

# The Mobile Application Hacker's Handbook



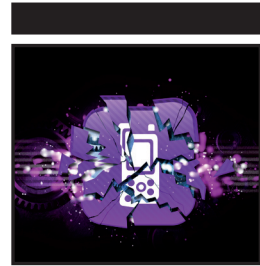
■ Dominic Chell ■ Tyrone Erasmus  
■ Shaun Colley ■ Ollie Whitehouse

WILEY



# **The Mobile Application Hacker's Handbook**





# **The Mobile Application Hacker's Handbook**

---

Dominic Chell  
Tyrone Erasmus  
Shaun Colley  
Ollie Whitehouse

**WILEY**

## The Mobile Application Hacker's Handbook

Published by  
**John Wiley & Sons, Inc.**  
10475 Crosspoint Boulevard  
Indianapolis, IN 46256  
[www.wiley.com](http://www.wiley.com)

Copyright © 2015 by John Wiley & Sons, Inc., Indianapolis, Indiana  
Published simultaneously in Canada

ISBN: 978-1-118-95850-6  
ISBN: 978-1-118-95852-0 (ebk)  
ISBN: 978-1-118-95851-3 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

**Limit of Liability/Disclaimer of Warranty:** The publisher and the author make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation warranties of fitness for a particular purpose. No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional services. If professional assistance is required, the services of a competent professional person should be sought. Neither the publisher nor the author shall be liable for damages arising herefrom. The fact that an organization or Web site is referred to in this work as a citation and/or a potential source of further information does not mean that the author or the publisher endorses the information the organization or website may provide or recommendations it may make. Further, readers should be aware that Internet websites listed in this work may have changed or disappeared between when this work was written and when it is read.

For general information on our other products and services please contact our Customer Care Department within the United States at (877) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://booksupport.wiley.com>. For more information about Wiley products, visit [www.wiley.com](http://www.wiley.com).

**Library of Congress Control Number:** 2014954689

**Trademarks:** Wiley and the Wiley logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

*I would like to dedicate this book to my wife Adele and thank her for her continued support not only whilst working on this book but throughout my career.*

—Dominic

*I would like to dedicate this book to Wendy, the love of my life. I cannot wait to spend my time with someone who understands me so well. You support me tirelessly in spite of me pursuing many time-consuming projects. You are owed many movie nights and a catch-up on time where I was absent while writing.*

—Tyrone

*I would like to dedicate this book to my parents, Jill and Andy, as well as my brother Dave, for all the support and encouragement they have given me over the years. My friends are also owed immensely for their support and friendship over the years.*

—Shaun

*I would like to dedicate this book to Ilma who for over a decade has kept the home fire burning whilst I've pursued my passion around the globe.*

—Ollie







## About the Authors

**Dominic Chell** is a cofounder of MDSec, where in addition to leading the mobile practice, he is responsible for delivering consultancy and training engagements for a variety of clients. Dominic's career has spanned over a decade and has been almost entirely focused on the technical aspects of application security. He has spoken at numerous conferences as well as releasing several publications on mobile security. Dominic is also listed as a subject matter expert for a secure iOS development exam.

**Tyrone Erasmus** has a degree in computer engineering and is currently the head of mobile security at MWR InfoSecurity South Africa. He enjoys delving into many different areas of penetration testing and security research, with a large portion of his research efforts in the past spent on Android. His interests lie predominantly in offensive security and the advancement of tools and new techniques in this sphere. He has spoken at various security conferences, and was part of the team that won the Android category at Mobile Pwn2Own in 2012. His work is acknowledged internationally in the Android hacking space, and he is known among peers as a well-rounded security professional.

**Shaun Colley** is a principal security consultant for IOActive where he focuses on mobile device security, native code review, and reverse engineering. During his career, he has been primarily focused on mobile security and reverse engineering. Shaun has also spoken several times at industry meets and conferences. He holds a BSc (Hons) in Chemistry from the University of Leeds, England.

**Ollie Whitehouse** is technical director for NCC Group, where he is responsible for Cyber Defence Operations, Managed Services, and its Exploit Development Group along technical innovation across the Technical Security Consulting practice. Ollie's career has spanned nearly two decades and included research, consultancy, and management positions at BlackBerry, Symantec, and @stake where he specialized in software, mobile, embedded, wireless, and telecommunications security.



