The Mobile Application Hacker’s Handbook

Dominic Chell
Tyrone Erasmus
Shaun Colley
Ollie Whitehouse

WILEY
Contents

Introduction xxxi

Chapter 1 Mobile Application (In)security 1
   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

Chapter 2 Analyzing iOS Applications 17
   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

Chapter 3  Attacking iOS Applications 69
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 Cycript 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

Chapter 4 Identifying iOS Implementation Insecurities 133
  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

Chapter 5 Writing Secure iOS Applications 149
  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
<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Securing Your Application with Binary Protections</td>
<td>158</td>
</tr>
<tr>
<td>Detecting Jailbreaks</td>
<td>159</td>
</tr>
<tr>
<td>Jailbreak Artifacts</td>
<td>160</td>
</tr>
<tr>
<td>Nondefault Open Ports</td>
<td>161</td>
</tr>
<tr>
<td>Weakening of the Sandbox</td>
<td>162</td>
</tr>
<tr>
<td>Evidence of System Modifications</td>
<td>162</td>
</tr>
<tr>
<td>Securing Your Application Runtime</td>
<td>163</td>
</tr>
<tr>
<td>Tamperproofing Your Application</td>
<td>167</td>
</tr>
<tr>
<td>Implementing Anti-Debugging Protections</td>
<td>168</td>
</tr>
<tr>
<td>Obfuscating Your Application</td>
<td>169</td>
</tr>
<tr>
<td>Summary</td>
<td>170</td>
</tr>
<tr>
<td>Chapter 6 Analyzing Android Applications</td>
<td>173</td>
</tr>
<tr>
<td>Creating Your First Android Environment</td>
<td>174</td>
</tr>
<tr>
<td>Understanding Android Applications</td>
<td>179</td>
</tr>
<tr>
<td>Reviewing Android OS Basics</td>
<td>179</td>
</tr>
<tr>
<td>Getting to Know Android Packages</td>
<td>181</td>
</tr>
<tr>
<td>Observing the Structure of a Package</td>
<td>182</td>
</tr>
<tr>
<td>Installing Packages</td>
<td>183</td>
</tr>
<tr>
<td>Using Tools to Explore Android</td>
<td>185</td>
</tr>
<tr>
<td>ADB</td>
<td>185</td>
</tr>
<tr>
<td>BusyBox</td>
<td>186</td>
</tr>
<tr>
<td>Standard Android Tools</td>
<td>188</td>
</tr>
<tr>
<td>drozer</td>
<td>189</td>
</tr>
<tr>
<td>Introduction to Application Components</td>
<td>196</td>
</tr>
<tr>
<td>Defining Components</td>
<td>198</td>
</tr>
<tr>
<td>Interacting with Components</td>
<td>199</td>
</tr>
<tr>
<td>Looking Under the Hood</td>
<td>201</td>
</tr>
<tr>
<td>Installing an Application</td>
<td>201</td>
</tr>
<tr>
<td>Running an Application</td>
<td>204</td>
</tr>
<tr>
<td>Understanding the Security Model</td>
<td>206</td>
</tr>
<tr>
<td>Code Signing</td>
<td>206</td>
</tr>
<tr>
<td>Discovered Vulnerabilities</td>
<td>210</td>
</tr>
<tr>
<td>Understanding Permissions</td>
<td>212</td>
</tr>
<tr>
<td>Inspecting the Android Permission Model</td>
<td>212</td>
</tr>
<tr>
<td>Protection Levels</td>
<td>216</td>
</tr>
<tr>
<td>Application Sandbox</td>
<td>219</td>
</tr>
<tr>
<td>Filesystem Encryption</td>
<td>221</td>
</tr>
<tr>
<td>Generic Exploit Mitigation Protections</td>
<td>222</td>
</tr>
<tr>
<td>Rooting Explained</td>
<td>226</td>
</tr>
<tr>
<td>Rooting Objectives</td>
<td>226</td>
</tr>
<tr>
<td>Rooting Methods</td>
<td>228</td>
</tr>
<tr>
<td>Reverse-Engineering Applications</td>
<td>233</td>
</tr>
<tr>
<td>Retrieving APK Files</td>
<td>234</td>
</tr>
<tr>
<td>Viewing Manifests</td>
<td>235</td>
</tr>
<tr>
<td>aapt</td>
<td>235</td>
</tr>
</tbody>
</table>
Chapter 7 Attacking Android Applications

