"A great reference for penetration testers and researchers who want to step up and broaden their skills in a wide range of IT security disciplines."

Peter Van Eeckhoutte (corelanc0d3r) – Founder, Corelan Team

Third Edition

GRAY HAT HACKING

The Ethical Hacker's Handbook

Allen Harper
Shon Harris
Jonathan Ness
Chris Eagle
Gideon Lenkey
Terron Williams

CONTENTS

Preface xxiii
Acknowledgments xxv
Introduction xxvii

Part I Introduction to Ethical Disclosure 1

Chapter 1 Ethics of Ethical Hacking 3
Why You Need to Understand Your Enemy's Tactics 3
Recognizing the Gray Areas in Security 8
How Does This Stuff Relate to an Ethical Hacking Book? 10
Vulnerability Assessment 10
Penetration Testing 11
The Controversy of Hacking Books and Classes 15
The Dual Nature of Tools 16
Recognizing Trouble When It Happens 18
Emulating the Attack 19
Where Do Attackers Have Most of Their Fun? 19
Security Does Not Like Complexity 20

Chapter 2 Ethical Hacking and the Legal System 23
The Rise of Cyberlaw 23
Understanding Individual Cyberlaws 25
18 USC Section 1029: The Access Device Statute 25
18 USC Section 1030 of the Computer Fraud and Abuse Act 29
18 USC Sections 2510, et. Seq., and 2701, et. Seq., of the Electronic Communication Privacy Act 38
Digital Millennium Copyright Act (DMCA) 42
Cyber Security Enhancement Act of 2002 45
Securely Protect Yourself Against Cyber Trespass Act (SPY Act) 46

Chapter 3 Proper and Ethical Disclosure 47
Different Teams and Points of View 48
How Did We Get Here? 49
CERT's Current Process 50
Full Disclosure Policy—the RainForest Puppy Policy 52
Organization for Internet Safety (OIS) 54
Discovery 54
Notification 55
Validation 57
Resolution 59
Release 61
Conflicts Will Still Exist 62
"No More Free Bugs" 63

Case Studies 67
Pros and Cons of Proper Disclosure Processes 67
Vendors Paying More Attention 71
So What Should We Do from Here on Out? 72
iDefense and ZDI 72

Part II Penetration Testing and Tools 75

Chapter 4 Social Engineering Attacks 77
How a Social Engineering Attack Works 77
Conducting a Social Engineering Attack 79
Common Attacks Used in Penetration Testing 81
The Good Samaritan 81
The Meeting 86
Join the Company 88
Preparing Yourself for Face-to-Face Attacks 89
Defending Against Social Engineering Attacks 91

Chapter 5 Physical Penetration Attacks 93
Why a Physical Penetration Is Important 94
Conducting a Physical Penetration 94
Reconnaissance 95
Mental Preparation 97
Common Ways into a Building 97
The Smokers' Door 98
Manned Checkpoints 99
Locked Doors 102
Physically Defeating Locks 103
Once You Are Inside 107
Defending Against Physical Penetrations 108

Chapter 6 Insider Attacks 109
Why Simulating an Insider Attack Is Important 109
Conducting an Insider Attack 110
Tools and Preparation 110
Orientation 111
Gaining Local Administrator Privileges 111
Disabling Antivirus 115
Raising Cain 116
Defending Against Insider Attacks 123

Chapter 7 Using the BackTrack Linux Distribution 125
BackTrack: The Big Picture 125
Installing BackTrack to DVD or USB Thumb Drive 126
Using the BackTrack ISO Directly Within a Virtual Machine 128
Creating a BackTrack Virtual Machine with VirtualBox 128
Booting the BackTrack LiveDVD System 129
Exploring the BackTrack X Windows Environment 130

Starting Network Services 130
Persisting Changes to Your BackTrack Installation 131
Installing Full BackTrack to Hard Drive or USB Thumb Drive 131
Creating a New ISO with Your One-time Changes 134
Using a Custom File that Automatically Saves and Restores Changes 135
Exploring the BackTrack Boot Menu 137
Updating BackTrack 139

Chapter 8 Using Metasploit 141
Metasploit: The Big Picture 141
Getting Metasploit 141
Using the Metasploit Console to Launch Exploits 142
Exploiting Client-Side Vulnerabilities with Metasploit 147
Penetration Testing with Metasploit's Meterpreter 149
Automating and Scripting Metasploit 155
Going Further with Metasploit 156

