle, Ralph C., 7
Merry Xmas (virus), 92
metamorphic virus detection, 461
code emulation, 463-466
disassembling techniques, 462-463
geometric detection, 461-462
metamorphic viruses, 20, 269-270
complex permutation techniques,
273-275
host application mutation, 276-277
MSIL metamorphic viruses, 286-288
simple permutation techniques, 270-272
W32/Simile virus, 281-286
W95/Zmist virus, 277-281
metamorphic worms, 576-577
MetaPHOR (virus engine), 281
MICE (Core War warrior program), 13
Michelangelo (virus), 301
Microsoft .NET. See .NET
Microsoft IIS servers, W32/Nimda.A@mm
worm, 414-415
Microsoft Security Bulletin MS03-007, 545
Microsoft SQL Server 2000
exploits, blocking, 559-560
W32/Slammer worm, 407
Microsoft Visual .NET 2003 (7.0 & 7.1),
549-552
Microsoft Xbox, security vulnerabilities, 347
MIME types, 387-388
W32/Badtrans.B@mm worm, 414
W32/Nimda.A@mm worm, 414-415
MIRA (Macro Identification and
Resemblance Analyzer), 620
mIRC, instant messaging viruses, 83
mismatches, first-generation antivirus
scanners, 432
Mistfall engine, 278
mitigation, return-to-LIBC attacks, 569-573
mixed techniques. See blended attacks
MMX instructions, 243
mobile phones, worms on, 359-361
modeling virus infections, 11-12
mathematical model, 18
modification to files (tracking), 635-637
<modifiers> (computer virus naming
conventions), 41
modules
altering, 592
logging, 592
Mole virus. See W32/IKX (virus)
monitoring
files, 635-637
malicious code, 634-655
ports, 641
processes, 641
registries, 640
threads, 641
Monxla (virus), 211
Morris (worm), 32, 315, 318, 538, 543
avoiding buffer overflow attacks, 413, 547
copycat Linux/ADM worm, 397-398
detailed description of, 395-397
history of blended attacks, 367-368
shellcode blocking, 558
weak passwords, 324
Morris, Robert, Sr. (Core War), 12
Mosquitos game, logic bomb in, 30
MPB/Kynel (virus), 88
Mr. Sandman (virus writer), 349
Anti-AVP virus, 248
MSAV (antivirus program), 247
MSIL metamorphic viruses, 286-288
MSIL/Gastropod (virus), 99
self-protection technique, 269, 286-288
MSIL/Impanate (virus), 100, 288
MtE (mutation engine), 262-264
static decryptor detection, 446
multipartite infection strategy, macro
viruses, 76
multipartite viruses, 115-116
multiple PE headers, 469
multiple virus sections, 235-236
multiple-fork support (NTFS), 58
multithreaded viruses, 246
Murkyy (virus writer), 27, 242
infection technique, 138
mutation engine (MtE), 262-264
static decryptor detection, 446
mutation. See corruption
Muttik, Igor, 74-75
metamorphic viruses, 269
MX queries and SMTP-based worm
attacks, 338
Mydoom (virus). See W32/Mydoom (worm)
Myname. See OS2/Myname (virus)

naming conventions
computer viruses, 38-39
@m, 42
@mm, 42
[<devolution=>], 41
<fAMILY_NAME>, 40
<GROUP_NAME>, 41
<infective_length>, 41
<locale_specifier>, 42
<malware_type>://, 40
<modifiers>, 41
#<packer>, 42
<platform>://, 40-46
<variant>, 41
<!vendor-specific_comment>, 42
native viruses, 63-64
native Windows NT viruses, 496, 512
natural infection testing, 637-638
natural infections, 600
nature-simulation games, 5
Core War, 12-16
Edward Fredkin structures, 7-8
game of Life (Conway), 8-12
John von Neumann theory, 5-7
Navrhar (virus). See W95/Navrhar (virus)
NC (NetCat) tool, 593, 642
NCAs (Nexus Agents), 534
NE viruses, 60
nearly-exact identification, 437-438
NEAT (WebTV worm), 86
Neat (worm), 911 attacks, 308
Nebbett, Gary, 616
Needham, Roger, 346
.NET
JIT-dependent viruses, 99-100
W32/Donut virus, 143-145
NET$DOS.SYS file, boot viruses in, 129
NetCat (NC) tool, 593, 642
network enumeration attacks, 393-394
network injectors, definition of, 34
network intrusion detection system (NIDS), 584, 591-592
network protocol dependency, 102
network scanning, 326-330
network share enumeration attacks, 324-326
network-level defense strategies, 584
counterattacks, 596
eyearly warning systems, 598
firewalls, 588-589
honeypot systems, 593-594
INDEX

network intrusion detection system (NIDS), 584, 591-592
router access lists, 585-587
worm behavior patterns, 598-608

networks
boot viruses, 129
in-memory injectors over networks, 215
peer-to-peer network attacks, 332-333, 352-354
ports, monitoring, 641
traffic, capturing, 643
neural networks, heuristic analysis using, 472-474
Nexiv_Der (virus), 146-147, 153
Nexus Agents (NCAs), 534
NGSCB (Next Generation Secure Computing Base), 534
NGVCK (Next Generation Virus Creation Kit), 291
NIDS (network intrusion detection system), 584, 591-592
Nimda. See W32/Nimda (worm)
NNTP attacks, worm infections, 338
NNTP-based e-mail address collection, 320-321

no-payload viruses, 296-297
NoKernel (virus), 219
non- TSR viruses, 497
nondestructive payload viruses, 297-300
nonexecutable (NX) pages, 534, 579
nonlinear decryption, 256
nonstateful firewalls, 588
normal COM, definition of, 132
Norton AntiVirus (antivirus program), 442
Norton, Peter (Programmer’s Guide to the IBM PC), 25
NOTEPAD.EXE
STR streams, 636
W32/Parvo (virus) inside, 511
Novell NetWare ExecuteOnly attribute, attacks via, 389-393
Nowhere Man (virus construction kit writer), 289
NTDLL functions, 524
NTFS file systems
compression viruses, 59
stream viruses, 58-59
NtOpenThread( ) function, 519
NtQueryInformationThread( ) API, 519
NtQuerySystemInformation( ) (NtQSI), 506-507
NtQueryVirtualMemory( ) API, 524
NumberOfSections field (PE header), 164
Number_Of_The_Beast (virus), 193, 207
NX (nonexecutable) pages, 534, 579