Exposing Security Model Quirks
  Interacting with Application Components
    Default Export Behavior
    Explicitly Exported
    Implicitly Exported
    Finding Exported Components
    Supreme User Contexts
    Permission Protection Levels

Attacking Application Components
  A Closer Look at Intents
  Introducing Sieve: Your First Target Application
  Exploiting Activities
    Unprotected Activities
    Tapjacking
    Recent Application Screenshots
    Fragment Injection
    Trust Boundaries
  Exploiting Insecure Content Providers
    Unprotected Content Providers
    SQL Injection
    File-Backed Content Providers
    Pattern-Matching Flaws
  Attacking Insecure Services
    Unprotected Started Services
    Unprotected Bound Services
  Abusing Broadcast Receivers
    Unprotected Broadcast Receivers
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intent Sniffing</td>
<td>299</td>
</tr>
<tr>
<td>Secret Codes</td>
<td>301</td>
</tr>
<tr>
<td>Accessing Storage and Logging</td>
<td>304</td>
</tr>
<tr>
<td>File and Folder Permissions</td>
<td>304</td>
</tr>
<tr>
<td>File Encryption Practices</td>
<td>309</td>
</tr>
<tr>
<td>SD Card Storage</td>
<td>310</td>
</tr>
<tr>
<td>Logging</td>
<td>311</td>
</tr>
<tr>
<td>Misusing Insecure Communications</td>
<td>312</td>
</tr>
<tr>
<td>Web Traffic Inspection</td>
<td>312</td>
</tr>
<tr>
<td>Finding HTTP Content</td>
<td>314</td>
</tr>
<tr>
<td>Finding HTTPS Content</td>
<td>314</td>
</tr>
<tr>
<td>SSL Validation Flaws</td>
<td>315</td>
</tr>
<tr>
<td>WebViews</td>
<td>317</td>
</tr>
<tr>
<td>Other Communication Mechanisms</td>
<td>322</td>
</tr>
<tr>
<td>Clipboard</td>
<td>322</td>
</tr>
<tr>
<td>Local Sockets</td>
<td>323</td>
</tr>
<tr>
<td>TCP/UDP Protocols with Other Hosts</td>
<td>324</td>
</tr>
<tr>
<td>Exploiting Other Vectors</td>
<td>326</td>
</tr>
<tr>
<td>Abusing Native Code</td>
<td>326</td>
</tr>
<tr>
<td>Finding Native Code</td>
<td>326</td>
</tr>
<tr>
<td>Attaching a Debugger</td>
<td>330</td>
</tr>
<tr>
<td>Exploiting Misconfigured Package Attributes</td>
<td>332</td>
</tr>
<tr>
<td>Application Backups</td>
<td>332</td>
</tr>
<tr>
<td>Debuggable Flag</td>
<td>334</td>
</tr>
<tr>
<td>Additional Testing Techniques</td>
<td>341</td>
</tr>
<tr>
<td>Patching Applications</td>
<td>342</td>
</tr>
<tr>
<td>Manipulating the Runtime</td>
<td>345</td>
</tr>
<tr>
<td>Tool: Xposed Framework</td>
<td>346</td>
</tr>
<tr>
<td>Tool: Cydia Substrate</td>
<td>346</td>
</tr>
<tr>
<td>Use Case: SSL Certificate Pinning</td>
<td>347</td>
</tr>
<tr>
<td>Use Case: Root Detection</td>
<td>349</td>
</tr>
<tr>
<td>Use Case: Runtime Monitoring</td>
<td>349</td>
</tr>
<tr>
<td>Summary</td>
<td>351</td>
</tr>
</tbody>
</table>

