Chapter 1 Looking at the Ecosystem 1

Understanding Android's Roots 1

Company History 2

Version History 2

Examining the Device Pool 4

Open Source, Mostly 7

Understanding Android Stakeholders 7

Google 8

Hardware Vendors 10

Carriers 12

Developers 13

Users 14

Grasping Ecosystem Complexities 15

Fragmentation 16

Compatibility 17

Update Issues 18

Security versus Openness 21

Public Disclosures 22

Summary 23

Chapter 2 Android Security Design and Architecture 25

Understanding Android System Architecture 25

Understanding Security Boundaries and Enforcement 27

Android's Sandbox 27

Android Permissions 30

Looking Closer at the Layers 34

Android Applications 34

The Android Framework 39

The Dalvik Virtual Machine 40

User-Space Native Code 41

The Kernel 49

Complex Security, Complex Exploits 55

Summary 56

Chapter 3 Rooting Your Device 57

Understanding the Partition Layout 58

Determining the Partition Layout 59

Understanding the Boot Process 60

Accessing Download Mode 61

Locked and Unlocked Boot Loaders 62

Stock and Custom Recovery Images 63

Rooting with an Unlocked Boot Loader 65

Rooting with a Locked Boot Loader 68

Gaining Root on a Booted System 69

NAND Locks, Temporary Root, and Permanent Root 70

Persisting a Soft Root 71

History of Known Attacks 73

Kernel: Wunderbar/asroot 73

Recovery: Volez 74

Udev: Exploid 74

Adbd: RageAgainstTheCage 75

Zygote: Zimperlich and Zysploit 75

Ashmem: KillingInTheNameOf and psneuter 76

Vold: GingerBreak 76

PowerVR: levitator 77

Libsysutils: zergRush 78

Kernel: mempodroid 78

File Permission and Symbolic Link–Related Attacks 79

Adb Restore Race Condition 79

Exynos4: exynos-abuse 80

Diag: lit / diaggetroot 81

Summary 81

Chapter 4 Reviewing Application Security 83

Common Issues 83

App Permission Issues 84

Insecure Transmission of Sensitive Data 86

Insecure Data Storage 87

Information Leakage Through Logs 88

Unsecured IPC Endpoints 89

Case Study: Mobile Security App 91

Profiling 91

Static Analysis 93

Dynamic Analysis 109

Attack 117

Case Study: SIP Client 120

Enter Drozer 121

Discovery 121

Snarfing 122

Injection 124

Summary 126

Chapter 5 Understanding Android's Attack Surface 129

An Attack Terminology Primer 130

Attack Vectors 130

Attack Surfaces 131

Classifying Attack Surfaces 133

Surface Properties 133

Classification Decisions 134

Remote Attack Surfaces 134

Networking Concepts 134

Networking Stacks 139

Exposed Network Services 140

Mobile Technologies 142

Client-side Attack Surface 143

Google Infrastructure 148

Physical Adjacency 154

Wireless Communications 154

Other Technologies 161

Local Attack Surfaces 161

Exploring the File System 162

Finding Other Local Attack Surfaces 163

Physical Attack Surfaces 168

Dismantling Devices 169

USB 169

Other Physical Attack Surfaces 173

Third-Party Modifications 174

Summary 174

Chapter 6 Finding Vulnerabilities with Fuzz Testing 177

Fuzzing Background 177

Identifying a Target 179

Crafting Malformed Inputs 179

Processing Inputs 180

Monitoring Results 181

Fuzzing on Android 181

Fuzzing Broadcast Receivers 183

Identifying a Target 183

Generating Inputs 184

Delivering Inputs 185

Monitoring Testing 185

Fuzzing Chrome for Android 188

Selecting a Technology to Target 188

Generating Inputs 190

Processing Inputs 192

Monitoring Testing 194

Fuzzing the USB Attack Surface 197

USB Fuzzing Challenges 198

Selecting a Target Mode 198

Generating Inputs 199

Processing Inputs 201

Monitoring Testing 202

Summary 204

Chapter 7 Debugging and Analyzing Vulnerabilities 205

Getting All Available Information 205

Choosing a Toolchain 207

Debugging with Crash Dumps 208

System Logs 208

Tombstones 209

Remote Debugging 211

Debugging Dalvik Code 212

Debugging an Example App 213

Showing Framework Source Code 215

Debugging Existing Code 217

Debugging Native Code 221

Debugging with the NDK 222

Debugging with Eclipse 226

Debugging with AOSP 227

Increasing Automation 233

Debugging with Symbols 235

Debugging with a Non-AOSP Device 241

Debugging Mixed Code 243

Alternative Debugging Techniques 243

Debug Statements 243

On-Device Debugging 244

Dynamic Binary Instrumentation 245

Vulnerability Analysis 246

Determining Root Cause 246

Judging Exploitability 260

Summary 261

Chapter 8 Exploiting User Space Software 263

Memory Corruption Basics 263

Stack Buffer Overflows 264

Heap Exploitation 268

A History of Public Exploits 275

GingerBreak 275

zergRush 279

mempodroid 283

Exploiting the Android Browser 284

Understanding the Bug 284

Controlling the Heap 287

Summary 290

Chapter 9 Return Oriented Programming 291

History and Motivation 291

Separate Code and Instruction Cache 292