0
obfuscated code, 222-224
obfuscated entry points, 233
obfuscated file formats, 233
obfuscated tricky jump (infection technique), 143-144
object code viruses, 66
Object Manager functions, 527
objects (network enumeration), 394
octopus (worm), definition of, 29
off-by-one buffer overflows, 371-373
OLE2 files, macro viruses, 67-68
oligomorphic viruses, 259-260
Olivia (virus), infection technique, 145-146
OllyDBG tool, 648
Omul (virus), infection technique, 132
on-access antivirus scanners, 426. See also scanners
on-demand antivirus scanners, 426. See also scanners
One_Half (virus), 277, 304
infection technique, 141
opcode mixing-based code confusion, 223-224
OpenSSL, vulnerabilities in, 401
OpenThread( ) function, 519
operating system dependency, 55
INDEX

operating system version dependency, 55-56
operating systems, buffer overflow attacks
  (worms), 552-554. See also names of specific
  operating systems
ordinal-based imports, 240, 469
original boot sector, 128-129
OS/2
  LX viruses, 60-61
  NE viruses, 60
OS2/Jiskefet (virus), 61
OS2/Myname (virus), 60
outbreak statistics (worm), 670
outgoing e-mail messages, harvesting e-mail
  addresses using, 322-323
overflows. See buffer overflow attacks
Overmars, Mark, 113
overwriting viruses (infection technique),
  130-131, 301-302

