App-Security in iOS und Android

Seminararbeit

0.1

Hochschule für angewandte Wissenschaften
Würzburg-Schweinfurt

David Artmann
Kristoffer Schneider

20. Mai 2015

Inhaltsverzeichnis

Abbildungsverzeichnis

1 Vorwort

Die aus Cupertino in Kalifornien stammende US-Amerikanische Apple Corporation ist eine der größten Firmen der Welt und hatte einen Umsatz von 182 Mrd. USD im Geschäftsjahr 2014. Sie ist ebenfalls der Erfinder des mobilen Betriebssystems iOS, welches auf den firmeneigenen Geräten iPad, iPad mini, iPhone, iPod touch und dem Apple TV ab der zweiten Generation zum Einsatz kommt. Der Kern des Betriebssystems basiert auf dem freien UNIX Betriebssystem Darwin, das auch als Vorlage für das Betriebssystem OS X genutzt wurde.
Im Februar 2015 hatte iOS in den USA einen Marktanteil von 38,8% und 17,4% in Deutschland. [1] Es existieren mehr als einhundert Millionen iPhones auf der ganzen Welt. Das Smartphone ob iOS, Android oder Windows Phone als Betriebsystem, ist in unserer heutigen Welt nicht mehr wegzudenken und mehrere Studien haben offen gelegt, dass das Smartphone den Computer bzw. Laptop - zumindest bei der Generation unter 18 Jahren - schlägt. [2] [3]
Welche Auswirkung hätte es, wenn Sicherheitslücken auf diesen zu finden wären, oder schlimmer noch, diese völlig ohne das Wissen des Nutzers ausgenutzt werden könnten? Mit dieser Arbeit möchte ich besonders auf Sicherheitstechnische Problematiken dieses Betriebssystems eingehen und zeigen, dass ein Proprietäres Betriebssystem viele Vorteile hat, aber ebenso auch Nachteile besitzt die es nicht zu verachten gilt. Ebenso werde ich versuchen mit praktischen Beispielen zu zeigen, dass iOS - obwohl proprietär - dennoch angreifbar ist und Lücken aufweist.

[1] http://www.kantarworldpanel.com/global/smartphone-os-market-share/
[2] http://www.bitkom.org/files/documents/BITKOM_PK_Kinder_und_Jugend_3_0.pdf
[3] http://www.mpfs.de/fileadmin/JIM-pdf13/JIMStudie2013.pdf

2 Das Android Betriebssystem

Das Unternehmen Android wurde 2003 von Andy Rubin gegründet und wurde 2005 von Google aufgekauft. Seitdem kümmert sich Google und das Android Open Source Project (AOSP) um die Weiterentwicklung des Systems. Aktuell ist Android, mit 55.6% Marktanteil[4], das vorherrschende Betriebssystem für mobile Endgeräte in den USA.

Basis fÃ¼r das Betriebssystem ist ein modifizierter Linux-Kernel und eine Java Virtual Machine (JVM). Bis einschliesslich Version 4.4 wurde hierfür die Dalivk Runtime und fÃ¼r alle neueren Versionen die Android Runtime (ART) verwendet. Jede App läuft in einer eigenen Instanz der entsprechenden Runtime und damit in einer Sandbox. Oberhalb der JVM sind die meisten Komponenten in Java implementiert.

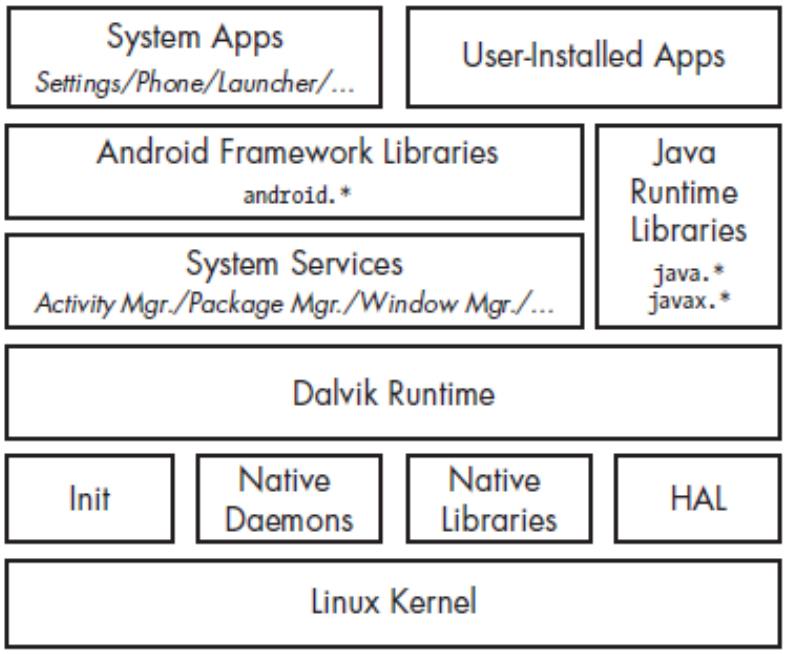

Abbildung 1: Die Architektur von Android ASI-P-2

[4]Kantar Worldpanel: Smartphone OS sales market share, http://www.kantarworldpanel.com/global/smartphone-os-market-share/, 17.5.2015

3 Apple's Law Enforcement Process Guidelines

Die Firma Apple schreibt auf ihrer Webseite:

> Our commitment to customer privacy doesn't stop because of a government information request.