**Chapter 8** Identifying and Exploiting Android Implementation Issues

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reviewing Pre-Installed Applications</td>
<td>353</td>
</tr>
<tr>
<td>Finding Powerful Applications</td>
<td>354</td>
</tr>
<tr>
<td>Finding Remote Attack Vectors</td>
<td>357</td>
</tr>
<tr>
<td>Browsers and Document Readers</td>
<td>357</td>
</tr>
<tr>
<td>BROWSABLE Activities</td>
<td>358</td>
</tr>
<tr>
<td>Custom Update Mechanisms</td>
<td>361</td>
</tr>
<tr>
<td>Remote Loading of Code</td>
<td>362</td>
</tr>
<tr>
<td>WebViews</td>
<td>362</td>
</tr>
<tr>
<td>Listening Services</td>
<td>363</td>
</tr>
<tr>
<td>Messaging Applications</td>
<td>363</td>
</tr>
<tr>
<td>Finding Local Vulnerabilities</td>
<td>364</td>
</tr>
</tbody>
</table>
Exploiting Devices
Using Attack Tools
  Ettercap
  Burp Suite
drozer
Explanation of Privilege Levels
  Non-System Application without Context
  Non-System Application with Context
  Installed Package
  ADB Shell Access
  System User Access
  Root User Access
Practical Physical Attacks
  Getting ADB Shell Access
  Bypassing Lock Screens
  Installing a Rogue drozer Agent through ADB
Practical Remote Attacks
  Remote Exploits
  Man-in-the-Middle Exploits
  Malware
Infiltrating User Data
  Using Existing drozer Modules
    Record Microphone
    Read and Send SMS Messages
    Read Contacts
    User GPS Location
    Capturing the User's Screen
    Stealing Files from SD Card
Other Techniques for Privileged Scenarios
  Extracting Wi-Fi Keys
  User Accounts
  Cracking Patterns, PINs, and Passwords
  Reading Extended Clipboards
  Simulating User Interaction
  Extracting Application Data with Physical Access
Summary

Chapter 9 Writing Secure Android Applications
Principle of Least Exposure
  Application Components
  Data Storage
  Interacting with Untrusted Sources
  Requesting Minimal Permissions
  Bundling Files Inside the APK
Essential Security Mechanisms
  Reviewing Entry Points into Application Components
  Securing Activities
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Securing Content Providers</td>
<td>433</td>
</tr>
<tr>
<td>Securing Broadcast Receivers</td>
<td>435</td>
</tr>
<tr>
<td>Storing Files Securely</td>
<td>436</td>
</tr>
<tr>
<td>Creating Files and Folders Securely</td>
<td>436</td>
</tr>
<tr>
<td>Using Encryption</td>
<td>436</td>
</tr>
<tr>
<td>Using Random Numbers, Key Generation, and Key Storage</td>
<td>437</td>
</tr>
<tr>
<td>Exposing Files Securely to Other Applications</td>
<td>440</td>
</tr>
<tr>
<td>Creating Secure Communications</td>
<td>441</td>
</tr>
<tr>
<td>Internet Communications</td>
<td>441</td>
</tr>
<tr>
<td>Local Communications</td>
<td>443</td>
</tr>
<tr>
<td>Securing WebViews</td>
<td>443</td>
</tr>
<tr>
<td>JavaScript</td>
<td>444</td>
</tr>
<tr>
<td>JavaScript Interface</td>
<td>444</td>
</tr>
<tr>
<td>Plug-Ins</td>
<td>444</td>
</tr>
<tr>
<td>Access to Information</td>
<td>445</td>
</tr>
<tr>
<td>Web Content Validation</td>
<td>445</td>
</tr>
<tr>
<td>Configuring the Android Manifest</td>
<td>446</td>
</tr>
<tr>
<td>Application Backups</td>
<td>446</td>
</tr>
<tr>
<td>Setting the Debuggable Flag</td>
<td>446</td>
</tr>
<tr>
<td>API Version Targeting</td>
<td>447</td>
</tr>
<tr>
<td>Logging</td>
<td>448</td>
</tr>
<tr>
<td>Reducing the Risk of Native Code</td>
<td>448</td>
</tr>
<tr>
<td>Advanced Security Mechanisms</td>
<td>450</td>
</tr>
<tr>
<td>Protection Level Downgrade Detection</td>
<td>450</td>
</tr>
<tr>
<td>Protecting Non-Exported Components</td>
<td>451</td>
</tr>
<tr>
<td>Slowing Down a Reverse Engineer</td>
<td>451</td>
</tr>
<tr>
<td>Obfuscation</td>
<td>451</td>
</tr>
<tr>
<td>Root Detection</td>
<td>453</td>
</tr>
<tr>
<td>Debugger Detection</td>
<td>454</td>
</tr>
<tr>
<td>Tamper Detection</td>
<td>454</td>
</tr>
<tr>
<td>Summary</td>
<td>455</td>
</tr>
</tbody>
</table>