Chapter 9 Managing a Penetration Test 157
Planning a Penetration Test 157
Types of Penetration Tests 157
Scope of a Penetration Test 158
Locations of the Penetration Test 158
Organization of the Penetration Testing Team 158
Methodologies and Standards 159
Phases of the Penetration Test 159
Testing Plan for a Penetration Test 161
Structuring a Penetration Testing Agreement 161
Statement of Work 161
Get-Out-of-Jail-Free Letter 162
Execution of a Penetration Test 162
Kickoff Meeting 162
Access During the Penetration Test 163
Managing Expectations 163
Managing Problems 163
Steady Is Fast 164
External and Internal Coordination 164
Information Sharing During a Penetration Test 164
Dradis Server 164
Reporting the Results of a Penetration Test 168
Format of the Report 169
Out Brief of the Report 169

Part III Exploiting 171

Chapter 10 Programming Survival Skills 173
C Programming Language 173
Basic C Language Constructs 173

Sample Program 178
Compiling with gcc 179
Computer Memory 180
Random Access Memory (RAM) 180
Endian 180
Segmentation of Memory 181
Programs in Memory 181
Buffers 182
Strings in Memory 182
Pointers 182
Putting the Pieces of Memory Together 183
Intel Processors 184
Registers 184
Assembly Language Basics 184
Machine vs. Assembly vs. C 185
AT&T vs. NASM 185
Addressing Modes 188
Assembly File Structure 189
Assembling 189
Debugging with gdb 190
gdb Basics 190
Disassembly with gdb 191
Python Survival Skills 192
Getting Python 192
Hello World in Python 193
Python Objects 193
Strings 193
Numbers 195
Lists 196
Dictionaries 197
Files with Python 197
Sockets with Python 199

Chapter 11 Basic Linux Exploits 201
Stack Operations 201
Function Calling Procedure 202
Buffer Overflows 203
Overflow of meet.c 204
Ramifications of Buffer Overflows 208
Local Buffer Overflow Exploits 209
Components of the Exploit 209
Exploiting Stack Overflows from the Command Line 211
Exploiting Stack Overflows with Generic Exploit Code 213
Exploiting Small Buffers 215
Exploit Development Process 217
Control eip 218
Determine the Offset(s) 218

Determine the Attack Vector 221
Build the Exploit Sandwich 222
Test the Exploit 222

Chapter 12 Advanced Linux Exploits 225
Format String Exploits 225
The Problem 225
Reading from Arbitrary Memory 229
Writing to Arbitrary Memory 231
Taking .dtors to root 233
Memory Protection Schemes 236
Compiler Improvements 236
Kernel Patches and Scripts 240
Return to libc Exploits 241
Bottom Line 249

Chapter 13 Shellcode Strategies 251
User Space Shellcode 251
System Calls 252
Basic Shellcode 252
Port Binding Shellcode 253
Reverse Shellcode 254
Find Socket Shellcode 256
Command Execution Code 257
File Transfer Code 257
Multistage Shellcode 258
System Call Proxy Shellcode 258
Process Injection Shellcode 259
Other Shellcode Considerations 260
Shellcode Encoding 260
Self-Corrupting Shellcode 261
Disassembling Shellcode 262
Kernel Space Shellcode 263
Kernel Space Considerations 264

Chapter 14 Writing Linux Shellcode 267
Basic Linux Shellcode 267
System Calls 268
System Calls by C 268
System Calls by Assembly 269
Exit System Call 269
setreuid System Call 271
Shell-Spawning Shellcode with execve 272
Implementing Port-Binding Shellcode 276
Linux Socket Programming 276
Assembly Program to Establish a Socket 279
Test the Shellcode 281

Implementing Reverse Connecting Shellcode 284
Reverse Connecting C Program 284
Reverse Connecting Assembly Program 285
Encoding Shellcode 287
Simple XOR Encoding 287
Structure of Encoded Shellcode 288
JMP/CALL XOR Decoder Example 288
FNSTENV XOR Example 289
Putting the Code Together 291
Automating Shellcode Generation with Metasploit 294
Generating Shellcode with Metasploit 294
Encoding Shellcode with Metasploit 295