## About the Technical Editor

**Rob Shimonski** ([www.shimonski.com](http://www.shimonski.com)) is a best-selling author and editor with over 15 years' experience developing, producing, and distributing print media in the form of books, magazines, and periodicals. To date, Rob has successfully created over 100 books that are currently in circulation. Rob has worked for countless companies to include CompTIA, Microsoft, Wiley, Cisco, the National Security Agency, and Digidesign.

Rob has over 20 years' experience working in IT, networking, systems, and security. He is a veteran of the US military and has been entrenched in security topics and assignments his entire professional career. In the military, Rob was assigned to a communications (radio) battalion supporting training efforts and exercises. Having worked with mobile phones since their inception, Rob is an expert in mobile phone development and security.





**Executive Editor**

Carol Long

**Project Editor**

Sydney Argenta

**Technical Editor**

Rob Shimonski

**Production Editor**

Rebecca Anderson

**Copy Editor**

Paula Lowell

**Manager of Content Development  
and Assembly**

Mary Beth Wakefield

**Marketing Director**

David Mayhew

**Marketing Manager**

Carrie Sherrill

**Professional Technology and  
Strategy Director**

Barry Pruett

**Business Manager**

Amy Knies

**Associate Publisher**

Jim Minatel

**Project Coordinator, Cover**

Patrick Redmond

**Proofreader**

Sarah Kaikini, Word One New York

**Indexer**

Johnna VanHoose Dinse

**Cover Designer**

Wiley

**Cover Image**

Clockwork gears © iStock.com/  
Ryhor Bruyeyu; App icon © iStock.  
com/ -cuba-





# Acknowledgments

Firstly, Dominic would like to thank the other authors for their hard work in developing this book; without their contributions it would have been too big a mountain to climb! Dominic would also like to acknowledge the support of his colleagues from MDSec, in particular Marcus Pinto, Dan Brown, Ryan Chell, and Matthew Hickey who worked tirelessly to pick up the slack whilst he was writing this book. He would also like to highlight the great work that the wider security community has done in this field and which provided a foundation for him to expand his knowledge—where applicable, this work has been properly referenced in this book. Dominic is also indebted to the numerous individuals that he has had the pleasure of working with through the years and from who he learnt so much, including Dafydd Stuttard, John Heasman, Peter Winter-Smith, Adam Matthews, Sherief Hammad, and the rest of the team at the old NGS Software. Finally, Dominic would like to thank his parents for everything that they have done and continue to do; their support has been invaluable over the years.

Tyrone would like to acknowledge Daniel and the rest of the team at MWR for tinkering alongside him on Android and sharing their knowledge, as well as Riaan and Harry for supporting him through his career. He would also like to acknowledge his family and friends who keep an active interest in his life and reminded him that there is life beyond his computer screen. Finally, Tyrone would like to thank Dominic for contacting him out of the blue to be a part of the author team!

Shaun would like to thank all the authors of this book in helping to make it a reality; who knows where the idea for this book would be without them. Shaun would also like to thank his colleagues at IOActive for their support while writing this book. He would like to acknowledge all those with whom

he has shared in interesting conversations about computer security and other completely unrelated real-life topics, including Dominic Chell, Marcus Pinto, Matthew Hickey, John Heasman, Ilja van Sprundel, Peter Winter-Smith, Ben Harrison-Smith, Vincent Berg, and Shane Macaulay, among others. Finally, Shaun would like to thank his parents, Jill and Andy, his brother Dave, and the rest of his family for their continued support during his career; as well as his friends, just for being awesome mates.

