CONTENTS IN DETAIL

0x100 INTRODUCTION 1

0x200 PROGRAMMING 5

0x210 What Is Programming? 6
0x220 Pseudo-code 7
0x230 Control Structures 8
    0x231 If-Then-Else 8
    0x232 While/Until Loops 9
    0x233 For Loops 10
0x240 More Fundamental Programming Concepts 11
    0x241 Variables 11
    0x242 Arithmetic Operators 12
    0x243 Comparison Operators 14
    0x244 Functions 16
0x250 Getting Your Hands Dirty 19
        firstprog.c 19
    0x251 The Bigger Picture 20
    0x252 The *x*86 Processor 23
    0x253 Assembly Language 25
        ASCII Table 33
0x260 Back to Basics 37
    0x261 Strings 38
        char_array.c 38
        char_array2.c 39
    0x262 Signed, Unsigned, Long, and Short 41
        datatype_sizes.c 42
    0x263 Pointers 43
        pointer.c 44
        addressof.c 46
        addressof2.c 47
    0x264 Format Strings 48
        fmt_strings.c 48
        input.c 50
    0x265 Typecasting 51
        typecasting.c 51
        pointer_types.c 52
        pointer_types2.c 53
        pointer_types3.c 55
        pointer_types4.c 56
        pointer_types5.c 57
    0x266 Command-line Arguments 58
        commandline.c 58
        convert.c 59
        convert2.c 60

0x267 Variable Scoping 62
scope.c 62
scope2.c 63
scope3.c 64
static.c 67
static2.c 68
0x270 Memory Segmentation 69
stack_example.c 71
0x271 Memory Segments in C 75
memory_segments.c 75
0x272 Using the Heap 77
heap_example.c 77
0x273 Error-checked malloc() 80
errorchecked_heap.c 80
0x280 Building on Basics 81
0x281 File Access 81
simplenote.c 82
bitwise.c 84
fcntl_flags.c 85
0x282 File Permissions 87
0x283 User IDs 88
uid_demo.c 90
hacking.h 91
notetaker.c 91
notesearch.c 93
0x284 Structs 96
time_example.c 97
time_example2.c 98
0x285 Function Pointers 100
funcptr_example.c 100
0x286 Pseudo-random Numbers 101
rand_example.c 101
0x287 A Game of Chance 102
game_of_chance.c 103

0x300
EXPLOITATION 117

0x310 Generalized Exploit Techniques 120
0x320 Buffer Overflows 121
overflow_example.c 121
exploit_notesearch.c 123
0x321 Stack-based Buffer Overflow Vulnerabilities 124
auth_overflow.c 124
auth_overflow2.c 128
0x330 Experimenting with BASH 135
From exploit_notesearch.c 142
0x331 Using the Environment 144
getenv_example.c 148
getenvaddr.c 150

Code from libc-2.2.2 151
exploit_notesearch_env.c 152
0x340 Overflows in Other Segments 152
0x341 A Basic Heap-based Overflow 153
0x342 Overflowing Function Pointers 158
From game_of_chance.c 158
0x350 Format Strings 170
0x351 Format Parameters 170
fmt_uncommon.c 170
0x352 The Format String Vulnerability 173
fmt_vuln.c 173
0x353 Reading from Arbitrary Memory Addresses 175
0x354 Writing to Arbitrary Memory Addresses 176
0x355 Direct Parameter Access 183
0x356 Using Short Writes 185
0x357 Detours with .dtors 187
dtors_sample.c 187
0x358 Another notesearch Vulnerability 192
0x359 Overwriting the Global Offset Table 193