**Chapter 10 Analyzing Windows Phone Applications**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understanding the Security Model</td>
<td>460</td>
</tr>
<tr>
<td>Code Signing and Digital Rights Management (DRM)</td>
<td>460</td>
</tr>
<tr>
<td>Application Sandboxing</td>
<td>460</td>
</tr>
<tr>
<td></td>
<td>461</td>
</tr>
<tr>
<td>Data Encryption ‘At Rest’</td>
<td>463</td>
</tr>
<tr>
<td>Internal Storage Volume</td>
<td>463</td>
</tr>
<tr>
<td>Secure Digital Card Encryption</td>
<td>464</td>
</tr>
<tr>
<td>Windows Phone Store Submission Process</td>
<td>464</td>
</tr>
<tr>
<td>Exploring Exploit Mitigation Features</td>
<td>466</td>
</tr>
<tr>
<td>Stack Canaries</td>
<td>467</td>
</tr>
<tr>
<td>Address Space Layout Randomization</td>
<td>467</td>
</tr>
<tr>
<td>Data Execution Prevention</td>
<td>469</td>
</tr>
</tbody>
</table>
Contents

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

Chapter 11 Attacking Windows Phone Applications 511
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

Chapter 12 Identifying Windows Phone Implementation Issues 587
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
**Chapter 13  Writing Secure Windows Phone Applications**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Security Design Considerations</td>
<td>629</td>
</tr>
<tr>
<td>Storing and Encrypting Data Securely</td>
<td>630</td>
</tr>
<tr>
<td>Safe Encryption Ciphers and Modes</td>
<td>630</td>
</tr>
<tr>
<td>Key Generation and Management</td>
<td>630</td>
</tr>
<tr>
<td>Encrypting Files</td>
<td>631</td>
</tr>
<tr>
<td>Encrypting Databases</td>
<td>633</td>
</tr>
<tr>
<td>Windows Phone Local Databases</td>
<td>633</td>
</tr>
<tr>
<td>SQLite-Based Databases</td>
<td>634</td>
</tr>
<tr>
<td>Secure Random Number Generation</td>
<td>634</td>
</tr>
<tr>
<td>Securing Data in Memory and Wiping Memory</td>
<td>635</td>
</tr>
<tr>
<td>Avoiding SQLite Injection</td>
<td>636</td>
</tr>
<tr>
<td>Implementing Secure Communications</td>
<td>638</td>
</tr>
<tr>
<td>Using SSL/TLS</td>
<td>638</td>
</tr>
<tr>
<td>SSL/TLS Certificate Validation</td>
<td>639</td>
</tr>
<tr>
<td>Avoiding Cross-Site Scripting in WebView and WebBrowser Components</td>
<td>640</td>
</tr>
</tbody>
</table>
Chapter 14 Analyzing BlackBerry Applications 647

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
Randomness Sources 712
SSL, TLS Certificate, and Public Key Pinning in OS 7x
and Earlier Legacy Java Applications 713
Defending Against Module Squatting 715
Obfuscation 716
Blackberry WebWorks Security on BlackBerry OS 7
or Lower 716
Securing BlackBerry 10 Native Applications 716
General C/C++ Secure Development Principals 717
Compiler and Linker Defenses 717
Memory Cleaning 718
File Access Control 718
File Encryption 719
Randomness Sources 720
SSL, TLS Certificate, and Public Key Pinning in
Blackberry 10 Native Applications 720
Security Builder Encryption API 720
Heap Robustness Against Corruption 720
QNX Native IPC Mechanism Security Considerations 721
Headless App Interprocess Communication 722
Securing BlackBerry 10 Cascades Applications 723
Securing BlackBerry 10 HTML5 and JavaScript
(WebWorks) Applications 724
App Invocation Parameters 724
Access App Configuration Option 724
Websecurity App Configuration Option 725
Content Injection Mitigations 726
Securing Android Applications on BlackBerry 10 726
Summary 726

Chapter 18 Cross-Platform Mobile Applications 729
Introduction to Cross-Platform Mobile Applications 729
Bridging Native Functionality 731
Exposing Native Functionality on Android 732
Exposing Native Functionality on iOS 733
Exposing Native Functionality on Windows Phone 734
Exposing Native Functionality on BlackBerry 735
Exploring PhoneGap and Apache Cordova 736
Standard Features of PhoneGap 736
PhoneGap and Cordova Security 737
Cross-Application and Cross-Site Scripting Attacks 738
Understanding Domain Whitelisting 739
Iframes and Callbacks 740
Encrypted Storage 741
Summary 741

Index 743