The Mobile Application Hacker's Handbook

Dominic Chell Tyrone Erasmus
Shaun Colley Ollie Whitehouse

WILEY

Contents

List of Tables

List of Illustrations

Introduction

Mobile computing has changed the game. Your personal data is no longer just stored on your desktop in the sanctuary of your office or home. You now carry personally identifiable information, financial data, personal and corporate email, and much more in your pocket, wherever you go. The smartphone is quickly becoming ubiquitous, and with at least 40 applications installed on the average smartphone the attack surface is significant.

Smartphones have become commonplace not only in the consumer markets but also now in the enterprise. Enterprise mobile applications extend the corporate environment beyond the workplace, introducing new security concerns and exposing organizations to new types of threats. Enterprises embracing “Bring Your Own Device” (BYOD) strategies should be particularly mindful of the array of applications that the smartphone may have installed and run within the corporate network.

This book is a practical guide to reviewing the security of mobile applications on the most widely adopted mobile operating systems: Apple iOS, Google Android, BlackBerry, and Windows Mobile. It focuses solely on the client-side, examining mobile applications in the context of these devices as opposed to server-side applications, where security is much more mature and better understood.

Overview of This Book

The focus of this book is highly practical. Although we provide some background theory for you to understand the fundamentals of mobile application vulnerabilities, our primary concern is documenting the techniques you need to master to attack and exploit them. Where applicable, we include real-world examples derived from our many years of experience and from publically documented vulnerabilities.

In addition to describing mobile application security vulnerabilities and attack techniques, we describe in detail the defense-in-depth strategies and countermeasures that application developers can use to effectively defend their applications. This information enables penetration testers, security consultants, and developers alike to provide high-quality remediation advice to application owners.

In short, this book is intended to act as an all-encompassing single point of reference for mobile application security, bringing together the publicly available knowledge on the attack and defense of mobile applications and combining it with the blended experience of the authors.

How This Book Is Organized

This book is roughly split into the topics covered for each of the mobile device platforms, you can think of it as four books in one! For each of the mobile platforms; we provide a pragmatic approach to performing a mobile application security assessment. First detailing the necessary background information on how to analyze the application itself, followed by detailed information on how to attack the application and the categories of vulnerability that affect the relevant platform, finally providing remedial action that can be implemented to develop secure mobile applications. If you are new to mobile application security, it is recommended that you read the book from start to finish, acquiring the knowledge and understanding to tackle later chapters. This can be applied to the relevant chapters for each mobile platform, or the entirety of the book. If you're only interested in one specific platform or only a specific area of a platform, you can jump straight into the subsection that interests you. Where applicable, we have included cross-references to other chapters, which can be used to fill any gaps in your understanding.

- Chapter 1, “Mobile Application (In) Security,” describes the current state of security in mobile applications today. As an area that has seen explosive and rapid growth over the past few years, security has been frequently overlooked or misunderstood in the fast evolving software lifecycles. As a consequence, mobile application vulnerabilities are rife and commonplace in the application ecosystem. This chapter examines the key attack surfaces for mobile applications, how mobile security has evolved and what standards and frameworks exist that can be used to categorize mobile application vulnerabilities. It then provides an overview of some mobile security resources that may prove useful in developing your assessment skills. Finally, it provides an insight into how mobile application security is, in our opinion, likely to evolve in the future.

- Chapter 2, “Analyzing iOS Applications,” is the first chapter to focus on iOS application assessment. It starts off by describing some foundational knowledge on the security features of the iOS platform and briefly touches on how they have been circumvented in the past through jailbreaking. Although jailbreaking weakens the security controls of the device, it provides the opportunity to gain interactive access to the operating system, which is essential to thoroughly assess the security of an iOS application. This chapter describes how to access the device, and the file system as well as important concepts such as the Data Protection API and Keychain. This chapter also describes a range of further interesting topics, including App Store encryption, reverse engineering of iOS binaries, generic exploit, and mitigation features.
- Chapter 3, “Attacking iOS Applications,” describes in detail the offensive techniques that can be used to attack iOS applications. It provides a brief introduction to Objective-C and Swift, the languages in which iOS applications are developed, and then outlines how the Swift and Objective-C runtimes can be manipulated to access and control the internals of an application. We then go on to describe the various types of client-side injection attacks that iOS applications can be susceptible to, including SQL injection, XML injection, and XML External Entity injection. It also dives into how data can be transmitted between applications on the same device through Inter Process Communication and how insecurities can arise that leave an application at risk of attack.