P
p-code, macro viruses, 75-76
packed code sections, 237
#<packer> (computer virus naming
  conventions), 42
packers. See compression
packets, decoders, 591
PAE (Physical Address Extension), 500
page directories (memory), 500
page directory entries (PDEs), 500
page frames (memory), 500
page table entry (PTE), 555
page tables (memory), 500
PAGE_READONLY access, 522
paging, memory scanning and, 515-517
Palm platform, resource-dependent
  viruses, 105
Palm/Phage (virus), 105
parasitic viruses. See classic parasitic viruses
  (infection technique)
parsing files for e-mail addresses, 319-320
partition table (PT) entries, 122
changing, 125-126
partitions, definition of, 122
password cracking, Morris worm, 367
password handling, vulnerabilities, 324
password protection, 249
password-capturing attacks, 325
  definition of, 32
passwords, security problems, 324-326
Pasteur (antivirus program), 26, 436
patching
  code in active pages, 522
  import address table (IAT), 469
Pathogen (virus), X-RAY scanning, 448
patterns
  of computer viruses, 630
  matching, 628
  worm behavior, 598-608
PaX (kernel mode extension), 554-556
payload activation
  accidentally destructive payload
    viruses, 297
  highly destructive payload viruses,
    301-306
  no-payload viruses, 296-297
  nondestructive payload viruses, 297-300
  somewhat destructive payload viruses,
    300-301
  types of, 296
  W32/Simile virus, 285-286
  of worms, 318
PDEs (page directory entries), 500
PDF viruses, 90
PDF/Yourde (virus), 90
PE (portable executable) file format,
  158-160, 513
  entry points, 468
  infection by W95/Zmist virus, 279-280
  infection techniques, 160-172, 235
  Windows CE, 110
PE header
  avoiding infection, 240
  code section sizes, 241
  infection, 469
<table>
<thead>
<tr>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>multiple headers, 469</td>
</tr>
<tr>
<td>SizeOfCode field, 471</td>
</tr>
<tr>
<td>virtual size, 468</td>
</tr>
<tr>
<td>PE viruses, 61-64</td>
</tr>
<tr>
<td>PEDUMP, 622, 645</td>
</tr>
<tr>
<td>PeElf (virus). See [W32, Linux]/Peelf (virus)</td>
</tr>
<tr>
<td>peer-to-peer network attacks, worm infections, 332-333, 352-354</td>
</tr>
<tr>
<td>Linux/Slapper worm, 406-407</td>
</tr>
<tr>
<td>PEID tools, 626</td>
</tr>
<tr>
<td>Pentium II processors, sysenter, 525</td>
</tr>
<tr>
<td>Perl viruses, 86</td>
</tr>
<tr>
<td>permutation</td>
</tr>
<tr>
<td>complex permutation techniques</td>
</tr>
<tr>
<td>(metamorphic viruses), 273-275</td>
</tr>
<tr>
<td>simple permutation techniques</td>
</tr>
<tr>
<td>(metamorphic viruses), 270-272</td>
</tr>
<tr>
<td>W95/Zmist virus, 279</td>
</tr>
<tr>
<td>Perriot, Frederic, 282, 317, 647</td>
</tr>
<tr>
<td>personal firewalls. See firewalls</td>
</tr>
<tr>
<td>Phager (virus), 101</td>
</tr>
<tr>
<td>Phalcon-Skism Mass Produced Code Generator (PS-MPC), 290</td>
</tr>
<tr>
<td>phishing attacks, 308-309</td>
</tr>
<tr>
<td>definition of, 35</td>
</tr>
<tr>
<td>phones, wireless mobile worms, 359-361</td>
</tr>
<tr>
<td>PHP viruses, 88</td>
</tr>
<tr>
<td>PHP/Caracula (virus), 88</td>
</tr>
<tr>
<td>PHP/Feast (virus), 88</td>
</tr>
<tr>
<td>Physical Address Extension (PAE), 500</td>
</tr>
<tr>
<td>Pietrek, Matt, 616</td>
</tr>
<tr>
<td>PIF viruses, 94</td>
</tr>
<tr>
<td>Pile, Christopher (virus writer), 448</td>
</tr>
<tr>
<td>Ping Pong (virus), 54</td>
</tr>
<tr>
<td>pings, W32/Welchia (worm), 605</td>
</tr>
<tr>
<td>&lt;platform&gt;/ (computer virus naming conventions), 40</td>
</tr>
<tr>
<td>list of officially recognized names, 42-46</td>
</tr>
<tr>
<td>platform dependency of macro viruses, 73-74</td>
</tr>
<tr>
<td>platform support for Win32, 158-160</td>
</tr>
<tr>
<td>Playgame (virus), nondestructive payload</td>
</tr>
<tr>
<td>viruses, 299</td>
</tr>
<tr>
<td>Ply (virus), self-protection technique, 253</td>
</tr>
<tr>
<td>Pobresito (virus), 92</td>
</tr>
<tr>
<td>Polimer.512.A (virus), 134</td>
</tr>
<tr>
<td>polymorphic decryptors</td>
</tr>
<tr>
<td>interrupts in, 246</td>
</tr>
<tr>
<td>W32/Simile virus, 282-283</td>
</tr>
<tr>
<td>polymorphic viruses, 261</td>
</tr>
<tr>
<td>32-bit polymorphic viruses, 264-268</td>
</tr>
<tr>
<td>1260 virus, 261-262</td>
</tr>
<tr>
<td>macro viruses, 76</td>
</tr>
<tr>
<td>MtE (mutation engine), 262-264</td>
</tr>
<tr>
<td>PHP viruses, 88</td>
</tr>
<tr>
<td>polymorphic worms, 576-577</td>
</tr>
<tr>
<td>polymorphism, virus detection, 455-458</td>
</tr>
<tr>
<td>Popp, Joseph, 31</td>
</tr>
<tr>
<td>port 80 (HTTP), NetCat, 594</td>
</tr>
<tr>
<td>port I/O, disk access, 219</td>
</tr>
<tr>
<td>portable executable. See PE (portable executable) file format</td>
</tr>
<tr>
<td>ports, monitoring, 641</td>
</tr>
<tr>
<td>PPE (Prizzy polymorphic engine), 243</td>
</tr>
<tr>
<td>predefined class table (network scanning), 326-329</td>
</tr>
<tr>
<td>prefetch-queue attacks, 230-231</td>
</tr>
<tr>
<td>prepending viruses (infection technique), 133-135, 174, 236</td>
</tr>
<tr>
<td>preprocessors, network intrusion detection system (NIDS), 591</td>
</tr>
<tr>
<td>printers, targeted by worms, 324</td>
</tr>
<tr>
<td>private pages, Win32 viruses that allocate, 310</td>
</tr>
<tr>
<td>Prizzy (virus writer), W32/Crypto virus, 257</td>
</tr>
<tr>
<td>Prizzy polymorphic engine (PPE), 243</td>
</tr>
<tr>
<td>process address space randomization, 570</td>
</tr>
<tr>
<td>processes</td>
</tr>
<tr>
<td>computer virus analysis, 618-624</td>
</tr>
<tr>
<td>context (memory scanning), 526</td>
</tr>
<tr>
<td>enumerating, 517</td>
</tr>
<tr>
<td>memory scanning, 507-508</td>
</tr>
<tr>
<td>monitoring, 641</td>
</tr>
<tr>
<td>terminating, 518</td>
</tr>
</tbody>
</table>
INDEX

user address space of (scanning), 523
viruses in, 211-212
PROCESS_TERMINATE access, 518
PROCESS_VM_OPERATION access, 522
profiles, tracking decryptors, 454
program shepherding, buffer overflow attacks (worms), 556
Programmer’s Guide to the IBM PC (Norton), 25
propagation (worms). See code propagation techniques (worms)
ProPolice, 548-549
Provos, Niels, 595
proxy firewalls, 588
PS-MPC (Phalcon-Skism Mass-Produced Code Generator), 290
PSD (virus), 621
pseudo-decryption loops, 460
PSMPC generators, 34
PT (partition table) entries, 122
changing, 125-126
PTE (page table entry), 555
Python viruses, 87

Q
Q the misanthrope (virus writer)
BAT/Ramble virus dropper, 96
GoldBug virus, 198
memory allocation techniques, 198
Qark (virus writer), 306
QAZ (virus), 309
Qpa (virus), infection technique, 136
Quantum (virus writer), 27, 61
Queeg (virus), X-RAY scanning, 448-450
quick examinations, process of computer virus analysis, 619

