Mobile Devices in Unternehmen mit erhöhtem Sicherheitsbedarf

Stefan Schauer[1] · Christian Kollmitzer[1] · Oliver Maurhart[1]
Peter Schartner[2] · Stefan Rass[2]

[1]Austrian Institute of Technology – Digital Safety & Security Department
{stefan.schauer | christian.kollmitzer | oliver.maurhart}@ait.ac.at

[2]Alpen-Adria-Universität Klagenfurt – Institute of Applied Informatics
System Security Group
{peter.schartner | stefan.rass}@aau.at

Zusammenfassung

In diesem Artikel betrachten wir die Anforderungen an die Verwendung von mobilen Endgeräten wie Smartphones oder Tablets in einem Unternehmen mit erhöhtem Sicherheitsbedarf. Der Schwerpunkt der Betrachtung liegt dabei vorrangig auf der sicheren Speicherung und der Übertragung von Daten zwischen bzw. auf mobile(n) Endgeräten (Smartphones, etc.). Diesen Anforderungen werden die Möglichkeiten für vertrauliche Kommunikation und Datenspeicherung gegenübergestellt, welche aktuelle mobile Plattformen (Android, Apple iOS, BlackBerry und Windows Phone) derzeit nativ, also ohne Modifikationen oder Erweiterungen, bieten. Hierfür werden die Sicherheitsarchitekturen der genannten mobilen Plattformen untersucht und die Vor- und Nachteile der entsprechenden Systeme in den einzelnen Bereichen diskutiert. Zusätzlich werden ergänzende Infrastrukturmaßnahmen wie etwa Mobile Device Management (MDM) Systeme umrissen, welche insbesondere plattformübergreifend eingesetzt werden können.

1 Einleitung

Der Einsatz von mobilen Endgeräten wie Smartphones oder Tablets in Unternehmen gewinnt immer mehr an Bedeutung. Diese ermöglichen de facto zu jeder Zeit Zugriff auf wichtige Informationen und Daten aus dem Unternehmensumfeld. Hiermit geht aber auch ein erhöhtes Risiko einher, da die Daten auf mobilen Endgeräten nicht den gleichermaßen umfassenden Schutz genießen können, den sie etwa auf einem Desktop-Gerät in einer sicheren Umgebung hätten. Der Diebstahl von Daten bei der Übertragung aus einer sicheren Umgebung auf ein mobiles Endgerät oder der Verlust bzw. Diebstahl des Geräts selbst stellt eine wesentliche Bedrohung dar.

In diesem Artikel wollen wir auf die Charakteristika und Möglichkeiten eingehen, welche die verbreiteten mobilen Plattformen Android 4.4, Apple iOS 8, BlackBerry 10 und Windows Phone 8.1 bieten, um eine sichere Kommunikation bzw. eine sichere Verwaltung und Speicherung von Daten zu gewährleisten. Zu diesem Zweck beschreiben wir im folgenden Abschnitt die aktuellen Systeme allgemein, um einen kurzen Überblick über die allgemeine Situation zu geben. Im Abschnitt 3 skizzieren wir kurz eine Reihe von grundlegenden

P. Schartner · K. Lemke-Rust · M. Ullmann (Hrsg.) · D•A•CH Security 2015 · syssec (2015) 253-266.

Anforderungen von Unternehmen mit einem erhöhten Sicherheitsbedarf an die Verwendung von mobilen Endgeräten in ihrem Netzwerk. Anhand dieser grundlegenden Anforderungen werden die vorgestellten Plattformen im Abschnitt 3 im Detail betrachtet und entsprechende Vor- und Nachteile der einzelnen Plattformen in den unterschiedlichen Bereichen beschrieben.

2 Übersicht aktueller Systeme

Für die nachfolgende Betrachtung sei darauf hingewiesen, dass die angeführten Bezeichnungen und Marken unterschiedliche Hard- und Softwareumfänge bezeichnen. Beispielsweise bezeichnet „Android“ lediglich den Betriebssystem-Kern - ein "Android-Gerät" kann je nach Anbieter völlig unterschiedliche Konfigurationen aufweisen - während „BlackBerry“ als Begriff die Gesamtheit von Betriebssystem und Apps umschreibt. Die nachfolgende Beschreibung geht auf diese Unterschiede nur punktuell ein.

2.1. Android

Die Android Plattform wird von der Open Handset Alliance (OHA) angeboten und zeichnet sich vor allem durch den frei verfügbaren Kernel aus (Open Source). Der Android Kernel basierte einst auf einem modifizierten Linux-Kernel der 2.6 Serie. Nach anfänglichen Problemen sind aber viele der Android Modifikationen wieder in die Linux Main Kernel Linie aufgenommen worden. Im Laufe der Entwicklung hat die OHA eine Serie von Versionen herausgegeben, wobei wir in diesem Artikel näher auf Android 4.4 („KitKat“) eingehen werden.

Ein großes Problem von Android ist die weltweite Fragmentierung der Versionen [Info13]. Oftmals betreffen einzelne Exploits bestimmte Versionen des Android Kernels, der Android Plattform selbst oder bestimmte Teile der Google Apps. Grund dafür ist, dass jeder Hersteller prinzipiell die Freiheit hat, sein Gerät nach Belieben mit Software und Treibern auszustatten. Das führt dazu, dass grundsätzlich einzelne Hersteller und in weiterer Folge auch die Mobilfunkanbieter erst die notwendigen Updates und Fixes in ihren Android Adaptionen einbringen müssen, bevor diese an die Kunden weitergereicht werden. Aus Mangel an Ressourcen oder Know-How erfolgt diese Adaption manchmal spät oder gar nicht, was zu einem erhöhten Risiko für Besitzer vom Exploit betroffener Geräte führt.

2.2. Apple iOS

Das iOS ist ein von der Firma Apple entwickeltes Betriebssystem für mobile Geräte wie dem iPhone und dem iPad. Zudem ist iOS aber auch auf anderen Geräten der Firma Apple zu finden – eine Unterstützung von Geräten anderer Hersteller ist aber nicht gegeben. Das System entstand im Jahr 2005 aus der Idee, ein Betriebssystem für Tablet-Computer zu entwickeln, wobei man sich später auf Mobiltelefone konzentrierte. Mit einer Verbreitung von 42,61% (laut http://www.netmarketshare.com) besitzt iOS den zweitgrößten Marktanteil im mobilen Bereich.