- Chapter 4, “Identifying iOS Implementation Issues,” contains information related to how implementation issues specific to the iOS platform can leave applications at risk. This chapter describes how iOS applications can be audited for vulnerabilities arising from improper use of the device's address book, geolocation frameworks, and logging system. We also examine iOS specific peculiarities that can leave residual data on a device and may expose sensitive content, including caching of snapshots, web view data, and pasteboards. Finally, the chapter concludes with an overview of the memory corruption issues that affect iOS applications and how and to what extent these can be exploited.
- Chapter 5, “Writing Secure iOS Applications,” transitions from the attacker’s perspective to that of the defender. In this chapter, we examine the techniques that developers can use in their applications to protect against manipulation. This chapter also serves as a reference point for professional security assessors who need to offer remedial advice following application assessments. We describe how to securely implement encryption, erase data from both memory and the file system, and embed binary protections such as tamper proofing, jailbreaking, and runtime validation.
- Chapter 6, “Analyzing Android Applications,” is the first section in a series of chapters on the Google Android platform. It starts by providing the necessary background on the security features of the platform, including code signing, sandboxing and a detailed description of the permission model. With the basics covered, we go on to examine how Android devices can be rooted to provide interactive super user access to the device. We also examine how Android applications are packaged, loaded onto devices, and some of the tools that can be used to build a test environment. The

chapter concludes by describing the different ways packages are compiled and how security assessments can be conducted by decompiling and examining the application packages.

- Chapter 7, “Attacking Android Applications,” provides a detailed description of the common areas of vulnerability in Android applications, along with the techniques to attack and exploit them. This chapter delves into many Android-specific attack categories, including exploitation of insecure services, content providers, broadcasts, intents, and activities. The chapter also examines how the Android runtime can be manipulated, exploring the various frameworks that can be used to implement function hooking in the Java Virtual Machine with sample use cases and practical examples. We also address perhaps two of the most important areas in mobile security, file system storage, and network communications. We explore how file and folder permissions can be exploited to leak sensitive information, how poor cryptographic practices can undermine secure storage, and how poorly implemented network access can be exploited from public or insecure networks. Finally, this chapter concludes with an insight into JavaScript interfaces, an area that has come under close scrutiny in 2014, and one that has exposed a significant number of Android devices to remote compromise.
- Chapter 8, “Identifying Android Implementation Issues,” teaches you how to become an Android hacker. It provides practical advice on how to identify vulnerabilities in OEM device applications, how to find and exploit powerful packages, and how to leverage privilege escalations to compromise other applications or, in some circumstances, the device itself. We also examine how to exploit applications from the network,

with insecurities in URI handlers, JavaScript bridges, handling of SSL certificates, and custom update mechanisms. This chapter also explores how to use Drozer, the Android attack tool, to gain access to a device, including chaining of remote and local exploits and the post exploitation activities that can be performed.

- Chapter 9, "Writing Secure Android Applications," concludes the series of Android chapters and, similarly to the iOS counterpart, provides a basis for which defensive advice can be offered. We provide security professionals and developers detailed instructions on how to correctly implement encryption, perform root detection, and protect intellectual property by obfuscating code. At the end of the chapter, an application checklist is provided that can be used as a reference point when auditing an Android application.
- Chapter 10, "Analyzing Windows Phone Applications," details the essential "need to know" knowledge for the Windows Phone (WP8) platform and application ecosystem. In this section, we examine the fundamental security protections that are employed by the platform, including exploit mitigation features and application capabilities. We then explain the inner workings of WP8 applications, how to develop, build, compile, and run them along with the essential toolkit needed to set up a test environment. We conclude with an analysis of the Windows Data Protection API (DPAPI) and how misconfigurations in the protection flags can leave application content at risk.
- Chapter 11, "Attacking Windows Phone Applications," provides an in-depth analysis of the common insecurities that occur with WP8 applications. It covers perhaps the most important and relevant topics that you will need to

learn in order to hack a Windows Phone application. This chapter examines and explains transport security in WP8 applications, how to intercept network communications, and how to bypass protection mechanisms such as certificate pinning. We also delve into reverse engineering of WP8 applications, including both native and managed code components and how information gained from this allows you to manipulate application behavior by patching application code. An important skill for professional security assessors reviewing mobile applications is the ability to identify the key data entry points in an application. This chapter explains how to analyze WP8 applications to identify data entry points, and how when tainted data enters an application it can lead to serious security vulnerabilities. Having identified the various entry points that can exist, we explore and examine the various injection attacks that can be exploited, including SQL injection, injection into web browser controls, XML-based injection, and injection into file handling routines.

- Chapter 12, “Identifying Windows Phone Implementation Issues,” deals with the common issues that arise through insecurely implemented WP8 applications. In particular, we focus on insecurities that arise through handling of log data, lack of protections on the clipboard, caching in keyboard and web browser controls, and geo-location leakages. This chapter provides security professionals and developers with the required knowledge to audit WP8 applications for not only the misuse of the platform APIs but also how to identify memory corruption issues. We examine the various types of memory corruption that can occur in WP8 applications, including the implications of traditional corruption bugs, read access violations, information leaks, and issues that arise in managed c# code.