R
rabbit (worm), definition of, 29
Radai, Yisrael, 302
Raiu, Costin, 75
Rajaat (virus writer), 78
Ralf Brown Interrupt List, 190
Ramble (virus), 96
Ramdhani, Denny Yanuar (virus writer), 127
Ramen (worm), 315
random decryption algorithm (RDA) viruses, 237, 245, 256
random entry points in code section, 237-238
random execution logic, 244-245
random overwriting viruses (infection technique), 131-132
randomization, process address space, 570
randomized network scanning, 329-330
Raptor (firewall), 590
Ratter (virus writer) W32/Kick virus, 65
WinCE/Duts.1520 virus, 109
RDA (random decryption algorithm) viruses, 237, 245, 256
RDA.Fighter (virus), 256
RDTSC instruction, 283
read stealth viruses, 203-205
read-only kernel memory, 529
ReadProcessMemory( ) API, 505-506
real permutating engine (RPME), 274
Reaper (antivirus program), 17
recalculating checksum, 239
reconfiguring key functions, 90-91
recycling compiler alignment areas, 238
Redcode language, 12-15
refiltering drivers (DeactivatorDriver), 529
registries, monitoring, 640
Registry keys
detecting debuggers, 229
macro viruses, 74
Registry-dependent viruses, 93-94
Regmon tool, 640
regular disinfection methods, 474-477
relative virtual address (RVA), 161
.reloc section (PE files), 167
relocation cavity viruses (infection technique), 137
remote control of worms, 316, 351-352
peer-to-peer network control, 352-354
remote login-based attacks, 341
RemoteExplorer virus. See WNT/RemEx (virus)
renaming sections, 239
replication. See self-replicating systems;
worm blocking techniques
Repus. See W95/Repus (virus)
requests
Address Resolution Protocol (ARP), 595
pings, capturing W32/Welchia (worm), 605
research honeypots, 596
research papers (virus), 670
resident viruses. See memory-resident viruses
resource dependency, 104-105
resources, early warning/up-to-date security
information, 669
retroviruses, 11, 247-249, 300
retroworms, 576
return-to-LIBC attacks, 543, 569-573
reviving dead virus code, 127
REXX viruses, 78-79
Riordan, Roger, 433
Ripper (virus), 303
Ritchie, Dennis (Core War), 12
rootkits, definition of, 36
routers, access lists, 585-587
Rowe, Mark, 360
roy g biv (virus writer), 27
Ginger virus, 198
MSIL/Impanate virus, 100, 288
W32/Chiton virus, 63, 154
W64/Rugrat.3344 virus, 62
RPME (real permutating engine), 274
\.rsrc section (PE files), 167
RT Fishel (virus writer), Ginger virus, 198
RTL (run-time library) functions, 545
Rugrat. See W64/Rugrat.3344 (virus)
run-time code injection attacks. See code
injection attacks
run-time library (RTL) functions, 545
run-time packers, 625
Russel, Ryan, 594
RVA (relative virtual address), 161
Sadmind (worm), 315
safe-for-scripting ActiveX controls, 388-389
VBS/BubbleBoy worm, 417-418
W32/Blebla worm, 418-419
Sandman (virus writer), 27, 299
SAP, ABAP viruses, 89
saving
files locally, W32/Blebla worm, 418-419
original boot sector at end of disk, 128-129
SC Magazine, 672
scanners, 252
algorithmic scanning methods, 441-443
filtering, 443-444
static decryptor detection, 444-446
X-RAY method, 446-451
code emulation, 451-454
dynamic decryptor detection, 459-461
encrypted/polymorphic virus detection, 455-458
disinfection methods, 474-475
generic decryptors, 477
standard, 475-477
first-generation antivirus, 428
bookmarks, 433-434
entry-point scanning, 435-436
fixed-point scanning, 435-436
generic detection, 432
hashing, 432-433
hyperfast disk access, 436
mismatches, 432
string scanning, 428-430
top-and-tail scanning, 435
wildcards, 430-431
heuristic analysis
of 32-bit Windows viruses, 467-472
using neural networks viruses,
472-474
INDEX

