Masterarbeit

Weiterentwicklung von Methoden und Werkzeugen zum Pentest b. mob. Applikationen

zur Erlangung des akademisches Grades eines
Master of Science

angefertigt von
Dominik Gunther Florian Schlecht
Matrikelnummer: 00032209

Betreuer

Erstprüfer:	Prof. Hahndel
Zweitprüfer:	Prof. von Koch
Allianz Deutschland AG:	Herr Muncan und Herr Gerhager

Fakultät:	Elektrotechnik und Informatik
Studiengang:	Informatik
Schwerpunkt:	Security & Safety
Abgabedatum:	01. April 2016

Ingolstadt, 11. April 2016

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Masterarbeit bis auf die offizielle Betreuung durch die Betreuer selbstständig und ohne fremde Hilfe verfasst habe.
Die verwendeten Quellen sowie die verwendeten Hilfsmittel sind vollständig angegeben.
Wörtlich übernommene Textteile und übernommene Bilder und Zeichnungen sind in jedem Einzelfall kenntlich gemacht.

Ingolstadt, 11. April 2016

Inhaltsverzeichnis

1 Einleitung

2 Arten von Penetrationstests

2.1 Web-Application

2.2 Wireless

2.3 Social-Engineering

2.4 Mobile-Applications

2.5 Blackbox/Whitebox

3 Prozesse zu Penetrationstests

3.1 Vorbereitung

3.1.1 Aufwandsschätzung

3.1.2 Rechtliche Aspekte

NDA

Haftungsausschluss

Absicherung gegen ğ203STGB

3.1.3 Technische Aspekte

Infrastruktur

Tools

3.2 Durchführung

3.2.1 Bewertung von Findings

CVSS

Alternative Modelle

3.2.2 Dokumentation

3.2.3 Reporting

3.3 Nachbereitung/Report

3.3.1 Inhalte

3.3.2 Vorlage

4 Penetrationstests mobiler Anwendungen

4.1 Aktuelle Situation und Vergleich

4.1.1 IOS

Emulation vs. Hardware

XCode

Debugging

XCode

4.1.2 Windows-Phone

Emulation vs. Hardware

VS

Debugging

VS

4.1.3 Android

Android ist ein Ursprünglich 2003 von der Android, Inc. entwickeltes mobiles Betriebssystem. 2005 wurde es durch Google übernommen und wird seit dem weiterentwickelt. 2015 liegt es bei einem Marktanteil von TODO %. Aufgrund der Quelloffenheit des Systems wird von vielen Herstellern auf verschiedensten Plattformen genutzt. Jedoch bringt die weitführende Fragmentierung des Betriebssystems auch Nachteile mit sich. So sind in 2015 nur TODO % der Android-Devices auf einer aktuellen Version.[2]

Android-Studio und SDK

Das Android-Studio ist eine umfassende IDE. Sie ermöglicht unter anderem das schnelle Entwickeln und Testen von Apps, sowie die Emulation von beliebigen Android-Versionen. AuSSerdem ist Android Studio kostenlos, Open-Source und für Linux, Mac und Windows erhältlich. Die aktuelle Version kann unter `http://developer.android.com/sdk/index.html` heruntergeladen werden. Die Installation unter Linux ist vergleichsweise einfach, da nur ein Archiv über das Kommando

```
1 unzip android-studio-ide-143.2739321-linux.zip
```

entpackt werden muss. Für alle anderen Betriebssysteme werden entsprechende Installationsroutinen zur Verfügung gestellt. AnschlieSSend kann die IDE über die Datei „bin/studio.sh“ gestartet werden. Neben dem Android-Studio gibt es noch das Android SDK, welches über die gleiche URL heruntergeladen werde kann. Es enthält wichtige Kommandozeilen-Tools wie *adb*, *fastboot* oder *logcat*, auf welche im weiteren Verlauf noch detailliert eingegangen wird.

Compatibility Testing Suite

[2] Seite 18

Emulation vs. Hardware

Android SDK

Debugging

Android Debug Bridge[1]

Logcat

Android Debug Bridge[1]

4.2 Anforderungen

- Automatisierung
- Blackbox/Whitebox
- Reporting
- False-Positive-Rate

4.3 Laboraufbau

4.4 Entwicklung der Umgebung

4.4.1 Aufbau

4.4.2 Schnittstellen

4.4.3 Technisches Detail 1

4.4.4 Technisches Detail 2

4.5 Abgleich mit Anforderungen

5 Anwendung der Umgebung

5.1 Pentest Anwendung 1

5.2 Pentest Anwendung 2

6 Fazit

Literatur

[1] *Android Debug Bridge*. URL: http://developer.android.com/tools/help/adb.html (besucht am 22. 02. 2016).

[2] Joshua J. Drake u. a. *Android Hacker's Handbook*. John Wiley & Sons, Inc., Indianaposil, Indiana, 2014.