- Chapter 13, “Writing Secure Windows Phone Applications,” like its counterparts on iOS and Android, details the necessary information about to develop secure WP8 applications. It covers the fundamental practices that application developers should be including in WP8 applications. If you're only looking for remediation and hardening advice, feel free to jump straight into this chapter. This chapter also examines how to securely implement encryption, securely erase data from both memory and the file system, and how to implement binary protections. We provide in-depth analysis on anti-tamper implementations, available compiler protections, and WP8 application obfuscation, none of which are widely documented in the public domain.
- Chapter 14, “Analyzing BlackBerry Applications,” is the backbone of the BlackBerry section, and provides the foundational knowledge needed to understand the different types of BlackBerry applications that exist and how they are developed and distributed. We also examine the BlackBerry platform itself, providing an in-depth evaluation of the core platform security features, including sandboxing, data-at-rest encryption, and process-level sandboxing. This chapter also details how to build a test environment using the simulator and developer mode, with some analysis of the Dingleberry jailbreak exploit. We explain how to access the device, where content can be found and the various files and file types that you will encounter when exploring your BlackBerry. We then conclude by discussing the Security Builder API, how and when transport insecurities occur, how certificate pinning works, and some of the strategies that can be used to bypass it.

- Chapter 15, "Attacking BlackBerry Applications," provides some much needed insight into the world of BlackBerry application security. In this chapter we discuss how the application runtime functions, including important subjects such as the System API and the various programming frameworks that BlackBerry applications take advantage of. We then examine the Inter-Process Communication (IPC) mechanisms that exist, how BlackBerry 10 applications differ from previous implementations, and detail how insecurely implemented IPC can be exploited by other applications on the device.
- Chapter 16, "Identifying BlackBerry Application Implementation Issues," discuses the common issues that arise in BlackBerry applications due to misuse of BlackBerry APIs. This chapter may be of particular interest to developers, and investigates the various types of information leakages that an application can be susceptible to with a particular focus on Personally Identifiable Information. Topics that are also explored are system logging and a brief review of memory corruption vulnerabilities that affect BB10 applications.
- Chapter 17, "Writing Secure BlackBerry Applications," is of particular relevance to application developers. This chapter pulls together some of the techniques that can be used to improve the security of BlackBerry applications. We discuss strategies for performing secure deletion of data, both in memory and from the filesystem, and how to securely implement encryption. Where applicable, we provide practical examples using both built-in APIs and custom developed functions.
- Chapter 18, "Cross Platform Applications," examines a growing trend in mobile development and cross-platform mobile applications. We explore the various

implementations that currently exist, and provide a breakdown of the functionality that they offer. We then detail the various vulnerability categories that affect cross-platform applications, with practical examples on how to exploit these to perform malicious actions in Apache Cordova.

Who Should Read This Book

This book's primary audience is anyone who has a personal or professional interest in attacking mobile applications. It also caters to anyone responsible for the development of mobile applications. This book not only provides a detailed analysis of how to attack and secure iOS, Android, BlackBerry, and Windows Phone applications, but also serves as a reference point for generic mobile application security regardless of operating platform.

In the course of illustrating many categories of security flaws, we provide code extracts showing how applications can be vulnerable. These examples are simple enough that you can understand them without any prior knowledge of the language in question. But they are most useful if you have some basic experience with reading or writing code.

Tools You Will Need

This book is strongly geared toward hands-on practical techniques that you can use to attack mobile applications. After reading this book you will understand the different types of vulnerabilities that affect mobile applications and have the practical knowledge to attack and exploit them. The emphasis of the book is on practical and human-driven exploitation as opposed to running automated tools on the target application.

That said, you will find several tools useful, and sometimes indispensable, when performing the tasks and techniques we describe. All of these are available on the Internet. We recommend that you download and experiment with each tool as you read about it.

While in most cases it is possible to follow the practical examples in a simulated or emulated environment, there is no substitute for running an application on a physical device. Therefore, we would recommend that, where possible, the examples be followed on a real device.

What's on the Website

The companion website for this book at www.mobileapphacker.com, which you can also link to from `www.wiley.com/go/mobileapplicationhackers`, contains several resources that you will find useful in the course of mastering the techniques we describe and using them to attack actual applications. In particular, the website contains access to the following:

- Source code for some of the scripts we present in the book
- A list of current links to all the tools and other resources discussed in the book
- A handy checklist of the tasks involved in attacking a typical application
- Answers to the questions posed at the end of each chapter

CHAPTER 1
Mobile Application (In)security

There is little doubt that mobile computing has changed the world; in particular, the way you work, interact, and socialize will never be the same again. It has brought infinite possibilities to your fingertips, available all the time. The ability to do your online banking, check your e-mail, play the stock market and much, much more are just a swipe away. Indeed, application development is now so popular that Apple's trademark, "There's an app for that" is bordering on reality.

This chapter takes a look how mobile applications have evolved and the benefits that they provide. It presents some metrics about the fundamental vulnerabilities that affect mobile applications, drawn directly from our experience, demonstrating that the vast majority of mobile applications are far from secure. We then examine a means to categorize these vulnerabilities based on the Open Web Application Security Project (OWASP) Top 10 mobile security risks. We also provide a high-level overview of some of the open source mobile security tools endorsed by OWASP, how you can use them to identify some of the issues detailed in the project, and where to find them. Finally, we describe the latest trends in mobile application security and how we expect this area to develop in the future.

The Evolution of Mobile Applications

The first mobile phone applications were developed by handset manufacturers; documentation was sparse, and little information existed in the public domain on the