Das Betriebssystem iOS ist aktuell in der Version 8 veröffentlicht und basiert technisch gesehen auf Mac OS X, welches wiederum auf das Unix-ähnliche BSD-Derivat Darwin zurückgeht.

2.3. BlackBerry

Der Fokus von Blackberry Ltd. lag ursprünglich im Business-Bereich und im Einsatz bei Regierungseinrichtungen. Dadurch verfügt das aktuelle Betriebssystem, BlackBerry 10, aufgrund des Micro-Kernel System QNX über eine stabile Umgebung und bereits eingebaute Security-Features wie etwa ein striktes und granulares Benutzer-Berechtigungssystem. Zusätzlich ist durch die BlackBerry Balance Funktion eine Trennung in einen geschäftlichen und einen privaten Bereich möglich. Dadurch werden geschäftliche von privaten Daten in separaten „Spaces“ getrennt und sind (je nach Einstellung) mit einem zusätzlichen Passwort geschützt.

BlackBerry Mobiltelefone können zwar als Einzellösungen verwendet werden, jedoch werden sie vermehrt in Verbindung mit einer Middleware, dem BlackBerry Enterprise Service, in Unternehmen eingesetzt. Diese Middleware stellt die zentrale Stelle für die Verwaltung aller mobilen Endgeräte eines Unternehmens dar und ermöglicht nicht nur einen Zugriff auf die diversen E-Mail Lösungen eines Unternehmens (Microsoft Exchange, Lotus Domino oder Novell GroupWise), sondern auch auf andere interne Informationen wie etwa LDAP oder interne Applikation wie SAP. Hierfür ist jedoch eine entsprechende Infrastruktur erforderlich.

2.4. Windows Phone

Die allgemeine Bezeichnung „Windows 8“ beschreibt aktuell eine Reihe verschiedener Weiterentwicklungen des bekannten Betriebssystems Windows 7. Diese sind u.a. „Windows RT“, sowie die Versionen Windows 8 und höher (8.1, Pro, etc.) und das „Windows 8 Phone“. Die genannten Derivate besitzen einen gemeinsamen Betriebssystemkern und unterscheiden sich in der jeweiligen Prozessor-Architektur, auf welche das Betriebssystem ausgelegt ist, sowie einigen (wenigen) Systemfunktionen. Unterschiede existieren etwa in der Verwaltung bzw. Verwendbarkeit von Apps: Hier lässt Windows 8.1 auf Tablets etwa die Installation beliebiger Software zu, während Windows Phone lediglich digital signierte Apps, etwa aus dem Windows Store oder einem unternehmensinternen Verteilungspunkt, genannt Company Hub, zulässt. Weitere Unterschiede bestehen auch bei Virtualisierung, zumal Windows Phone – anders als Windows 8.1 – keinen Hypervisor unterstützt. Gesonderte Betrachtungen der verschiedenen Ausprägungen können abhängig vom Anwendungsgebiet zweckmäßig sein, da auf den relevanten mobilen Endgeräten (Smartphone und Tablet) unterschiedliche Derivate von Windows 8 zum Einsatz kommen können. Die Betrachtungen in diesem Artikel sind jedoch auf Windows Phone ausgerichtet.

3 Evaluierung der aktuellen Systeme

In der vorliegenden Studie bezeichnet ein „erhöhter“ Sicherheitsbedarf primär die Sicherheitsanforderungen für Informationen der Geheimhaltungsstufe „EU Restricted“ [EC11, EC13]. Anwendungen in diesem Umfeld sind zumeist militärischer Art oder stehen im Zusammenhang mit Regierungs- oder regierungsnahen Institutionen bzw. Unternehmen. Da eine vollständige Darstellung der relevanten Sicherheitsanforderungen den Rahmen dieser Studie sprengen würde, beschränken wir uns auf einige ausgewählte Aspekte, und verweisen auf die Literatur für ergänzende Details. Nachfolgend betrachten wir exemplarisch Aspekte der *sicheren Kommunikation*, *Verfügbarkeit* und *Zugriffskontrolle* (i.S.v. Kapselung bzw. Abschirmung von Prozessen, Apps und Daten gegeneinander). Prinzipiell gilt bei erhöhtem Sicherheitsbedarf (im hier betrachteten Sinne), dass die *vollständige* Kontrolle über die

Funktionalität des mobilen Geräts, insbesondere der darin enthaltenen kryptographischen Schlüssel, *ausschließlich* beim Administrator liegen muss. Für das Management von Schlüsseln etwa bedeutet dies, dass Schlüssel (egal welcher Art) vom Benutzer nicht geändert werden dürfen. Umgekehrt darf im Gerät auch kein Schlüssel existieren, der nicht vom Administrator geändert werden kann. Diese exklusive Rechte des Administrators umfassen sinngemäß auch die Verwaltung von Apps, die Freigabe bzw. Sperre von beliebigen, insbesondere die Kommunikation betreffenden, Funktionen und Hardware, sowie jedwede Ereignis-Protokollierung innerhalb des Geräts.

Die Diskussion der Sicherheitsmaßnahmen erfolgt hiernach insbesondere bezüglich der Möglichkeiten einer Umsetzung auf den vorhin vorgestellten Plattformen. Eine Betrachtung von Spezial-Lösungen (etwa dem Blackphone oder anderer Sonderanfertigungen im Mobilfunkbereich) ist von unserer Darstellung explizit ausgenommen.