Ollie would like to say thanks to all the security researchers who have published their security research relating to BlackBerry technologies, including Zach Lanier, Ben Nell, Ralf-Philipp Weinmann, Shivang Desa, Tim Brown, Alex Plaskett, Daniel Martin Gomez, and Andy Davis. Without the hard work and perseverance of these individuals, public understanding would not be where it is today. He would also like to thank the numerous individuals he's been lucky enough to work closely with over the years and from whom he learned so much, including Foob, Nathan Catlow, Bambam, Rob Wood, Aaron Adams, Pete Beck, Paul Collett, Paul Ashton, Jeremy Boone, Jon Lindsay, Graham Murphy, and Ian Robertson. Ollie's final thanks is to Twitter, for providing continual distractions, and Kismet (the cat) for keeping him company on weekends whilst he wrote his chapters.

Finally, as a team, we are grateful to the people at Wiley—in particular, to Carol Long, Sydney Argenta, and the rest of our editorial team. Their help in developing and polishing our manuscript was invaluable, and apologies again for testing our deadlines. In particular, a big apology from Shaun who loves nothing more than leaving everything till the last minute!





# Contents at a Glance

Introduction	xxxix
<b>Chapter 1 Mobile Application (In)security</b>	<b>1</b>
<b>Chapter 2 Analyzing iOS Applications</b>	<b>17</b>
<b>Chapter 3 Attacking iOS Applications</b>	<b>69</b>
<b>Chapter 4 Identifying iOS Implementation Insecurities</b>	<b>133</b>
<b>Chapter 5 Writing Secure iOS Applications</b>	<b>149</b>
<b>Chapter 6 Analyzing Android Applications</b>	<b>173</b>
<b>Chapter 7 Attacking Android Applications</b>	<b>247</b>
<b>Chapter 8 Identifying and Exploiting Android Implementation Issues</b>	<b>353</b>
<b>Chapter 9 Writing Secure Android Applications</b>	<b>427</b>
<b>Chapter 10 Analyzing Windows Phone Applications</b>	<b>459</b>
<b>Chapter 11 Attacking Windows Phone Applications</b>	<b>511</b>
<b>Chapter 12 Identifying Windows Phone Implementation Issues</b>	<b>587</b>
<b>Chapter 13 Writing Secure Windows Phone Applications</b>	<b>629</b>
<b>Chapter 14 Analyzing BlackBerry Applications</b>	<b>647</b>
<b>Chapter 15 Attacking BlackBerry Applications</b>	<b>681</b>
<b>Chapter 16 Identifying BlackBerry Application Issues</b>	<b>693</b>
<b>Chapter 17 Writing Secure BlackBerry Applications</b>	<b>705</b>
<b>Chapter 18 Cross-Platform Mobile Applications</b>	<b>729</b>
Index	743





# Contents

<b>Introduction</b>	<b>xxxi</b>
<b>Chapter 1 Mobile Application (In)security</b>	<b>1</b>
The Evolution of Mobile Applications	2
Common Mobile Application Functions	3
Benefits of Mobile Applications	4
Mobile Application Security	4
Key Problem Factors	7
Underdeveloped Security Awareness	7
Ever-Changing Attack Surfaces	7
Economic and Time Constraints	7
Custom Development	8
The OWASP Mobile Security Project	8
OWASP Mobile Top Ten	9
OWASP Mobile Security Tools	12
The Future of Mobile Application Security	13
Summary	15
<b>Chapter 2 Analyzing iOS Applications</b>	<b>17</b>
Understanding the Security Model	17
Initializing iOS with Secure Boot Chain	18
Introducing the Secure Enclave	19
Restricting Application Processes with Code Signing	19
Isolating Applications with Process-Level Sandboxing	20
Protecting Information with Data-at-Rest Encryption	20
Protecting Against Attacks with Exploit Mitigation Features	21
Understanding iOS Applications	22
Distribution of iOS Applications	23
Apple App Store	23
Enterprise Distribution	24