Basics of ROP on ARM 294

ARM Subroutine Calls 295

Combining Gadgets into a Chain 297

Identifying Potential Gadgets 299

Case Study: Android 4.0.1 Linker 300

Pivoting the Stack Pointer 301

Executing Arbitrary Code from a New Mapping 303

Summary 308

Chapter 10 Hacking and Attacking the Kernel 309

Android’s Linux Kernel 309

Extracting Kernels 310

Extracting from Stock Firmware 311

Extracting from Devices 314

Getting the Kernel from a Boot Image 315

Decompressing the Kernel 316

Running Custom Kernel Code 316

Obtaining Source Code 316

Setting Up a Build Environment 320

Configuring the Kernel 321

Using Custom Kernel Modules 322

Building a Custom Kernel 325

Creating a Boot Image 329

Booting a Custom Kernel 331

Debugging the Kernel 336

Obtaining Kernel Crash Reports 337

Understanding an Oops 338

Live Debugging with KGDB 343

Exploiting the Kernel 348

Typical Android Kernels 348

Extracting Addresses 350

Case Studies 352

Summary 364

Chapter 11 Attacking the Radio Interface Layer 367

Introduction to the RIL 368

RIL Architecture 368

Smartphone Architecture 369

The Android Telephony Stack 370

Telephony Stack Customization 371

The RIL Daemon (rild) 372

The Vendor-RIL API 374

Short Message Service (SMS) 375

Sending and Receiving SMS Messages 376

SMS Message Format 376

Interacting with the Modem 379

Emulating the Modem for Fuzzing 379

Fuzzing SMS on Android 382

Summary 390

Chapter 12 Exploit Mitigations 391

Classifying Mitigations 392

Code Signing 392

Hardening the Heap 394

Protecting Against Integer Overflows 394

Preventing Data Execution 396

Address Space Layout Randomization 398

Protecting the Stack 400

Format String Protections 401

Read-Only Relocations 403

Sandboxing 404

Fortifying Source Code 405

Access Control Mechanisms 407

Protecting the Kernel 408

Pointer and Log Restrictions 409

Protecting the Zero Page 410

Read-Only Memory Regions 410

Other Hardening Measures 411

Summary of Exploit Mitigations 414

Disabling Mitigation Features 415

Changing Your Personality 416

Altering Binaries 416

Tweaking the Kernel 417

Overcoming Exploit Mitigations 418

Overcoming Stack Protections 418

Overcoming ASLR 418

Overcoming Data Execution Protections 419

Overcoming Kernel Protections 419

Looking to the Future 420

Official Projects Underway 420

Community Kernel Hardening Efforts 420

A Bit of Speculation 422

Summary 422

Chapter 13 Hardware Attacks 423

Interfacing with Hardware Devices 424

UART Serial Interfaces 424

I²C, SPI, and One-Wire Interfaces 428

JTAG 431

Finding Debug Interfaces 443

Identifying Components 456

Getting Specifications 456

Difficulty Identifying Components 457

Intercepting, Monitoring, and Injecting Data 459

USB 459

I ²C, SPI, and UART Serial Interfaces 463

Stealing Secrets and Firmware 469

Accessing Firmware Unobtrusively 469

Destructively Accessing the Firmware 471

What Do You Do with a Dump? 474

Pitfalls 479

Custom Interfaces 479

Binary/Proprietary Data 479

Blown Debug Interfaces 480

Chip Passwords 480

Boot Loader Passwords, Hotkeys, and Silent Terminals 480

Customized Boot Sequences 481

Unexposed Address Lines 481

Anti-Reversing Epoxy 482

Image Encryption, Obfuscation, and Anti-Debugging 482

Summary 482

Appendix A Tool Catalog 485

Development Tools 485

Android SDK 485

Android NDK 486

Eclipse 486

ADT Plug-In 486

ADT Bundle 486

Android Studio 487

Firmware Extraction and Flashing Tools 487

Binwalk 487

fastboot 487

Samsung 488

NVIDIA 489

LG 489

HTC 489

Motorola 490

Native Android Tools 491

BusyBox 491

setpropex 491

SQLite 491

strace 492

Hooking and Instrumentation Tools 492

ADBI Framework 492

ldpreloadhook 492

XPosed Framework 492

Cydia Substrate 493

Static Analysis Tools 493

Smali and Baksmali 493

Androguard 493

apktool 494

dex2jar 494

jad 494

JD-GUI 495

JEB 495

Radare 2 495

IDA Pro and Hex-Rays Decompiler 496

Application Testing Tools 496

Drozer (Mercury) Framework 496

iSEC Intent Sniffer and Intent Fuzzer 496

Hardware Hacking Tools 496

Segger J-Link 497

JTAGulator 497

OpenOCD 497

Saleae 497

Bus Pirate 497

GoodFET 497

Total Phase Beagle USB 498

Facedancer 21 498

Total Phase Beagle I^2c 498

Chip Quik 498

Hot air gun 498

Xeltek SuperPro 498

IDA 499

Appendix B Open Source Repositories 501

Google 501

AOSP 501

Gerrit Code Review 502

SoC Manufacturers 502

AllWinner 503

Intel 503

Marvell 503

MediaTek 504

Nvidia 504

Texas Instruments 504

Qualcomm 505

Samsung 505

OEMs 506

ASUS 506

HTC 507

LG 507

Motorola 507

Samsung 508

Sony Mobile 508

Upstream Sources 508

Others 509

Custom Firmware 509

Linaro 510

Replicant 510

Code Indexes 510

Individuals 510

Appendix C References 511

Index 523