3.1. Sicherheitsarchitektur

Die Sicherheitsarchitekturen der hier betrachteten Plattformen folgen einer gemeinsamen Grundstruktur, welche in Abbildung 1 skizziert wird, und deren zentrales Element die teilweise Auslagerung bzw. Unterstützung von Kryptographiediensten in Hardware ist (etwa bei iOS durch eigene Krypto-HW, oder durch eine softwaremäßige Unterstützung von TPM). Die Sicherheitsarchitektur im iOS ist auf den Schutz der Software als auch der Hardware in den Kern-Elementen aller Geräte ausgelegt. Dadurch soll erreicht werden, dass jede Komponente eines Systems vertrauenswürdig ist. Dies gilt sowohl für den Boot-Prozess, als auch für die Software-Updates bis hin zu Apps von Drittanbietern. Einen zentralen Punkt stellt hier die *Secure Boot Chain* dar (vgl. [Appl14], S. 4). Hier wird jede einzelne Komponente, die in einem Schritt des Boot-Prozesses verwendet wird von Apple signiert, um ihre Integrität zu gewährleisten. Hierzu zählen der Bootloader, der Kernel, die Kernel-Erweiterungen und die Firmware. Nach dem Einschalten eines iOS-Device, wird Code aus dem Boot ROM ausgeführt. Dieser Code wird während der Herstellung des Chips eingespielt und kann danach nicht mehr verändert werden. Darin ist der Public Key des Apple Root Certificate gespeichert, mit dem die Integrität des Low-Level Bootloaders verifiziert wird. Ist dieser in Ordnung, so wird der nächste Bootloader „iBoot“, geladen, welcher den iOS Kernel verifiziert und lädt (siehe [Appl14], S. 4). Durch die Secure Boot Chain wird somit garantiert, dass die Basis-Strukturen und die Basis-Komponenten eines Device nicht verändert wurden und dass das iOS nur auf zugelassenen Apple-Geräten läuft.

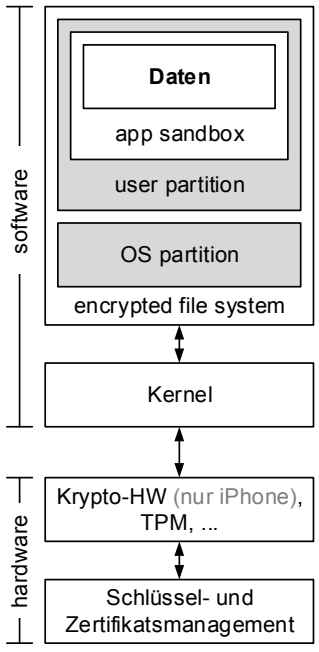

Abb. 1: Sicherheitsarchitektur (allgemein)

Auch BlackBerry hat von Beginn der Fertigung seiner Produkte an in jede wichtige Komponente Security-Features eingebaut, die nur schwer zu entfernen oder zu überwinden sind (siehe [Blac13], S. 10). Hierzu gehört etwa ein Schlüssel, der im Prozessor hinterlegt ist und dazu verwendet wird, den *Replay Protection Memory Block* eines Device zu verschlüsseln (siehe auch Abschnitt 3.2). Um diese Fertigung von Trusted Devices zu garantieren, verfolgt BlackBerry die Strategie einer sehr engen Kooperation der Produktionspartner, der Supply Chain und der BlackBerry Infrastruktur. In jedes Mobiltelefon wird ein hardware-basiertes ECC 521 Bit Schlüssel-Paar eingebaut, welches etwa für die

Verifikation oder die Nachverfolgung verwendet werden kann. Diese ermöglicht die Identifikation von gefälschten Geräten und verhindert eine Verbindung solcher Geräte mit der BlackBerry Infrastruktur bzw. eine Verwendung von BlackBerry Services mit solchen Geräten. Dadurch ist sichergestellt, dass nur Geräte, welche von BlackBerry gefertigt wurden und den Verifizierungsprozess durchlaufen haben, mit der BlackBerry Infrastruktur interagieren können.

Im Falle von Windows Phone ist jedes mobile Endgerät mit einem sogenannten OEM (Original Equipment Manufactured) Verifikationsschlüssel (*OEM Verification Key* – OVK) ausgerüstet, welcher zum Nachweis der Echtheit der Hardware dient [Micr14]. Hersteller von mobilen Endgeräten für den Einsatz von Windows Phone können sich die notwendigen öffentlichen Schlüssel selbst erzeugen (Microsoft empfiehlt hierfür den Einsatz eines Hardware Security Moduls – HSM), und diesen von Microsoft signieren zu lassen (in der Rolle einer Zertifizierungsinstanz – siehe [Micr14]).

Android 4.4 greift auf Technologien von SELinux zurück. Android Prozesse laufen in Sandboxen ab und über Mandatory Access Control wird der Zugriff auf Ressourcen geregelt. Als Android wird jedoch lediglich das Betriebssystem bezeichnet welches an keine explizite Hardwareumgebung gebunden ist. Daraus folgt, dass Sicherheitsarchitekturen erst zusätzlich als Erweiterung des Android Systems nachgereicht werden müssen, wenn diese auch Hardware Komponenten (bsp. ARM TrustZone) berücksichtigen. Dies erfordert allerdings einen massiven Eingriff in das System. Stock Android verlegt Sicherheitsaspekte vorrangig auf die Softwareebene.

3.2. Verschlüsselung und Schlüsselmanagement

Für die Verschlüsselung besitzen Endgeräte mit iOS einen eigenen Coprozessor, den *Secure Enclave*, der direkt in den A7 Prozessor von Apple integriert ist. Er verfügt über eine eigene Secure Boot Chain und auch über einen separaten Update-Prozess für die Software. Der Secure Enclave Coprozessor ist für alle kryptographischen Operationen im Bereich Datensicherheit und Schlüsselmanagement zuständig und erlaubt die Aufrechterhaltung der Integrität der Daten, auch wenn der Kernel kompromittiert wurde (siehe [Appl14], S. 5). Bei der Herstellung des Secure Enclave wird dieser mit einer eigenen UID versehen, welche von keiner anderen Komponente des Systems ausgelesen werden kann und auch Apple selbst nicht bekannt ist. Auf Basis dieser UID wird beim Booten ein Schlüssel generiert, mit dem der Speicherplatz des Secure Enclave im Hauptspeicher verschlüsselt wird (siehe [Appl14], S. 6).

Für die Speicherung von Schlüsseln stellt ein iOS-Device einen speziellen Speicherbereich zur Verfügung, den *Effaceable Storage*, der direkt adressiert und durch Überschreiben sicher

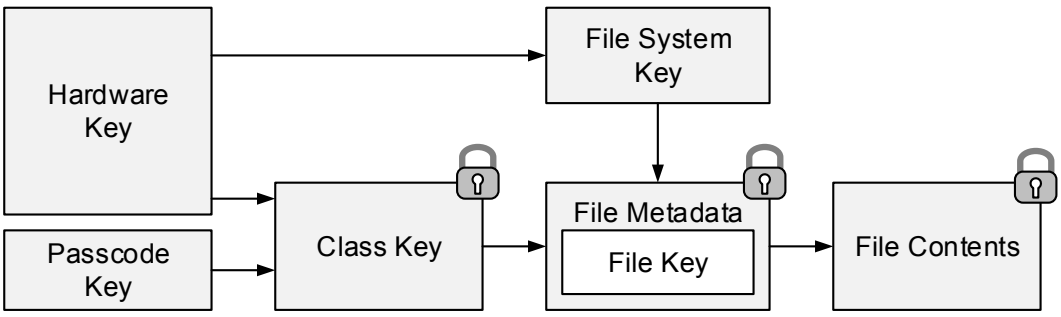

Abb. 2: Verschlüsselungsarchitektur im iOS (vgl. [Appl14])

gelöscht werden kann. Nachdem dieser Speicherbereich keinem physischen Angriff standhält, wird er nicht für langfristig genutzte Schlüssel verwendet. Er ermöglicht aber den Aufbau einer hierarchischen Schlüsselstruktur und Funktionen für Forward Security oder einen Remote Wipe.