0x400
NETWORKING 197

0x410 OSI Model 198
0x420 Sockets 200
0x421 Socket Functions 201
From /usr/include/bits/socket.h 202
From /usr/include/bits/socket.h 202
0x422 Socket Addresses 202
From /usr/include/bits/socket.h 203
From /usr/include/bits/socket.h 203
From /usr/include/netinet/in.h 204
0x423 Network Byte Order 204
0x424 Internet Address Conversion 205
0x425 A Simple Server Example 205
Added to hacking.h 206
simple_server.c 206
From a remote machine 209
On local machine 209
0x426 A Web Client Example 209
From /etc/services 210
hacking-network.h 211
From /usr/include/netdb.h 213
host_lookup.c 213
webserver_id.c 214
0x427 A Tiny Webserver 215
tinyweb.c 216
0x430 Peeling Back the Lower Layers 219
0x431 Data-Link Layer 220
0x432 Network Layer 222
From RFC 791 222

0x433 Transport Layer 223

From RFC 793 224

0x440 Network Sniffing 226

0x441 Raw Socket Sniffer 228

raw_tcpsniff.c 228

0x442 libpcap Sniffer 230

pcap_sniff.c 230

0x443 Decoding the Layers 232

From /usr/include/if_ether.h 232

Added to hacking-network.h 233

From /usr/include/netinet/ip.h 233

From RFC 791 234

Added to hacking-network.h 234

From /usr/include/netinet/tcp.h 235

From RFC 793 235

Added to hacking-network.h 236

decode_sniff.c 237

0x444 Active Sniffing 241

From nemesis-arp.c 247

From nemesis.h 248

From nemesis-arp.c 248

From nemesis-proto_arp.c 249

From the libnet man page 250

From the arpspoof man page 251

arpspoof.c 252

From the libnet man page 253

0x450 Denial of Service 253

0x451 SYN Flooding 254

synflood.c 254

0x452 Ping of Death 258

0x453 Teardrop 258

0x454 Ping Flooding 259

0x455 Amplification Attacks 259

0x456 Distributed DoS Flooding 260

0x460 TCP/IP Hijacking 260

0x461 RST Hijacking 261

rst_hijack.c 263

0x462 Continued Hijacking 266

0x470 Port Scanning 266

0x471 Stealth SYN Scan 266

0x472 FIN, X-mas, and Null Scans 267

0x473 Spoofing Decoys 267

0x474 Idle Scanning 268

0x475 Proactive Defense (Shroud) 269

shroud.c 271

0x480 Reach Out and Hack Someone 275

From hacking-network.h 275

0x481 Analysis with GDB 276

0x482 Almost Only Counts with Hand Grenades 278

tinyweb_exploit.c 278

0x483 Port-Binding Shellcode 281

New line from tinyweb_exploit2.c 281

0x500 SHELLCODE 285

0x510 Assembly vs. C 286
helloworld.c 286
Manual page for the write() system call 287
From /usr/include/unistd.h 288
0x511 Linux System Calls in Assembly 288
From /usr/include/asm-i386/unistd.h 288
helloworld.asm 289
0x520 The Path to Shellcode 290
0x521 Assembly Instructions Using the Stack 291
helloworld1.s 291
0x522 Investigating with GDB 293
0x523 Removing Null Bytes 294
helloworld2.s 295
helloworld3.s 298
0x530 Shell-Spawning Shellcode 299
exec_shell.c 300
exec_shell.s 301
tiny_shell.s 302
0x531 A Matter of Privilege 303
drop_privs.c 304
priv_shell.s 305
0x532 And Smaller Still 306
shellcode.s 306
0x540 Port-Binding Shellcode 307
bind_port.c 307
From /usr/include/linux/net.h 308
bind_port.s 310
0x541 Duplicating Standard File Descriptors 311
New instructions from bind_shell1.s 312
0x542 Branching Control Structures 313
bind_shell.s 316
0x550 Connect-back Shellcode 318
connectback_shell.s 318

0x600 COUNTERMEASURES 323

0x610 Countermeasures That Detect 324
0x620 System Daemons 325
0x621 Crash Course in Signals 326
signal_example.c 326
0x622 Tiny Web Daemon 328
tinywebd.c 328
0x630 Tools of the Trade 332
0x631 tinywebd Exploit Tool 332
xtool_tinywebd.sh 336