Application Structure	24
Installing Applications	25
Understanding Application Permissions	26
Jailbreaking Explained	29
Reasons for Jailbreaking	29
Types of Jailbreaks	30
Building a Test Environment	33
Accessing the Device	33
Building a Basic Toolkit	34
Cydia	34
BigBoss Recommended Tools	34
Apple's CC Tools	35
Debuggers	38
Tools for Signing Binaries	39
Installipa	40
Exploring the Filesystem	40
Property Lists	42
Binary Cookies	42
SQLite Databases	42
Understanding the Data Protection API	43
Understanding the iOS Keychain	46
Access Control and Authentication Policies in iOS 8	48
Accessing the iOS Keychain	49
Understanding Touch ID	51
Reverse Engineering iOS Binaries	53
Analyzing iOS Binaries	53
Identifying Security-Related Features	56
Position-Independent Executable	56
Stack-Smashing Protection	57
Automatic Reference Counting	58
Decrypting App Store Binaries	59
Decrypting iOS Binaries Using a Debugger	59
Automating the Decryption Process	61
Inspecting Decrypted Binaries	62
Inspecting Objective-C Applications	62
Inspecting Swift Applications	63
Disassembling and Decompiling iOS Applications	67
Summary	67
<b>Chapter 3 Attacking iOS Applications</b>	<b>69</b>
Introduction to Transport Security	69
Identifying Transport Insecurities	70
Certificate Validation	71
SSL Session Security	76
Intercepting Encrypted Communications	78
Bypassing Certificate Pinning	80

---

Identifying Insecure Storage	81
Patching iOS Applications with Hopper	85
Attacking the iOS Runtime	92
Understanding Objective-C and Swift	93
Instrumenting the iOS Runtime	95
Introduction to Cydia Substrate	96
Using the Cydia Substrate C API	98
Tweak Development Using Theos and Logos	101
Instrumentation Using Cycrypt	104
Instrumentation Using Frida	110
Instrumenting the Runtime Using the Dynamic Linker	113
Inspecting iOS Applications using Snoop-it	115
Understanding Interprocess Communication	118
Attacking Protocol Handlers	118
Application Extensions	121
Attacking Using Injection	123
Injecting into UIWebViews	124
Injecting into Client-Side Data Stores	126
Injecting into XML	128
Injecting into File-Handling Routines	129
Summary	131
<b>Chapter 4 Identifying iOS Implementation Insecurities</b>	<b>133</b>
Disclosing Personally Identifiable Information	133
Handling Device Identifiers	134
Processing the Address Book	135
Handling Geolocation Data	135
Identifying Data Leaks	136
Leaking Data in Application Logs	137
Identifying Pasteboard Leakage	137
Handling Application State Transitions	138
Keyboard Caching	140
HTTP Response Caching	141
Memory Corruption in iOS Applications	142
Format String Vulnerabilities	142
Object Use-After-Free	145
Other Native Code Implementation Issues	146
Summary	146
<b>Chapter 5 Writing Secure iOS Applications</b>	<b>149</b>
Protecting Data in Your Application	149
General Design Principles	149
Implementing Encryption	151
Protecting Your Data in Transit	154
Avoiding Injection Vulnerabilities	156
Preventing SQL Injection	156
Avoiding Cross-Site Scripting	157

Securing Your Application with Binary Protections	158
Detecting Jailbreaks	159
Jailbreak Artifacts	160
Nondefault Open Ports	161
Weakening of the Sandbox	162
Evidence of System Modifications	162
Securing Your Application Runtime	163
Tamperproofing Your Application	167
Implementing Anti-Debugging Protections	168
Obfuscating Your Application	169
Summary	170
<b>Chapter 6 Analyzing Android Applications</b>	<b>173</b>
Creating Your First Android Environment	174
Understanding Android Applications	179
Reviewing Android OS Basics	179
Getting to Know Android Packages	181
Observing the Structure of a Package	182
Installing Packages	183
Using Tools to Explore Android	185
ADB	185
BusyBox	186
Standard Android Tools	188
drozer	189
Introduction to Application Components	196
Defining Components	198
Interacting with Components	199
Looking Under the Hood	201
Installing an Application	201
Running an Application	204
Understanding the Security Model	206
Code Signing	206
Discovered Vulnerabilities	210
Understanding Permissions	212
Inspecting the Android Permission Model	212
Protection Levels	216
Application Sandbox	219
Filesystem Encryption	221
Generic Exploit Mitigation Protections	222
Rooting Explained	226
Rooting Objectives	226
Rooting Methods	228
Reverse-Engineering Applications	233
Retrieving APK Files	234
Viewing Manifests	235
aapt	235