Die BlackBerry Plattform verfügt über zwei separate Bereiche; den geschäftlichen Bereich und den privaten Bereich. Während die Verschlüsselung des geschäftlichen Bereichs obligatorisch ist (siehe [Blac13], S. 119), kann der private Bereich optional ebenfalls verschlüsselt werden. Die Daten werden in beiden Fällen AES-256-XTS-verschlüsselt, wobei ein entsprechender File Encryption Key direkt im Gerät erzeugt wird. Ähnlich wie im iOS werden die File Encryption Keys durch ein hierarchisches System von Schlüsseln geschützt (siehe Abbildung 3 und [Blac13], S. 50):

- Der *File Encryption Key* wird mit dem Work Domain Key verschlüsselt und das Chiffrat in den Meta-Daten der Datei abgespeichert.
- Der *Work Domain Key* ist ebenfalls ein durch das Gerät zufällig generierter Schlüssel und wird, verschlüsselt mit dem Work Master Key, in den Meta-Daten des Dateisystems abgelegt.
- Auch der *Work Master Key* ist durch das Gerät zufällig erzeugt und im Non-Volatile Random-Access Memory (NVRAM) der Hardware abgelegt, wo er mit dem System Master Key verschlüsselt ist.
- Letztlich ist der *System Master Key* im Replay Protection Memory Block der Geräte abgelegt. Die Verschlüsselung dieses Blocks erfolgt mit einem Schlüssel, welcher zum Zeitpunkt der Fertigung im Prozessor eingebettet wird.

Die File Encryption Keys, der Work und der System Master Key werden durch den BlackBerry OS Cryptographic Kernel generiert, welcher nach FIPS 140-2 zertifiziert ist.

Bei Android gibt es keine vergleichbare Schlüsselhierarchie, jedoch kann ein Gerät mit Android 4.4.4 als gesamtes verschlüsselt werden, wobei durch Eingabe eines PINs das Gerät entsperrt wird. Die Verschlüsselung ist rein in Software implementiert und basiert dabei auf dem aus Linux bekannten DM-Crypt [Andr14]. Verschlüsselt wird seit Android 3.0 Honeycomb mit AES-128-CBC und ESSIV:SHA256 unter Rückgriff auf die OpenSSL Bibliothek. Google versuchte bei der Adaption des DM-Crypt Verfahren des Linux Kernels weitestgehend auf GPL Code zu verzichten und baute die erforderlichen Funktionalitäten nach (z.B. cryptsetup-Kommando und libdevmapper Bibliothek). Auch beherrschen aktuelle Linux Kernel ein wesentlich größeres Repertoire an Verschlüsselungsalgorithmen und Schlüssellängen.

Als Alternative zum Linux Unified Key Setup (LUKS), dem Schlüsselverwaltungssystem in Linux, verfügt Android 4.4 über eigene Mechanismen, welche auf PBKDF2 oder auf scrypt aufsetzen. Da sich jedoch der Schlüssel entweder aus dem PIN oder dem Passwort ableitet, ist der Schlüsselraum mit Bruteforce Attacken ggf. schnell durchsuchbar. So können etwa 20.000 PBKDF2 Testberechnungen pro Sekunde (entspricht einem 6 stelligen PIN in unter 10 Sekunden oder einen 6 Kleinbuchstaben in etwa 4 Stunden) bzw. 1200 PIN Kombinationen mit scrypt in nahezu 5 Minuten analysiert werden [Elen14]. Entgegen ersten Ankündigen verfügt Android auch in der Version 5 nicht über standardmäßig aktivierte Full Disk Encryption (FDE) [Andr15].

Beim Windows Phone steht von Seiten des Betriebssystems für Apps die transparent arbeitende Laufwerksverschlüsselung Bitlocker zur Verfügung. Die explizite Verschlüsselung von Daten durch eine App ist unabhängig von Bitlocker ebenfalls möglich. Das API bietet als vorhandene

Algorithmen AES und RSA an (für eine Auflistung der vorhandenen kryptographischen Funktionen siehe [Micr14b] [Micr14c]), sowie geeignete Interfaces zur Implementierung eigener Verschlüsselungsverfahren. Zu den unterstützten kryptographischen Verfahren zählen Verschlüsselung (RSA, AES), digitale Signaturen (RSA), Hashing (SHA-1, SHA-256 und darauf aufbauend Hash-MACs) sowie Password-based Key Derivation Function (PBKDF2) und Pseudozufallsgeneratoren (auf Basis der oben genannten Hashfunktionen gemäß RFC2898 [NeWG00]).

Apps können diese Verschlüsselung sehr einfach durch API Funktionen nutzen, wobei der erforderliche Ver- und Entschlüsselungsschlüssel aus den Benutzer-Daten, den Gerät-Informationen und einem zusätzlichen Geheimnis („optionalEntropy“) abgeleitet wird. Letztgenanntes Geheimnis ist nicht zwingend erforderlich, dient jedoch der Vermeidung identischer Schlüssel zwischen verschiedenen Apps (da ohne das zusätzliche Geheimnis zwei unabhängige Apps dieselben Daten für die Schlüsselerzeugung verwenden, und damit auch identische Schlüssel erhalten würden). Die Ableitung der Schlüssel (bei Bedarf) dient zur Vermeidung einer Speicherung von Schlüsseln direkt im Gerät, und damit der Erhöhung der Sicherheit. Diese Routinen bieten jedoch nicht die Möglichkeit einen selbst-gewählten (oder vom Unternehmen vorgegebenen) Schlüssel zu verwenden.