second-generation antivirus, 437
  exact identification, 439-441
  nearly-exact identification, 437-438
  skeleton detection, 437
  smart scanning, 437
scanning
  file images, 517
  IP addresses, 326-330
  memory. See memory scanning
SCANPROC.EXE, 515
Schneier, Bruce, 347
science versus art, 4
script viruses, REXX viruses, 78-79
scripts, blocking, 539-542
search engines, harvesting e-mail addresses using, 321
searching VOOGLE, 621
second-generation antivirus scanners, 437
  exact identification, 439-441
  nearly-exact identification, 437-438
  skeleton detection, 437
  smart scanning, 437
second-generation buffer overflows, 371-378
  definition of, 369
section table (PE files), 165-168
SectionAlignment field (PE header), 165
sections
  code sections
    naming, 469
    sizes in header, 241
    gaps between, 468
  packed code sections, 237
  PE files, 161
  random entry points, 237-238
  renaming, 239
  shifting, 236
  slack area infections, 236
  suspicious characteristics, 468
  writeable flag, 238
sector-level stealth viruses, 207-208
sectors
  formatting extra, 126-128
  marking as BAD, 128
security
  exploits. See blended attacks
  information of, 669
  updates, 669
    buffer overflow attacks (worms), 544-545
security_cookie values, 550
seeding, definition of, 34
SEH (structured exception handling), 243-244, 565
self-contained environment dependency, 113-115
self-detection techniques, memory-resident viruses, 198-199
self-modifying code. See obfuscated code
self-protection techniques (of viruses)
  armored viruses. See armored viruses
  encrypted viruses, 253-258
  metamorphic viruses. See metamorphic viruses
  oligomorphic viruses, 259-260
  polymorphic viruses, 261-268
  retrorviruses, 247-249
  tunneling viruses, 218-220
  virus construction kits, 288-293
self-replicating systems, history of, 4
Core War, 12-16
  Edward Fredkin structures, 7-8
  game of Life (Conway), 8-12
  John von Neumann theory, 5-7
self-sending code blocking, 563-565
self-tracking of worms, 318
semistealth viruses, 200-203
sending, self-sending code blocking, 563-565
sendmail, Morris worm, 367
service viruses, native Windows NT, 512
SETI, use by computer worms, 318
sexual reproduction of viruses, 359
SH/Renepo.A (worm), 81
shape heuristic, 461
share-level password vulnerability, 324
sharepoints (network enumeration), 394
shel shell scripts, 80-81
code, blocking, 558-562
code-based attacks, 342-344, 543
Shifter (virus), 66
shifting sections, 236
Shockwave Rider (Brunner), 29
“Shooter” starting structure (game of Life), 9-10
Short Message Service (SMS), 30
Sieben, Na’ndor, 13
signatures, 608
flirt, 628
Simile virus. See [W32, Linux]/Simile (virus)
simple worm communication protocol (SWCP), 359
Simulated “Metamorphic” Encryption Generator (SMEG), 448
simulations of nature. See nature-simulation games
single-layer classifiers with thresholds, 473
single-stepping, detecting, 227
Sircam (worm)
    e-mail address harvesting, 320
    SMTP-based attacks, 335
SizeOfCode field (PE header), 164, 471
SizeOfImage field (PE header), 165, 468
skeleton detection, 437
Skrenta, Rich (Elk Cloner virus), 17
Skulason, Fridrik, 39, 115, 438
slack area infections, 236
Slammer (worm). See W32/Slammer (worm)
Slapper (worm). See Linux/Slapper (worm)
Sma. See W95/Sma (virus)
smart scanning, 437
SMEG (Simulated “Metamorphic” Encryption Generator), 448-450
SMS (Short Message Service), 30
SMTP, blocking, 539-542
SMTP proxy-based attacks, worm infections, 334-335
SMTP spam relay, use by computer worms, 318
SMTP-based attacks, worm infections, 335-338, 643
SnakeByte (virus writer)
    NGVCK (virus construction kit), 291
Perl viruses, 86
sniffing traffic, 643
SoftIce Debugger (antivirus program), 527
SoftICE tool, 648
Solaris on SPARC, 553-554
Solaris/Sadmind (virus), 98, 543
Solomon, Alan, 37, 39, 200, 293
somewhat destructive payload viruses, 300-301
source code, macro viruses, 75-76
source code dependency, 102-104
source spoofing, 587
Sourcer (disassembler), 221
SP (stack pointer), decryption with, 230
spammer programs, definition of, 35
Spanska (virus writer), 27, 350
    Happy99 worm, 62
    IDEA viruses, 256, 299
    self-protection technique, 245, 248
spoofing source, 587
spyware, definition of, 38
SQL Server 2000, W32/Slammer worm, 407
ssnetlib.dll, W32/Slammer worm, 408
stack buffer overflows, 369-370
    causes of, 371
    CodeRed worm, 398-401
    exploiting, 370
    Linux/ADM worm, 397-398
    Morris worm, 395-397
    W32/Blaster worm, 410-413
    W32/Slammer worm, 407-410
stack pointer (SP), decryption with, 230
stack smashing, 546
INDEX

stack state, checking, 227
stack-based overflow attacks, compiler-level solutions, 546
StackGuard, 546-548
stacks
  definition of, 91
  exception-handler validation, 568
  return-to-LIBC attacks, 569-573
standard access lists, 586
standard disinfection, 475-477
Starship (virus), 126, 198
stateful firewall solutions, 588
static decryptor detection, algorithmic scanning methods, 444-446
static heuristics, 234
stealing data. See data stealing viruses
stealth viruses, 199-200
  cluster and sector-level stealth viruses, 207-208
  full-stealth viruses, 205-206
  hardware-level stealth viruses, 208-209
  read stealth viruses, 203-205
  semistealth viruses, 200-203
Stoll, Clifford, 593
Stoned (virus), 24-25
  accidentally destructive payload viruses, 297
  bookmarks, 433
  exact identification, 439-440
  infection technique, 124-126
  interrupt hooking, 192-193
  nearly exact identification, 437
  string scanning, 429-430
stopping break points, 454
Stormbringer (virus writer), Shifter virus, 66
Strack, Stefan, 13
Strange (virus), 208
stream viruses, file system dependency, 58-59
Strike (virus), infection technique, 128
string scanning, 428-430
strings
  API strings, 241-242
dumps, 623-624
  mismatches, first-generation antivirus scanners, 432
  wildcards, first-generation antivirus scanners, 430-431
structured exception handling (SEH), 243-244, 565
structures, self-replicating, 7-8
Struss, J. (virus construction kit writer), 289
Stupid (virus), 196
submissions, worm-blocking, 541
subsystems
  extensions, buffer overflow attacks (worms), 554
  Win32 viruses, 508-511
super fast infectors, 56
Super Logo viruses, 83-85
Suslikov, Eugene, 633
swapping viruses, 211
SWCP (simple worm communication protocol), 359
Symantec Security Response, 540
Symboot, 619
SymbOS/Cabir (worm), 359-361
sysenter, 525
system buffer viruses, 209-210
system call tracing, 647-648
System File Checker feature (Windows 2000/XP), 417
system loader, Windows 95 versus Windows NT, 181-183
system modification attacks, 389
  Novell NetWare ExecuteOnly attribute, 389-393
  W32/Bolzano virus, 415-417
system rights, memory scanning, 507-508
INDEX

U

UDP-based attacks versus TCP-based attacks, 539
Ulam, Stanislaw, 6
UMB (upper memory block), 198
undocumented CPU instructions, 245
undocumented functions, virus self-protection techniques, 219-220
Unicode strings. See strings
University of Hamburg’s Virus Test Center (VTC), 672
University of Magdeburg, 672
UNIX
   ELF viruses, 64-65
   shell scripts, 80-81
   shellcode blocking, 558-562
unknown entry points (infection technique), 154-155
unpacking, malicious code analysis techniques, 625
up-conversion of macro viruses, 71
update interface of worms, 316, 345-346
   authenticated updates, 346-351
   backdoor-based updates, 351
updates, security, 669
   buffer overflow attacks (worms), 544-545
upper 2G of address space (memory scanning), 527
upper memory block (UMB), 198
UPX (run-time packer), 625
URL encoding, 385-386
user address space of processes, scanning, 523
user macros, infecting, 77
user mode
   debuggers, 648
   memory scanning in, 505-506
      executed images (Win32 viruses), 512-514
      hidden window procedure (Win32 viruses), 512
native Windows NT service viruses, 512
NtQuerySystemInformation() (NtQSI), 506-507
   processes/rights, 507-508
   Win32 viruses, 508-511
viruses in processes, 211-212
user mode rootkits, definition of, 31, 36
UTF-8 encoding, 385-386