AXMLPrinter2	236
drozer	237
Disassembling DEX Bytecode	237
Dexdump	238
Smali and Baksmali	238
IDA	239
Decompiling DEX Bytecode	240
Dex2jar and JD-GUI	240
JEB	240
Decompiling Optimized DEX Bytecode	242
Reversing Native Code	244
Additional Tools	244
Apktool	244
Jadx	245
JAD	246
Dealing with ART	246
Summary	246
<b>Chapter 7 Attacking Android Applications</b>	<b>247</b>
Exposing Security Model Quirks	248
Interacting with Application Components	248
Default Export Behavior	248
Explicitly Exported	249
Implicitly Exported	249
Finding Exported Components	250
Supreme User Contexts	250
Permission Protection Levels	251
Attacking Application Components	255
A Closer Look at Intents	255
Introducing Sieve: Your First Target Application	258
Exploiting Activities	262
Unprotected Activities	262
Tapjacking	267
Recent Application Screenshots	268
Fragment Injection	269
Trust Boundaries	271
Exploiting Insecure Content Providers	272
Unprotected Content Providers	272
SQL Injection	275
File-Backed Content Providers	282
Pattern-Matching Flaws	284
Attacking Insecure Services	285
Unprotected Started Services	285
Unprotected Bound Services	286
Abusing Broadcast Receivers	295
Unprotected Broadcast Receivers	295

Intent Sniffing	299
Secret Codes	301
Accessing Storage and Logging	304
File and Folder Permissions	304
File Encryption Practices	309
SD Card Storage	310
Logging	311
Misusing Insecure Communications	312
Web Traffic Inspection	312
Finding HTTP Content	314
Finding HTTPS Content	314
SSL Validation Flaws	315
WebViews	317
Other Communication Mechanisms	322
Clipboard	322
Local Sockets	323
TCP/UDP Protocols with Other Hosts	324
Exploiting Other Vectors	326
Abusing Native Code	326
Finding Native Code	326
Attaching a Debugger	330
Exploiting Misconfigured Package Attributes	332
Application Backups	332
Debuggable Flag	334
Additional Testing Techniques	341
Patching Applications	342
Manipulating the Runtime	345
Tool: Xposed Framework	346
Tool: Cydia Substrate	346
Use Case: SSL Certificate Pinning	347
Use Case: Root Detection	349
Use Case: Runtime Monitoring	349
Summary	351
<b>Chapter 8 Identifying and Exploiting Android</b>	
<b>Implementation Issues</b>	<b>353</b>
Reviewing Pre-Installed Applications	353
Finding Powerful Applications	354
Finding Remote Attack Vectors	357
Browsers and Document Readers	357
BROWSABLE Activities	358
Custom Update Mechanisms	361
Remote Loading of Code	362
WebViews	362
Listening Services	363
Messaging Applications	363
Finding Local Vulnerabilities	364



Exploiting Devices	365
Using Attack Tools	365
Ettercap	366
Burp Suite	368
drozer	370
Explanation of Privilege Levels	374
Non-System Application without Context	374
Non-System Application with Context	375
Installed Package	375
ADB Shell Access	375
System User Access	375
Root User Access	376
Practical Physical Attacks	376
Getting ADB Shell Access	376
Bypassing Lock Screens	379
Installing a Rogue drozer Agent through ADB	386
Practical Remote Attacks	387
Remote Exploits	387
Man-in-the-Middle Exploits	401
Malware	410
Infiltrating User Data	416
Using Existing drozer Modules	416
Record Microphone	416
Read and Send SMS Messages	417
Read Contacts	417
User GPS Location	418
Capturing the User's Screen	418
Stealing Files from SD Card	420
Other Techniques for Privileged Scenarios	421
Extracting Wi-Fi Keys	421
User Accounts	421
Cracking Patterns, PINs, and Passwords	422
Reading Extended Clipboards	423
Simulating User Interaction	425
Extracting Application Data with Physical Access	426
Summary	426
<b>Chapter 9 Writing Secure Android Applications</b>	<b>427</b>
Principle of Least Exposure	427
Application Components	428
Data Storage	428
Interacting with Untrusted Sources	428
Requesting Minimal Permissions	428
Bundling Files Inside the APK	429
Essential Security Mechanisms	429
Reviewing Entry Points into Application Components	429
Securing Activities	430