3.3. Benutzergruppen und Benutzermanagement

Der *BlackBerry Enterprise Server* verfügt über eine weitreichende Rollen-, Gruppen- und Benutzerverwaltung [Blac14a]. Diese kann als Stand-Alone Variante oder in Abstimmung mit bereits bestehenden Rollen-, Gruppen- und Benutzer-Konzepten eines Unternehmens erstellt werden. Dadurch besteht die Möglichkeit, auf den Geräten die gleichen Zugriffsberechtigungen und Zugriffskontrollen umzusetzen, wie sie bereits im Unternehmen bestehen. Für die zu verwendenden Passwörter können im BlackBerry Device Service eine Reihe von Policies frei definiert werden (siehe [Blac13], S. 105). Hierzu zählen etwa die maximale Länge eines Passworts sowie dessen Komplexität, die maximale Gültigkeitsdauer, die maximale Anzahl an Eingabeversuchen sowie eine Passwort-Historie. Durch die entsprechenden Regeln [Blac14b] kann die Passwortstruktur an die individuellen Anforderungen eines Unternehmens angepasst werden.

Für den Fall, dass ein Passwort vergessen wird oder das BlackBerry Device verloren geht, kann es über das *BlackBerry Device Service* von zentraler Stelle aus zurückgesetzt werden (siehe [Blac13], S. 106). Bei Verlust eines Device ist es aber auch möglich, einen Data Wipe, entweder nur auf dem geschäftlichen Bereich oder aber für das gesamte Device, remote durchzuführen (siehe [Blac13], S. 113). Sollte das Device keine Verbindung zum BlackBerry Device Service haben, wird das Kommando abgeschickt, sobald sich das Device das nächste Mal mit dem Netzwerk verbindet. Ein Wipe der gesamten Daten des Device wird ebenfalls durchgeführt, sollte die Anzahl der erlaubten Eingabeversuche für das Device-Passwort überschritten werden.

Bei einigen Systemen bietet das Windows Phone eine „Kinderecke“ für den eingeschränkten Zugriff durch andere Benutzer an. Die Einschränkung kann die Freigabe von beliebigen Spielen, Apps, Musikdateien und Videos durch den normalen Benutzer betreffen. Hervorzuheben ist aber auch, das „Unternehmensbereichs-Konto“ anlegen zu können. Dieses bietet u.a. die Möglichkeit zur Erfassung persönlicher Informationen, der (De)Aktivierung von Apps oder Features, dem Verhindern einer Rücksetzung des Endgeräts oder der Löschung des Unternehmensbereichs-Kontos sowie dem Löschen oder Ändern aller Inhalte und Einstellungen per Remote-Verbindung [Micr14d].

Ein Hard-Reset bzw. eine Löschung der Daten nach einer maximalen Anzahl von Fehlversuchen ist über ein Mobile Device Management (MDM) System möglich bzw. alternativ auch nach Registrierung des Telefons auf http://www.windowsphone.com. Folgende Optionen sind u.a. verfügbar:

- Remote wipe: (automatisches) Löschen aller Daten im mobilen Endgerät, etwa nach Überlauf eines Fehlbedienungszählers.
- Remote retirement: Remote wipe und Entfernen des Geräts (bzw. aller relevanten Daten bzw. Informationen zum Gerät) aus dem MDM.
- Remote lock: Sperrung der Benutzung (Daten bleiben erhalten).

Passwort-Recovery ist über Exchange ActiveSync möglich, sofern die entsprechenden Berechtigungen vorab erteilt wurden (Authentifizierung). Bei Registrierung des Telefons auf http://www.windowsphone.com steht eine Funktion „Remote password (PIN) reset“ zur Verfügung. Diese dient dem Rücksetzen des PIN-Codes und ist für Reaktivierung durch einen Administrator gedacht, falls ein mobiles Endgerät irrtümlich gesperrt wurde.

Mobiltelefone mit einem nativen Android-System verfügen von sich aus nicht über Multi-User Funktionalität oder spezielle Eigenschaften zum Benutzermanagement. Seit Android 4.3 unterstützt das System zwar sog. „Restricted Profiles“, was einem Multi-User Konzept nahekommt, jedoch ist dieser Mechanismus auf Telefonen nicht aktiviert sondern ist in erster Linie für Tablets gedacht. Es lassen sich aber Funktionen wie etwa Remote Wipe oder Password Reset, bis hin zu Virtualisierungslösungen und Aufspaltung in verschiedene Bereiche für verschiedene Benutzer am gleichen Gerät nachrüsten. So bietet etwa der Android Device Manager [Andr14b] die Möglichkeit, über eine Weboberfläche ein remote Wipe für ein beliebiges Android-Gerät eines Users durchzuführen. Ein remote Kill ist hier allerdings nicht möglich. Auch kann der Anwender kein bestimmtes Passwort zum Entsperren des Geräts hier hinterlegen. Dies ist erst mit weiteren MDM Lösungen möglich.

Ähnlich wie Android bietet iOS ebenfalls generell keine Multi-User Funktionalität an. Somit können auch nicht mehrere Benutzerprofile gleichzeitig verwendet werden. Eine entsprechende Erweiterung in Richtung der Verwaltung von Benutzerkonten und -strukturen kann über die von Apple zur Verfügung gestellten Programme „iPhone Configuration Utility“ [Appl14b] oder „Apple Configurator“ [Appl14c] bzw. über MDM Systeme vorgenommen werden.

3.4. Sicherheit von Apps und App-Stores

Das iOS lässt sich durch Apps erweitern, welche aus dem *AppStore* geladen werden können. Die dort vorhandenen Apps werden von externen Entwicklern geschrieben und nach Kontrolle durch Apple in den AppStore aufgenommen. Hierfür sind im iOS eine Reihe von Schutzschichten installiert, die gewährleisten sollen, dass Apps signiert und verifiziert werden. Zudem soll sichergestellt werden, dass die Apps keinen schadhaften Code ausführen. Dies wird erreicht, indem die Secure Boot Chain auch im Bereich der User Prozesse und Apps weitergeführt wird (siehe [Appl14], S. 14). Im Detail wird durch eine Signatur des Codes einer App mit einem Zertifikat von Apple sichergestellt, dass nur der Code aus einer vertrauenswürdigen Quelle kommt und nicht verändert wurde. Bei Apps, die auf dem Device mitgeliefert werden (wie etwa Mail oder Safari) erfolgt das Signieren von Apple direkt. Apps von Drittanbietern müssen einen entsprechenden Signaturprozess durchlaufen. Das Installieren von unautorisierten Apps ist offiziell nicht möglich. Allerdings wird mit Hilfe eines Jailbreaks

die Software derartig verändert, dass unautorisierte Apps auf den Geräten installiert werden können.