0x640 Log Files 337
tinywebd log file 337
0x641 Blend In with the Crowd 338
xtool_tinywebd_stealth.sh 338
0x650 Overlooking the Obvious 339
0x651 One Step at a Time 340
mark.s 343
0x652 Putting Things Back Together Again 344
mark_break.s 346
mark_restore.s 348
0x653 Child Laborers 349
loopback_shell_restore.s 350
0x660 Advanced Camouflage 351
0x661 Spoofing the Logged IP Address 351
Code segment from tinywebd.c 351
addr_struct.c 352
xtool_tinywebd_spoof.sh 353
0x662 Logless Exploitation 355
xtool_tinywebd_silent.sh 357
0x670 The Whole Infrastructure 358
0x671 Socket Reuse 358
Excerpt from tinywebd.c 358
socket_reuse_restore.s 360
xtool_tinywebd_reuse.sh 361
0x680 Payload Smuggling 362
0x681 String Encoding 363
encoded_sockreuserestore_dbg.s 363
0x682 How to Hide a Sled 366
0x690 Buffer Restrictions 367
update_info.c 367
0x691 Polymorphic Printable ASCII Shellcode 369
printable_helper.c 373
printable.s 375
0x6a0 Hardening Countermeasures 380
0x6b0 Nonexecutable stack 380
0x6b1 ret2libc 380
0x6b2 Returning into system() 381
vuln.c 381
0x6c0 Randomized Stack Space 383
aslr_demo.c 383
0x6c1 Investigations with BASH and GDB 384
0x6c2 Bouncing Off linux-gate 388
find_jmpesp.c 389
0x6c3 Applied Knowledge 392
0x6c4 A First Attempt 392
aslr_execl.c 392
0x6c5 Playing the Odds 393
aslr_execl_exploit.c 394

0x700
CRYPTOLOGY 397

0x710 Information Theory 398
0x711 Unconditional Security 398
0x712 One-Time Pads 398
0x713 Quantum Key Distribution 399
0x714 Computational Security 400
0x720 Algorithmic Runtime 401
0x721 Asymptotic Notation 402
0x730 Symmetric Encryption 402
0x731 Lov Grover's Quantum Search Algorithm 403
0x740 Asymmetric Encryption 404
0x741 RSA 404
0x742 Peter Shor's Quantum Factoring Algorithm 408
0x750 Hybrid Ciphers 410
0x751 Man-in-the-Middle Attacks 410
On machine 192.168.42.250 (tetsuo), connecting to 192.168.42.72 (loki) 412
Attacker's machine 413
0x752 Differing SSH Protocol Host Fingerprints 414
From 192.168.42.250 (tetsuo), just an innocent machine on the network 416
On the attacker's machine, setting up mitm-ssh to only use SSH1 protocol 416
Now back on 192.168.42.250 (tetsuo) 416
0x753 Fuzzy Fingerprints 417
Normal connection 421
Man-in-the-middle attacked connection 421
0x760 Password Cracking 422
crypt_test.c 422
0x761 Dictionary Attacks 423
crypt_crack.c 424
0x762 Exhaustive Brute-Force Attacks 426
0x763 Hash Lookup Table 427
0x764 Password Probability Matrix 428
ppm_gen.c 430
ppm_crack.c 432
0x770 Wireless 802.11b Encryption 437
0x771 Wired Equivalent Privacy (WEP) 438
0x772 RC4 Stream Cipher 439
0x780 WEP Attacks 440
0x781 Offline Brute-Force Attacks 440
0x782 Keystream Reuse 441
0x783 IV-Based Decryption Dictionary Tables 442
0x784 IP Redirection 442
0x785 Fluhrer, Mantin, and Shamir (FMS) Attack 443
fms.c 447

0x800
CONCLUSION 455

0x801 References 456
0x802 Sources 458