Chapter 15 Windows Exploits **297**
Compiling and Debugging Windows Programs 297
Compiling on Windows 297
Debugging on Windows with OllyDbg 299
Writing Windows Exploits 304
Exploit Development Process Review 305
ProSSHD Server 305
Control eip 306
Determine the Offset(s) 308
Determine the Attack Vector 309
Build the Exploit Sandwich 312
Debug the Exploit if Needed 314
Understanding Structured Exception Handling (SEH) 316
Implementation of SEH 316
Understanding Windows Memory Protections (XP SP3, Vista, 7, and Server 2008) 318
Stack-Based Buffer Overrun Detection (/GS) 318
Safe Structured Exception Handling (SafeSEH) 320
SEH Overwrite Protection (SEHOP) 320
Heap Protections 320
Data Execution Prevention (DEP) 321
Address Space Layout Randomization (ASLR) 321
Bypassing Windows Memory Protections 322
Bypassing /GS 323
Bypassing SafeSEH 323
Bypassing ASLR 324
Bypassing DEP 325
Bypassing SEHOP 331
Summary of Memory Bypass Methods 338

Chapter 16 Understanding and Detecting Content-Type Attacks **341**
How Do Content-Type Attacks Work? 341
Which File Formats Are Being Exploited Today? 343
Intro to the PDF File Format 345

Analyzing a Malicious PDF Exploit 348
Implementing Safeguards in Your Analysis Environment 350
Tools to Detect Malicious PDF Files 351
PDFiD 351
pdf-parser.py 355
Tools to Test Your Protections Against Content-type Attacks 358
How to Protect Your Environment from Content-type Attacks 359
Apply All Security Updates 359
Disable JavaScript in Adobe Reader 359
Enable DEP for Microsoft Office Application and Adobe Reader 360

Chapter 17 Web Application Security Vulnerabilities 361
Overview of Top Web Application Security Vulnerabilities 361
Injection Vulnerabilities 361
Cross-Site Scripting Vulnerabilities 362
The Rest of the OWASP Top Ten 362
SQL Injection Vulnerabilities 362
SQL Databases and Statements 365
Testing Web Applications to Find SQL Injection Vulnerabilities 367
Cross-Site Scripting Vulnerabilities 373
Explaining "Scripting" 373
Explaining Cross-Site Scripting 374

Chapter 18 VoIP Attacks 379
What Is VoIP? 379
Protocols Used by VoIP 380
SIP 381
Megaco H.248 382
H.323 382
TLS and DTLS 383
SRTP 384
ZRTP 384
Types of VoIP Attacks 384
Enumeration 384
SIP Password Cracking 386
Eavesdropping/Packet Capture 386
Denial of Service 387
How to Protect Against VoIP Attacks 393

Chapter 19 SCADA Attacks 395
What Is SCADA? 395
Which Protocols Does SCADA Use? 396
OPC 396
ICCP 396
Modbus 397
DNP3 398

SCADA Fuzzing 399
SCADA Fuzzing with Autodafé 399
SCADA Fuzzing with TFTP Daemon Fuzzer 405
Stuxnet Malware (The New Wave in Cyberterrorism) 408
How to Protect Against SCADA Attacks 408

Part IV Vulnerability Analysis 411

Chapter 20 Passive Analysis 413
Ethical Reverse Engineering 413
Why Bother with Reverse Engineering? 414
Reverse Engineering Considerations 415
Source Code Analysis 416
Source Code Auditing Tools 416
The Utility of Source Code Auditing Tools 418
Manual Source Code Auditing 420
Automated Source Code Analysis 425
Binary Analysis 427
Manual Auditing of Binary Code 427
Automated Binary Analysis Tools 441

Chapter 21 Advanced Static Analysis with IDA Pro 445
Static Analysis Challenges 445
Stripped Binaries 446
Statically Linked Programs and FLAIR 448
Data Structure Analysis 454
Quirks of Compiled C++ Code 459
Extending IDA Pro 461
Scripting with IDC 461
IDA Pro Plug-In Modules and the IDA Pro SDK 464
Building IDA Pro Plug-Ins 466
IDA Pro Loaders and Processor Modules 468

Chapter 22 Advanced Reverse Engineering 471
Why Try to Break Software? 471
Overview of the Software Development Process 472
Instrumentation Tools 473
Debuggers 474
Code Coverage Analysis Tools 476
Profiling Tools 477
Flow Analysis Tools 477
Memory Use Monitoring Tools 480
Fuzzing 484
Instrumented Fuzzing Tools and Techniques 484
A Simple URL Fuzzer 485
Fuzzing Unknown Protocols 487
SPIKE 488