Securing Content Providers	433
Securing Broadcast Receivers	435
Storing Files Securely	436
Creating Files and Folders Securely	436
Using Encryption	436
Using Random Numbers, Key Generation, and Key Storage	437
Exposing Files Securely to Other Applications	440
Creating Secure Communications	441
Internet Communications	441
Local Communications	443
Securing WebViews	443
JavaScript	444
JavaScriptInterface	444
Plug-Ins	444
Access to Information	445
Web Content Validation	445
Configuring the Android Manifest	446
Application Backups	446
Setting the Debuggable Flag	446
API Version Targeting	447
Logging	448
Reducing the Risk of Native Code	448
Advanced Security Mechanisms	450
Protection Level Downgrade Detection	450
Protecting Non-Exported Components	451
Slowing Down a Reverse Engineer	451
Obfuscation	451
Root Detection	453
Debugger Detection	454
Tamper Detection	454
Summary	455
<b>Chapter 10 Analyzing Windows Phone Applications</b>	<b>459</b>
Understanding the Security Model	460
Code Signing and Digital Rights Management (DRM)	460
Application Sandboxing	460
AppContainer	461
Chambers and Capabilities	461
Data Encryption 'At Rest'	463
Internal Storage Volume	463
Secure Digital Card Encryption	464
Windows Phone Store Submission Process	464
Exploring Exploit Mitigation Features	466
Stack Canaries	467
Address Space Layout Randomization	467
Data Execution Prevention	469

Safe Structured Exception Handling	470
Userland Heap Safe Unlinking	472
Mitigations in Kernel Space	472
Understanding Windows Phone 8.x Applications	473
Application Packages	473
Programming Languages and Types of Applications	474
Application Manifests	475
Attack Surface Enumeration	476
Application Directories	480
Distribution of Windows Phone Applications	481
Windows Phone Store	481
Store Sideloading	482
Company App Sideloading/Distribution	483
Targeted Application Distribution	483
Developer Sideloading	483
Building a Test Environment	484
SDK Tools	485
Obtaining the Development Tools	485
Visual Studio	486
Emulator	488
Developer Unlocking Your Device	489
Capability Unlocking Your Device	491
Samsung Ativ Full Capability Unlock and Filesystem	
Access on Windows Phone 8	493
Samsung Ativ Interop Unlock and Filesystem Access on	
Windows Phone 8.1 via Custom MBN	498
Huawei Ascend W1 Full Capability Unlock and	
Filesystem Access on Windows Phone 8	502
Huawei Ascend W1-U00 Full Capability Unlock and	
Filesystem Access on Windows Phone 8.1	503
Using Filesystem Access	503
Using Registry Access	505
Useful Hacking Tools	506
Analyzing Application Binaries	506
Reverse Engineering	507
Analyzing Exploit Mitigation Features	508
Summary	509
<b>Chapter 11 Attacking Windows Phone Applications</b>	<b>511</b>
Analyzing for Data Entry Points	511
WebBrowser and WebView Controls	512
Bluetooth	515
HTTP Sessions	516
Network Sockets	517
Near Field Communication	518
Barcodes	519
SD Cards	520

Interprocess Communications Interfaces	522
Protocol Handlers	523
File Extension Handlers	524
Toast Notifications	525
Attacking Transport Security	525
Identifying and Capturing Cleartext HTTP Communications	526
Identifying and Capturing HTTPS Communications	529
Capturing Non-HTTP/HTTPS Traffic	531
SSL Certificate Validation Flaws	532
Attacking WebBrowser and WebView Controls	534
Cross-Site Scripting	534
Local Scripting Attacks	536
JavaScript-C# Communication	541
Identifying Interprocess Communication Vulnerabilities	542
Protocol Handlers	542
File Handlers	546
Toast Notifications	550
Sending Arbitrary Toasts	552
Sending Toast Notifications Remotely	556
Attacking XML Parsing	560
Introducing the XDocument API	560
Entity Expansion Denial-of-Service Attacks	563
External Entity Expansion Attacks	565
Attacking Databases	568
LINQ to SQL	568
SQLite and SQLCipher	569
Attacking File Handling	573
Introduction to File Handling	573
Directory Traversal Attacks	576
Patching .NET Assemblies	578
Summary	585
<b>Chapter 12 Identifying Windows Phone Implementation Issues</b>	<b>587</b>
Identifying Insecure Application Settings Storage	588
Identifying Data Leaks	591
HTTP(S) Cookie Storage	592
HTTP(S) Caching	593
Application Logging	593
Identifying Insecure Data Storage	593
Unencrypted File Storage	594
Insecure Database Storage	596
Local Databases	597
SQLite-Based Databases	600
Insecure Random Number Generation	601
System.Random's Predictability	601
Multiple Instances of System.Random	604
System.Random Thread Safety	604