V

V.T. (virus writer), Darth_Vader virus, 197
V2Px (virus), self-protection technique, 226
Vacsina (virus), 26, 132
Vajda, Ferenc, 11
validation
   application rights verification, 388
   exception-handler, 565-569
   input validation attacks, 385-388, 414-415
ValleZ (virus writer), W32/Zelly virus, 255
vampire attacks, 358
vampire warriors (Core War game), 16
van Wyk, Ken, 137
<variant> (computer virus naming conventions), 41
Varicella (virus), self-protection technique, 248
VAT (Virus Analysis Toolkit), 613, 656-659
VAX/VMS systems, DCL viruses, 79-80
VBA document macros, 112-113
VBS/Bubbleboy (worm)
   detailed description of, 417-418
   HTML-based mail, 340
   safe-for-scripting ActiveX controls, 389
VBS/LoveLetter.A@mm (worm), 29, 81, 314, 538
   infection technique, 130
   script blocking, 539
VBS/VBSWG.J (Anna Kournikova virus), 35.
   See also Anna Kournikova virus
VBScript viruses, 81-82
VCL (Virus Creation Laboratory), 34, 289-290
INDEX

VCL.428 (virus), 186
VCS (Virus Construction Set), 289
Vecna (virus writer), 27
  W32/Borm worm, 332
  W32/Coke virus, 255
  W32/HybrisF virus, 139, 248
  W95/Fabi virus, 107
  W95/Regswap virus, 270
Veldman, Frans, 264, 433, 447
Velvet (virus), self-protection technique, 229
!<vendor-specific_comment> (computer virus naming conventions), 42
vendors, antivirus software (contact information), 670
VET (antivirus program), 433
VGrep, 619
video memory, checking, 232
Vienna (virus), 26, 132, 186, 200
VIM viruses, 87
Virdem (virus), 59, 135, 186
VIRKILL (antivirus program), 436
VIROCRK (decryption tool), 451
virtual address spaces, 501-505
virtual addresses, translation of, 500
virtual debuggers, 649
virtual machine manager (VMM), 179, 471
virtual machines, 451-458, 465
Virtual Memory Manager, 503
virtual memory systems (Windows NT), 499-505
VirtualAlloc() function, 510
VirtualProtectEx() function, 522
VirtualQueryEx() API, 524
VirtualRoot (Trojan horse), 310
Virus Analysis Toolkit (VAT), 656, 659
Virus Bulletin Web site, 672
virus construction kits, 288
  ethics of using, 293
  GenVir, 289
  list of, 291-292
  NGVCK, 291
  PS-MPC, 290
VCL (Virus Creation Laboratory), 34, 289-290
VCS (Virus Construction Set), 289
Virus Construction Set (VCS), 289
Virus Creation Laboratory (VCL), 34, 289-290
virus generators, definition of, 34
Virus Patrol (antivirus service), 320
virus research
  art versus science, 4
  author’s start in, 24-26
  common patterns, 26-27
Virus Research Unit of the University of Tampere in Finland, 673
virus throttling, 575
viruses
  antivirus defense techniques, 426-427
  code evolution, 252-253
  definition of, 18-20, 28
  history of, 17-18
  interactions, 354
    competition, 357-358
    cooperation, 354-357
    sexual reproduction, 359
  SWCP (simple worm communication protocol), 359
  modeling virus infections, 11-12
  naming conventions, 38-39
    [<devolution>], 41
    <family_name>, 40
    .<group_name>, 41
    <infective_length>, 41
    :<locale_specifier>, 42
    <malware_type>://, 40
    <modifiers>, 41
    #<packer>, 42
    <platform>/, 40-46
    <variant>, 41
    @m, 42
    @mm, 42
    !<vendor-specific_comment>, 42
  retro viruses, 11
  terminology, 28-36
  versus worms, 314
INDEX

Visual .NET 2003 (Microsoft), 549-552
VLAD (virus writer), 53
W95/Boza virus, 61
VM. See virtual machines
VMM (virtual machine manager), 179, 471
VMWARE, 613-617, 642
von Neumann, John, 4-7
von Neumann, Nicholas, 5
VOOGLE, 621
VPN (virtual private network). See
network-level defense strategies
VTC (University of Hamburg’s Virus Test
Center), 672
vulnerability dependency, 98. See also
blended attacks
VxD-based viruses (infection technique), 65,
178-180
VxDs, LE (linear executable) file format, 160
Vyssotsky, Victor (Core War), 12