Um zur Laufzeit ebenfalls zu gewährleisten, dass Apps sich nicht gegenseitig kompromittieren können, ist jede App in einer Sandbox gekapselt. Dabei wird jeder App ein eigenes Home-Verzeichnis zugewiesen, welches an einer zufälligen Stelle im Speicher abgelegt wird und auf das nur diese eine App zugreifen kann. Dieses Feature wird als *Address Space Layout Randomization* (ASLR) bezeichnet und es dient als Schutz gegen komplexe Angriffe, welche auf Speicher-Exploits aufbauen (siehe [Appl14], S. 15). Durch den Einsatz der Sandbox wird etwa verhindert, dass eine App (lesenden oder schreibenden) Zugriff auf die Daten der anderen Apps hat oder Veränderungen an den System-Dateien des Device machen kann. Muss eine App dennoch auf die Daten einer anderen App zugreifen, erfolgt dies über ein Application Programming Interface (API), welches vom iOS zur Verfügung gestellt wird. Wenn eine App auf zusätzliche Features wie etwa die iCloud zugreifen möchte, muss eine entsprechende Autorisierung über sogenannte *Entitlements* erfolgen (siehe [Appl14], S. 15).

Bei der Windows Phone Plattform können Apps generell über den *Microsoft Store* bezogen werden. Die Ausführung einer App setzt dabei immer eine gültige digitale Signatur voraus. Microsoft ermöglicht allen Programmierern die Entwicklung von Apps; ein Vertrieb über den Windows Store setzt jedoch das Durchlaufen diverser von Microsoft vorgeschriebener Tests voraus, nach deren positivem Abschluss ein Zertifikat von Microsoft im Namen der Entwickler ausgestellt wird [Micr14e]. Im Einzelnen umfassen diese Anforderungen Aspekte wie Zahlungssysteme (etwa bei Shopping Apps), Inhalte (z.B. Urheberrechtsschutz, Verbot diskriminierender oder pornographischer Inhalte), die Nutzung des System-API [Micr14f], sowie eine Reihe weiterer Aspekte.

Wie bei anderen Plattformen laufen auch im Windows Phone Apps in gekapselten Bereichen (Sandboxes), was jedoch nicht per se zu einer vollständigen Isolation führt. So eröffnen etwa URI Assoziationen (z.B. „http:“ → Webbrowser, „mailto:“ → E-Mail-Client, etc.) einer App die Möglichkeit, Daten direkt an eine andere App zu übermitteln, und diese ggf. sogar zu starten. Neben dieser einfachen Technik bietet die Plattform auch noch weitere Varianten der (u.U. unzureichend kontrollierbaren) Interprozesskommunikation an. Die logische Isolation der Speicherbereiche mancher Apps kann Angriffe bzw. Malware somit nicht zwingend verhindern.

Zusätzlich zum Windows Store können Apps aber auch von einer Institution über ein selbstbetriebenes Management System (analog zu einem MDM), genannt *Company Hub*, bezogen werden [Micr14g]. Der Company Hub ist eine spezielle App, welche als Portal innerhalb des mobilen Endgeräts agiert, und wenigstens die Suche, Installation und Ausführung von Apps der Institution (Firma) ermöglicht. Allerdings ist bei einem Company Hub zu beachten, dass die oben genannten Sicherheitsanforderungen und -kontrollen nur jene Apps betreffen, welche über den Windows Store vertrieben werden. Die über den Company Hub von dem Unternehmen vertriebenen Apps unterliegen jedoch keinen weiteren Kontrollen durch Microsoft und müssen durch geeignete Sicherheitsmaßnahmen im Entwicklungsprozess durch das Unternehmen geschützt werden.

Ähnlich wie bei den anderen Anbietern können auch bei BlackBerry Apps von Drittanbietern über eine zentrale Plattform, *BlackBerry World*, bezogen werden. Wie bei den Systemen von Apple und Windows verfügt BlackBerry über einen Verifikationsprozess, um etwa Malware zu entdecken. Dabei kooperiert BlackBerry mit dem japanischen Unternehmen Trend Micro, und setzt seit der Einführung von BlackBerry 10 dessen Mobile App Reputation Service zusätzlich

zum eigenen System für die Analyse von Apps ein [Blac14c] [TrMi13]. Das Mobile App Reputation Service ist ein cloud-basierter Dienst von Trend Micro, der Apps auf bösartige Aktivitäten, Ressourcenverbrauch sowie auf Verstöße gegen die Vertraulichkeit der Daten prüft [TrMi14]. Hierfür wird sowohl ein statisches Code Review als auch einen dynamischen Scan des Verhaltens der App durchgeführt. Erst wenn eine App diesen Review bestanden hat, kann sie in den BlackBerry World Store aufgenommen werden.

Seit der Version 10.2.1 des BlackBerry Betriebssystems ist es auch möglich, Android Apps aus APK-Pakete auf dem BlackBerry 10 zu installieren. Derzeit erfolgt die Installation der Apps nicht über einen App-Store sondern muss manuell – durch explizite Installation des APK-Pakets – durchgeführt werden. Die verwendeten APK-Pakete sollten daher aus einer vertrauenswürdigen Quelle bezogen werden (z.B. von der Seite des Herstellers), was aber nicht immer möglich ist. Somit könnten Apps mit schadhaftem Code auf das Gerät gelangen. Daher werden die Android Apps in einer Sandbox, dem sogenannten *Android Player*, ausgeführt und besitzen dadurch nur eingeschränkte Berechtigungen [Gole14]. Zudem sind Android-Apps nur für den privaten Bereich zugelassen und können nicht im geschäftlichen Bereich installiert werden

Zusätzlich zum öffentlichen BlackBerry World App-Store bietet BlackBerry ähnlich wie Windows für Unternehmen die Möglichkeit, einen eigenen App-Store einzurichten, den *BlackBerry World for Work*. Alle Apps im BlackBerry World for Work können nur im geschäftlichen Bereich des Device installiert werden und von dort auf die unternehmensinternen Strukturen zugreifen. Die Steuerung des BlackBerry World for Work erfolgt über den BlackBerry Device Service und ermöglicht sehr spezifische Einstellungen bzgl. der Verbreitung der Apps.