---

Insecure Cryptography and Password Use	605
Hard-Coded Cryptography Keys	605
Insecure Storage of Cryptography Keys	606
Storing Keys and Passwords in Immutable String Objects	607
Failure to Clear Cryptography Keys and Passwords from Memory	608
Insecure Key Generation	608
Insecure Random Key Generation	609
Insecure Password-Based Key Generation and Password Policy	609
Use of Weak Cryptography Algorithms, Modes, and Key Lengths	611
Data Encryption Standard (DES)	611
AES in ECB Mode	611
Other Weak Algorithms	613
Minimum Public-Private Key Length	613
Use of Static Initialization Vectors	613
Data Protection API Misuse on Windows Phone	614
Identifying Native Code Vulnerabilities	616
Stack Buffer Overflows	617
Heap Buffer Overflows	619
Other Integer-Handling Bugs	621
Integer Underflows	622
Signedness Errors	623
Format String Bugs	624
Array Indexing Errors	625
Denial-of-Service Bugs	625
Unsafe C# Code	626
Summary	626
<b>Chapter 13 Writing Secure Windows Phone Applications</b>	<b>629</b>
General Security Design Considerations	629
Storing and Encrypting Data Securely	630
Safe Encryption Ciphers and Modes	630
Key Generation and Management	630
Encrypting Files	631
Encrypting Databases	633
Windows Phone Local Databases	633
SQLite-Based Databases	634
Secure Random Number Generation	634
Securing Data in Memory and Wiping Memory	635
Avoiding SQLite Injection	636
Implementing Secure Communications	638
Using SSL/TLS	638
SSL/TLS Certificate Validation	639
Avoiding Cross-Site Scripting in WebViews and WebBrowser Components	640

Using SSL/TLS for Network Communications	640
Disabling JavaScript	640
Safe Construction of Dynamic HTML and JavaScript	641
Avoiding Local Scripting Attacks	642
Secure XML Parsing	642
Clearing Web Cache and Web Cookies	642
Clearing Cookies	643
Clearing Web Cache	643
Avoiding Native Code Bugs	644
Using Exploit Mitigation Features	644
Summary	645
<b>Chapter 14 Analyzing BlackBerry Applications</b>	<b>647</b>
Understanding BlackBerry Legacy	647
Architecture, Security, and the Simulator	648
Apps and COD Files	648
Reverse Engineering COD Files	649
Java COD Files	649
Zip COD Files	650
Java Development Environment and JVM Interface	650
App Code Signing	651
BlackBerry Mobile Data System	652
Device Event Log	652
Understanding BlackBerry 10	652
The BlackBerry 10 Platform	653
Authman and Launcher	654
Apps Packages and BAR Files	655
Native Applications	656
Cascades Applications	657
HTML5 and JavaScript Applications	658
Android Applications	658
Distributing Applications	659
PPS Objects	659
Understanding the BlackBerry 10 Security Model	660
Process Sandboxing	660
Application Capabilities	661
Code Signing	664
<client-PBDT-xxxxx.csj file>BlackBerry Balance	664
BlackBerry 10 Jailbreaking	665
Using Developer Mode	666
The BlackBerry 10 Device Simulator	667
Accessing App Data from a Device	668
Accessing BAR Files	669
Looking at Applications	670
Network Traffic Analysis and Interception	670
BAR Archives	673