W

W2K/Installer (virus), 137
[W2K, WNT]/Infis (virus), 65, 213-215
W16/Tentacle_II (virus), 60, 147-150
W16/Winvir (virus), 60
W32/Aldebera (virus), 139
W32/Aliz (worm), 337, 643
W32/Aplore (worm), 340
W32/Apparition (virus), 269
W32/Badtrans.B@mm (worm), 414
W32/Beagle (worm), 100
backdoor-based updates, 351
cooperation with viruses, 356
self-protection technique, 249, 258
W32/Beagle.T (worm), 340
W32/Blaster (worm), 315, 98
capturing, 598-600
competition between worms, 358
detailed description of, 410-413
DoS attack, 306-307
exploits, blocking, 561
return-to-LIBC attacks, 571
self-protection technique, 225
shell code-based attacks, 343
W32/Blebla (worm), 418-419
W32/Bofax (worm), 318
W32/Bolzano (virus)
detailed description of, 415-417
system modification attacks, 389
W32/Borm (worm)
backdoor-compromised systems, 331-332
cooperation with viruses, 356
W32/Brid@mm (worm), 539
W32/Bugbear (worm), 311
network share enumeration, 324
SMTP worm blocking, 539
W32/Bymer (worm), 318
W32/Cabanas (virus), 157, 201-203
infection technique, 144, 175, 183
self-protection technique, 232, 243
W32/Cabanas.3014.A (virus), 510
W32/Chiton (virus), 63-64
infection technique, 154
memory scanning attacks, 533
self-protection technique, 256-258
W32/Choke (worm), 333
W32/Cholera (worm), 356
W32/CodeGreen (antiworm), 318, 357-358
W32/CodeRed (worm), 98, 215, 315, 318, 366,
496, 517, 520, 538, 542
avoiding buffer overflow attacks, 413
blocking, 564-565
code injection attacks, 342, 543
competition between worms, 357-358
computer security versus antivirus pro-
grams, 366
detailed description of, 398-401
DoS attack, 307
exception-handler validation, 568
exploits, blocking, 560-561
history of blended attacks, 368
return-to-LIBC attacks, 570
self-sending code blocking, 563
stack buffer overflows, 370
system modification attacks, 389
virus throttling, 575
W32/CodeRed_II (worm), 310, 520
W32/Coke (virus), 76, 255, 266
W32/Crypto (virus), 257, 305
W32/CTX (virus), 628
cooperation with W32/Cholera worm, 356
infection technique, 137, 150
W32/Dabber (worm), 358
W32/Dengue (virus)
dynamic decryptor detection, 459
infection technique, 150
self-protection technique, 241
W32/Donut (virus), 99
infection technique, 143-144
naming, 145
W32/Doomjuice (worm)
backdoor-based updates, 351
cooperation with viruses, 356
W32/Elkern (virus), 532
W32/Evol (virus)
code emulation, 464
self-protection technique, 273
W32/ExploreZip (worm), 538
self-protection technique, 235
SMTP worm blocking, 541
SMTP-based attacks, 335
W32/Franvir (virus), 113-115
W32/Funlove (virus), 416, 427
blocking, 579
cooperation with worms, 356
network enumeration attacks, 324, 394
W32/Gaobot.AJS (worm)
competition between worms, 358
memory scanning attacks, 533
W32/Ghost (virus), 271
W32/Gobi (virus)
filtering, 443
self-protection technique, 247
W32/Harrier (virus), 255
W32/Heathen.12888 (virus), 73
W32/Heretic (virus), 522
W32/Heretic.1986.A (virus), 512-513
W32/HIV (virus), 59
W32/HLLP.Cramb (virus), 236
W32/HLLP.Sharpei (virus), 99
W32/HLLW.Bymer (virus), 394
W32/HLLW.Lovgate@mm (worm), 539
W32/HLLW.Qaz.A (worm), 309
W32/Holar@mm (worm), 539
W32/Hybris (worm), 577
W32/HybrisF (virus)
infestation technique, 139
self-protection technique, 248
W32/Hyd (worm), 318, 334
W32/Idele (virus), 153
W32/IKX (virus), 236, 241
W32/Infynca (virus), 229
W32/Kick (virus), 65
W32/Klez (worm), 538
infestation technique, 136
MIME header exploits, 414
SMTP worm blocking, 539-541
W32/Klez.H (worm), 320
W32/Kriz (virus), 239-240
W32/Leaves (worm), 332
W32/Legacy (virus), 243
{W32, Linux}/Peelf (virus), 52, 286
{W32, Linux}/Simile (virus), 258, 281-286
{W32, Linux}/Simile.D (virus), 53, 64,
256, 576
W32/LesPaul@mm (worm), 342
W32/Lirva@mm (worm), 539
W32/Lovegate@mm (worm), 533
W32/Maax (worm), 333
W32/Magistr (virus)
e-mail address harvesting, 319
heuristics, 466
SMTP-based attacks, 336
W32/Mimail.I@mm (phishing attack), 309
W32/Mydoom (worm)
backdoor-based updates, 351
cooperation with worms, 356
e-mail address harvesting, 320
self-protection technique, 249
INDEX