Im Gegensatz zu den anderen Anbietern ist die zentrale Anlaufstelle für das Kaufen und Herunterladen von Applikation, genannt *Google Play*, völlig frei. Prinzipiell darf jeder Entwickler eine Applikation für Android schreiben und diese auf Google Play anbieten. Es erfolgt lediglich eine automatisierte Malware-Überprüfung durch das Tool „Bouncer“ [Time14] und erst, wenn Apps als unsicher oder regelwidrig gemeldet werden, wird eine genauere Prüfung der Software-Qualität durch Google durchgeführt.

Es existiert aber beim Installieren und somit in Folge auch beim Ausführen von Applikationen auch für Android Apps eine Sicherheitsschranke. Einzelnen Apps müssen durch den Benutzer ausdrücklich die Berechtigungen für den Zugriff auf einzelne Ressourcen gewährt werden. Werden diese Rechte nicht gewährt, wird die App nicht installiert. Diese Vorgehensweise hat jedoch zwei gravierende Nachteile (siehe [Info13]): zum einen verlangen Apps im Hinblick auf zukünftige Funktionalitäten oft mehr Berechtigungen, als sie brauchen, wodurch zum anderen die Anwender mit der Fülle an zu erteilenden Berechtigungen überfordert sind und diese zu vorschnell gewähren.

4 Infrastrukturelle Maßnahmen

Die Vielfalt und unterschiedlichen – bisweilen auch nicht vergleichbaren – Eigenschaften der dargestellten Plattformen, ermöglichen keine objektive Auswahl oder Empfehlung eines „geeignetsten“ Kandidaten für beliebige Anwendungen im Hochsicherheitsbereich. Es erscheint daher erforderlich, die Auswahl der konkreten mobilen Plattform an das geplante Einsatzumfeld anzupassen. Die hierdurch implizierte Heterogenität erfordert ein zentrales

Management der Endgeräte, welches die Spezifika und Möglichkeiten der Endgeräte aus Sicht der Administration und Sicherheitsbeauftragten transparent nutzt.

Alle genannten mobilen Endgeräte lassen sich über Systeme von Drittanbietern oder auch über selbst betriebene Server fernwarten. Diese Mobile Device Management Systeme (MDM) können zur Erhöhung der Sicherheit eingesetzt werden, etwa indem die Berechtigung bestimmter Zugriffe an die Zustimmung einer weiteren Instanz über das MDM gebunden wird, oder auch indem im Verlustfall des mobilen Endgeräts dieses lokalisiert bzw. das Gerät oder die Daten darin auch unbrauchbar gemacht werden können.

Insbesondere im Hinblick auf die Vermeidung von Malware am Mobiltelefon bietet sich die Verwaltung der Gesamtheit aller unternehmenseigenen mobilen Endgeräte über ein selbst-betriebenes MDM an. Dies ermöglicht die Durchsetzung strengerer Überprüfungen (etwa tiefere Code-Reviews) als sie standardmäßig durch die AppStores angeboten werden, oder die Einschränkung bzw. Überwachung der Nutzung eines mobilen Endgeräts. Darüber hinaus sind einige MDM Systeme unabhängiger Anbieter auch in der Lage, Plattformen verschiedener Anbieter auf einer gemeinsamen Basis zu verwalten. Eine solche Lösung ist insbesondere geeigneten Sicherheitsüberprüfungen und -vorkehrungen im Unternehmen zu unterwerfen, um keine Sicherheitslücken aufgrund der „Mischung“ verschiedener Sicherheitsarchitekturen zu erzeugen.

5 Zusammenfassung und Fazit

Generell kann gesagt werden, dass der Bereich Sicherheit in allen vier Systemen eine wesentliche Rolle spielt. Es ist aber zu bemerken, dass in den geschlossenen Systemen von Apple und Blackberry, die ihre Systeme nur in Verbindung mit speziell gefertigter Hardware anbieten, einige Security-Features bereits in dieser Hardware integriert sind (z.B. Secure Boot Chain bzw. Hardware Root of Trust). Hier bietet ein komplett offenes System wie Android allerdings die Möglichkeit, das System mit einer Hardware zu kombinieren, die erweiterte Security-Konzepte (wie z.B. eine microSD Karte mit Krypto-Funktionalität) anbietet.

In den angesprochenen Systemen werden jedoch die Komponenten für die Verschlüsselung oft bereits während der Fertigung der Hardware integriert, was eine Verifikation der Qualität etwa des Schlüsselmaterials erschwert. Wird Schlüsselmaterial im Endgerät auf Bedarf erzeugt, ist die Qualität der verwendeten Zufallsgeneratoren kritisch zu reflektieren. Hierbei sind offene Systeme i.A. einfacher zu analysieren.

Die Tiefe der von den verschiedenen AppStores durchgeführten Code-Reviews ist unterschiedlich und für einen konkreten Einsatz bzw. Auswahl einer mobilen Endgerätelösung detailliert zu hinterfragen, zumal manche der Überprüfungen im Black-Box-Verfahren durchgeführt werden (aus nachvollziehbaren wirtschaftlichen Gründen). Zudem ist eine Einschränkung der verfügbaren Apps oder ein Einsatz von unternehmenseigenen Apps über einen speziellen AppStore zu berücksichtigen.

Der Grad bzw. die Detailtiefe der Dokumentation einer mobilen Plattform kann als Kriterium für die Auswahl, neben dem Kriterium der Quelloffenheit herangezogen werden. So sind die Interna von Windows Phone, BlackBerry und Apple iOS etwa relativ gut und ausführlich dokumentiert, obgleich die Systeme selbst nicht quelloffen sind. Bei einem quelloffenen System wie Android ist es hingegen möglich, direkt auf das System Einfluss zu nehmen und bei Bedarfs Anpassungen (z.B. Härten des Betriebssystems) vorzunehmen. Tabelle 1 stellt die wesentlichen Vor- und Nachteile der einzelnen Plattformen einander gegenüber.