SPIKE Static Content Primitives 489
SPIKE Proxy 492
Sharefuzz 492

Chapter 23 Client-Side Browser Exploits **495**
Why Client-Side Vulnerabilities Are Interesting 495
Client-Side Vulnerabilities Bypass Firewall Protections 495
Client-Side Applications Are Often Running with Administrative Privileges 496
Client-Side Vulnerabilities Can Easily Target Specific People or Organizations 496
Internet Explorer Security Concepts 497
ActiveX Controls 497
Internet Explorer Security Zones 498
History of Client-Side Exploits and Latest Trends 499
Client-Side Vulnerabilities Rise to Prominence 499
Notable Vulnerabilities in the History of Client-Side Attacks 500
Finding New Browser-Based Vulnerabilities 506
mangleme 506
Mozilla Security Team Fuzzers 509
AxEnum 510
AxFuzz 515
AxMan 515
Heap Spray to Exploit 521
InternetExploiter 521
Protecting Yourself from Client-Side Exploits 522
Keep Up-to-Date on Security Patches 522
Stay Informed 522
Run Internet-Facing Applications with Reduced Privileges 522

Chapter 24 Exploiting the Windows Access Control Model **525**
Why Access Control Is Interesting to a Hacker 525
Most People Don't Understand Access Control 525
Vulnerabilities You Find Are Easy to Exploit 526
You'll Find Tons of Security Vulnerabilities 526
How Windows Access Control Works 526
Security Identifier 527
Access Token 528
Security Descriptor 531
The Access Check 535
Tools for Analyzing Access Control Configurations 538
Dumping the Process Token 538
Dumping the Security Descriptor 541
Special SIDs, Special Access, and "Access Denied" 543
Special SIDs 543
Special Access 545
Investigating "Access Denied" 545

Analyzing Access Control for Elevation of Privilege 553
Attack Patterns for Each Interesting Object Type 554
Attacking Services 554
Attacking Weak DACLs in the Windows Registry 560
Attacking Weak Directory DACLs 564
Attacking Weak File DACLs 569
What Other Object Types Are Out There? 573
Enumerating Shared Memory Sections 573
Enumerating Named Pipes 574
Enumerating Processes 575
Enumerating Other Named Kernel Objects (Semaphores, Mutexes, Events, Devices) 576

Chapter 25 Intelligent Fuzzing with Sulley **579**
Protocol Analysis 579
Sulley Fuzzing Framework 581
Installing Sulley 581
Powerful Fuzzer 581
Blocks 584
Monitoring the Process for Faults 588
Monitoring the Network Traffic 589
Controlling VMware 589
Putting It All Together 590
Postmortem Analysis of Crashes 592
Analysis of Network Traffic 593
Exploring Further 594

Chapter 26 From Vulnerability to Exploit **595**
Exploitability 596
Debugging for Exploitation 596
Initial Analysis 597
Understanding the Problem 601
Preconditions and Postconditions 602
Repeatability 603
Payload Construction Considerations 611
Payload Protocol Elements 612
Buffer Orientation Problems 612
Self-Destructive Shellcode 613
Documenting the Problem 614
Background Information 614
Circumstances 614
Research Results 615

Chapter 27 Closing the Holes: Mitigation **617**
Mitigation Alternatives 617
Port Knocking 618
Migration 618

Patching 619
Source Code Patching Considerations 620
Binary Patching Considerations 622
Binary Mutation 626
Third-Party Patching Initiatives 631

Part V Malware Analysis 633

Chapter 28 Collecting Malware and Initial Analysis 635
Malware 635
Types of Malware 635
Malware Defensive Techniques 636
Latest Trends in Honeynet Technology 637
Honeypots 637
Honeynets 637
Why Honeypots Are Used 637
Limitations of Honeypots 638
Low-Interaction Honeypots 639
High-Interaction Honeypots 639
Types of Honeynets 640
Thwarting VMware Detection Technologies 642
Catching Malware: Setting the Trap 644
VMware Host Setup 644
VMware Guest Setup 644
Using Nepenthes to Catch a Fly 644
Initial Analysis of Malware 646
Static Analysis 646
Live Analysis 648
Norman SandBox Technology 653

Chapter 29 Hacking Malware 657
Trends in Malware 657
Embedded Components 657
Use of Encryption 658
User Space Hiding Techniques 658
Use of Rootkit Technology 659
Persistence Measures 659
De-obfuscating Malware 660
Packer Basics 660
Unpacking Binaries 661
Reverse-Engineering Malware 669
Malware Setup Phase 670
Malware Operation Phase 670
Automated Malware Analysis 671

Index 673