SMTP-based attacks with MX queries, 338
W32/Mydoom.A@mm (worm), 540
W32/Mydoom.M@mm (worm), 321
W32/Niko.5178 (virus), 513-514
W32/Nimda (worm), 97, 311, 314, 366, 538
backdoor-compromised systems, 332
SMTP worm blocking, 539
SMTP-based attacks, 335
W32/Nimda.A@mm (worm), 29, 414-415
W32/Opaserv (worm), 318
network enumeration attacks, 394
password handling, 324
W32/Parvo (worm), 518
e-mail address harvesting, 321
e-mail worm attacks, 334
W32/Parvo.13857 (virus), 510-511
W32/Perenast (virus)
infection technique, 153
self-protection technique, 237
W32/Perrun (virus), 116
W32/Press (virus), 78
W32/PrettyPark (worm), 93
W32/Qint@mm (worm), 257
W32/RainSong (virus), 152
W32/Redemption (virus), 139
W32/Resure (virus), 235
W32/Sand.12300 (virus), 140
W32/Sasser (worm), 358
W32/Sasser.D (worm), 603
W32/Semisoft (virus), 518
W32/Serot (worm), 319
W32/SKA (worm), 299, 314, 538. See also Happy99 worm
W32/SKA.A (worm), 29, 62, 522
W32/Slammer (worm), 215, 316, 496, 538-539, 542
blocking, 564
capturing, 607-608
code injection attacks, 341
detailed description of, 407-410
DoS attack, 306
randomized network scanning, 329-330
self-sending code blocking, 563
virus throttling, 575
worm blocking techniques, 557
W32/Smorph (Trojan), 277
W32/Sobig (worm)
e-mail address harvesting, 321
SMTP worm blocking, 539
W32/Subit (virus), 102-103
W32/Taripox@mm (worm), 334
W32/Tendoolf (worm), 351
W32/Thorin (virus), 243
W32/Toal@mm (worm), 322
{W32, W97M}/Beast.41472.A (virus), 112, 512
W32/Wangy (worm), 324
W32/Welchia (worm), 98
backdoor-based updates, 351
capturing, 605
competition between worms, 358
exploits, blocking, 562
network scanning and fingerprinting, 330
shell code-based attacks, 344
W32/Welchia.A (worm), 316-317
W32/Witty (worm), 34, 302, 316
large-scale damage, 578
self-sending code blocking, 565
W32/Yaha@mm (worm), 539
W32/Yourde (virus), 90
W32/Zelly (virus)
infection technique, 175
self-protection technique, 255
W64/Rugrat.3344 (virus), 62, 580
W64/Shruggle (virus), 62
W95/Aldabera (virus), 237
W95/Anxiety (virus), 166, 174, 179
W95/Babylonia (worm), 345-346, 349
W95/Bistro (virus), 275
W95/Boza (virus), 55, 61, 157, 166, 171, 174
heuristic analysis, 468
infection technique, 182
W95/Boza.A (virus), 172-173
W95/Cerebrus (virus), 178
INDEX

Windows 95
appending viruses, 174-175
boot viruses, 129
companion viruses, 176
first-generation viruses, 172-173
fractionated cavity viruses, 177
header infection viruses, 173
history of Win32 viruses, 157
KERNEL32.DLL infection, 175-176
LE (linear executable) file format, 160
lfanew field modification, 178
prepending viruses, 174
system loader comparison with Windows NT, 181-183
VxD-based viruses, 178-180
Win32 platform support, 158

Windows 95 System Programming Secrets, 616

Windows 98/ME, Win32 platform support, 158
Windows 9x, kernel mode, 228-229

Windows CE
device translator layer dependent viruses, 109-112
Win32 platform support, 158

Windows NT
class of context (memory scanning), 526
executed images (Win32 viruses), 512-514
filter driver virus deactivation (memory scanning), 527-529
functions (memory scanning), 525
hidden window procedure (Win32 viruses), 512
memory scanning
and paging, 515-517
processes/rights, 507-508
native viruses, 496
service API entry points (memory scanning), 524
service viruses, 512
system loader comparison with Windows, 95, 181-183
upper 2G of address space (memory scanning), 527
virtual memory system, 499-505

Win32 platform support, 158
Win32 viruses, 508-511

Windows Update Web site, DoS attack against, 413

Windows XP, Win32 platform support, 158
WinNT/RemEx (virus), 496

Winvir. See W16/Winvir (virus)
wireless mobile worms, 359-361

WM/Cap.A (virus), 72, 157
WM/Concept (virus), 296
WM/Concept.A (virus), 67
WM/DMV (virus), 67
WM/Hot.A (virus), 73
WM/Npad (virus), 70
WM/ShareFun (worm), 314
WM/Wazzu.A (virus), 301
WNT/RemEx (virus), 512, 518
WNT/Stream (virus), 58
Word Pro viruses, 94
Word viruses. See macro viruses
WordSwap (virus), 260, 303

worm blocking techniques, 538-542, 557
buffer overflow attacks
blocking, 543-544
code reviews, 544
compiler-level solutions, 545-552
kernel-mode extensions, 554-556
opreating system-level solutions, 552-554
program shepherding, 556
subsystem extensions, 554
connections, 574-575
exception-handler validation, 565-569
GOT/IAT page attributes, 574
injected code detection, 557-562
return-to-LIBC attacks, 569-573
script/SMTP blocking, 539-542
self-sending code blocking, 563-565

worms
backdoor features, 309-311
behavior patterns, 598-608
code propagation techniques, 338
INDEX

code injection attacks, 341-342
executable code-based attacks, 339
HTML-based mail, 340
links to Web sites or proxies, 339-340
remote login-based attacks, 341
shell code-based attacks, 342-344
competition between, 357-358
cooperation with viruses, 354-357
definition of, 29-30, 314-315
future attacks, 575-578
outbreak statistics, 670
structure of, 315
   infection propagator, 315-316, 331-338
   life-cycle manager, 316-317
   payload activation, 318
   remote control, 316, 351-354
   self-tracking, 318
   target locator, 315, 319-330
   update interface, 316, 345-351
SWCP (simple worm communication protocol), 359
versus computer viruses, 314
wireless mobile worms, 359-361
writeable flag, 238
WS2_32!sentto() API, 564

X

X-RAY method, algorithmic scanning methods, 446-451
X97M/Jini.A (virus), 76
Xbox, security vulnerabilities, 347
XF/Paix (virus), 77
XM/Laroux (virus), 67
XML, macro viruses, 77
Xmorfic (virus writer), 88
XMS (Extended Memory Specification), 198
XTEA (extended tiny encryption algorithm), 346

Y

Yankee_Doodle (virus), 26, 54, 157, 219, 233

Z

Zachary, William B., 7
Zafi.A (worm), 320
Zbikowski, Mark, 60
zero bytes, 433
Zhengxi (virus writer), 100, 248, 348
   heuristic analysis, 472
   infection technique, 152
Zmist virus. See W95/Zmist (virus)
Zombie (virus writer), 27, 349
   ETG (executable trash generator) engine, 280
   ISO image infection, 59
   W95/Zmist virus, 155, 277
   W95/Zperm virus, 279
zoo viruses, 26
Zox. See INF/Zox (virus)