Weiterhin wird beteuert:

> In addition, Apple has never worked with any government agency from any country to create a back door in any of our products or services.

Apple beteuert hier, seine Verpflichtung zur Einhaltung der Privatssphäre des Kunden auch nicht einzustellen, wenn die Regierung um Auskunft genau dieser Daten bittet. Zusätzlich wird versichert, dass Apple niemals mit Regierungsbehörden jedweder Lander gearbeitet hat, um Trojaner in eines ihrer Produkte oder Dienstleistungen einzubauen.

Eine Einhaltung dieser Richtlinien ist von Kundenseite natürlich gewünscht. Ob Apple hier wirklich Wort hält, werde ich auf den kommenden Seiten genauer beleuchten.

4 Grundlegender Aufbau einer Android App

> Android apps are written in the Java programming language. The Android SDK tools compile your code - along with any data and resource files - into an APK: an *Android package*, which is an archive file with an .apk suffix. One APK file contains all the contents of an Android app and is the file that Android-powered devices use to install the app.
> (Android: Application Fundamentals,
> http://developer.android.com/guide/components/fundamentals.html, 20.5.2015)

Eine App besteht im Kern aus zwei Teilen. Den eigentlichen Programmkomponenten und einer Manifest Datei.
Als Programmkomponenten können unter anderem vorkommen:

- Activities - stellen die Benutzeroberfläche dar
- Services - kann im Hintergrund laufen, auch wenn die App minimiert ist
- Content Provider - stellt Daten für die eigene und evtl für andere Apps zur Verfügung
- Broadcast Receiver - um Systemweite Benachrichtigungen zu empfangen (z.B dass ein Download beendet wurde)

In der Manifest-Datei werden Eigenschaften der App definiert. Darunter zählen beispielsweisse:

- Name der App
- Ziel SDK-Versionen
- Versionsnummer
- optional eine UserId (siehe 6.1)
- Permissions (siehe 6.2)

Desweiteren muss jede App signiert werden. Das hierfür benötigte Zertifikat kann sich jeder selbst generieren und muss nicht durch einen Certification Authority (CA) beglaubigt werden. Dabei wird angeraten, dass ein Entwickler für all seine Apps dasselbe Zertifikat nutzt.

5 Sicherheitsarchitektur iOS

Sicherheitsarchitektur...

5.1 Historisches

Historisches...

6 Sicherheitsaspekte der Android-Architektur

Bereits durch die Architektur des Betriebssystems, insbesondere durch die restriktive Rechtevergabe und das Sandboxing, wird versucht ein mÃ¶glichst sicheres System bereitzustellen.

6.1 Basis Rechtesystem

Von Linux wurde auch das Basis-Rechtesystem übernommen. Hierbei bekommt jede App eine eindeutige User-ID (UID) zugewiesen, welche im Normalfall zur Installationszeit zugeteilt wird. Jeder Nutzer, und somit auch jede App, arbeitet grundsätzlich erst einmal nur innerhalb der ihm zugewiesenen virtuellen Maschine und dem damit verbundenen Dateisystem.

Da es dennoch in vielen Fällen nötig ist Daten zwischen verschiedenen Apps auszutauschen, gibt es mehrer Möglichkeiten dies zu tun. Die üblichen Wege wären Intends oder SharedPreferences. Zusätzlich gibt es noch die Möglichkeit mehreren Apps dieselbe UID zuweisen zu lassen. Dies ist allerdings nur möglich wenn die entsprechenden Applikationen mit dem selben Zertifikat signiert wurden und in deren Manifest Datei eine gemeinsame UID festgelegt wurde. Durch dieses Rechtesystem wird versucht sicherzustellen, dass kein Nutzerprogramm als *root* ausgefÃ$\frac{1}{4}$hrt wird.

6.2 Sandboxing und Permissions

7 Geheime Dienste

Apple kann SMS, Fotos, Videos, Kontakte, Musik, Aufnahmen und Anruferhistorie aus passcode geschützten Geräten auslesen. Möglich machen dies nicht dokumentierte Dienste, welche auf jedem Gerät mit iOS installiert sind. In diesem Kapitel will ich auf diese Dienste eingehen und deren genauen Einsatzzweck erläutern.

7.1 lockdownd - remote access

Der Dienst *lockdownd* ermölicht den Zugriff auf ein iOS Gerät per TCP oder USB-Anschluss auf Port 62078.