Tab. 1: Gegenüberstellung der betrachteten Plattformen

Android	Apple iOS	BlackBerry	Windows Phone
+ Offenes System + Jederzeit erweiterbar (alternative Sicherheitslösungen) – Versionsvielfalt im Umlauf (Updates werden Hersteller/ Anbieter überlassen) – Keine Anbindung an TPM bei Stock Android	+ Rasche Updates für iOS und Apps + Auf HW optimiertes System – System ist Großteils eine Blackbox – Hohe Abhängigkeit vom Hersteller	+ Balance-Funktion: Getrennte Bereiche (Geschäftlich / Privat) + Zentrale Verwaltung der Geräte durch Middleware – Verwendung fester werksseitiger Schlüssel – System ist Großteils eine Blackbox	+ Intuitive Bedienung + Umfassende Dokumentation – Standard Krypto-API umfasst nur wenige Algorithmen – Potentiell unzureichende Prozesskapselung

Eine effiziente Verwaltung einer großen Anzahl von Endgeräten ohne dem Einsatz einer entsprechenden Middleware ist aber nicht möglich. Wie angemerkt existieren zwar Einzellösungen für Android, Apple, und Windows, jedoch bietet nur BlackBerry mit dem Enterprise Service eine entsprechende Möglichkeit zu Integration in die Infrastruktur eines Unternehmens an. Eine entsprechende, plattformübergreifende Lösung stellen Mobile Device Management Systeme wie etwa die aktuellen Umsetzungen von AirWatch, Citrix, IBM oder MobileIron dar.

Danksagung

Dieser Artikel wurde im Rahmen des Projekts „SeCom – Sichere IT-Services auf mobilen Endgeräten“ erstellt, welches durch das KIRAS-Programm der österreichischen Forschungsförderungsgesellschaft (FFG) gefördert wird (Projekt-Nr. 840813).

Literatur

[BuIn13] Business Insider, „How Android Grew To Be More Popular Than The iPhone“, 13.8.2013. http://www.businessinsider.com/history-of-android-2013-8?op=1.

[Info13] Infoworld, „A clear-eyed guide to Android's actual security risks“, 9.12.2013. http://www.infoworld.com/d/mobile-technology/clear-eyed-guide-androids-actual-security-risks-232034.

[EC11] Council of the European Union, “Policy on creating EU classified information,” Doc. No. 10872/11, 2011

[EC13] European Council Decision 2013/488/EU, “On the security rules for protecting EU classified information”, Official Journal of the European Union, October 2013, S. L274/1.

[Appl14] Apple Inc., „iOS Security February 2014“, 2/2014. https://www.apple.com/iphone/business/docs/iOS_Security_Feb14.pdf.

[Appl14b] Apple Inc., „iPhone Configuration Utility 3.6.2 for Windows“,. http://support.apple.com/kb/dl1466.

[Appl14c] Apple Inc., „Apple Configurator“,. https://itunes.apple.com/at/app/apple-configurator/id434433123?mt=12.

[Blac13] BlackBerry Ltd., „BlackBerry Enterprise Service 10. BlackBerry Device Service Solution (Version 10.2). Security Technical Overview“, 28.11.2013. http://docs.blackberry.com/en/admin/subcategories/?userType=2&category=BlackBerry+Enterprise+Service+10

[Blac14a] BlackBerry Ltd., „BlackBerry Enterprise Service. BlackBerry Device Service (Version 10.2.4) Advanced Administration Guide“, 10.9.2014. http://docs.blackberry.com/en/admin/subcategories/?userType=2&category=BlackBerry+Enterprise+Service+10

[Blac14b] BlackBerry Ltd., „BlackBerry Enterprise Service 10. BlackBerry Device Service (Version 10.2.3). Policy and Profile Reference Guide“, 16.6.2014. http://docs.blackberry.com/en/admin/subcategories/?userType=2&category=BlackBerry+Enterprise+Service+10

[Blac14c] BlackBerry, „BlackBerry Works with Trend Micro to Expand Protection for Customers Against Malware, Privacy Issues in Third-Party Applications“, 4.2.2013. http://press.blackberry.com/content/rim/press/2013/blackberry-works-with-trend-micro-to-expand-protection-for-custo.html.

[Micr14] Microsoft, „OEM Verifikation Key (OVK)“, 3.10.2014. https://dev.windowsphone.com/de-de/OEM/docs/Manufacturing_Retail/OEM_Verification_Key__OVK_

[Micr14b] Microsoft, „How to encrypt data for Windows Phone 8“, http://msdn.microsoft.com/en-us/library/windows/apps/hh487164%28v=vs.105%29.aspx.

[Micr14c] Microsoft, „Cryptography Namespace“, 28.7.2014. http://msdn.microsoft.com/en-us/library/windows/apps/system.security.cryptography%28v=vs.105%29.aspx..

[Micr14d] Microsoft, „Was ist ein Unternehmensbereichskonto?“, Microsoft, http://www.windowsphone.com/de-at/how-to/wp8/accounts-and-billing/what-is-a-workplace-account, 2014

[Micr14e] Microsoft, „App certification requirements for Windows Phone“, 19.8.2014. http://msdn.microsoft.com/en-us/library/windows/apps/hh184843%28v=vs.105%29.aspx.

[Micr14f] Microsoft, „App submission requirements for Windows Phone“, 19.8.2014. http://msdn.microsoft.com/en-us/library/windows/apps/hh184844(v=vs.105).aspx.

[Micr14g] Microsoft, „Company app distribution for Windows Phone“, 19.8.2014. http://msdn.microsoft.com/en-us/library/windows/apps/jj206943%28v=vs.105%29.aspx.

[NeWG00] Network Working Group, „RFC 2898. PKCS #5: Password-Based Cryptography Specification“, 09/2000. https://www.ietf.org/rfc/rfc2898.txt.

[Andr14] Android.com, „Encryption“. https://source.android.com/devices/tech/encryption 06.10.2014.

[Andr14b] Android.com, „Android Device Manager“. www.google.com/android/devicemanager

[Andr15] Android.com, „Android 5.0 Compatibility Definition“, 01/2015, p. 60, http://static.googleusercontent.com/media/source.android.com/en/us/compatibility/android-cdd.pdf

[Elen14] N. Elenkov, „Revisiting Android disk encryption“. http://nelenkov.blogspot.co.at/2014/10/revisiting-android-disk-encryption.html.

[Gole14] Golem, „Neue Blackberry-Version enthält abgespecktes Android 4.2.2“, 30 1 2014. www.golem.de/news/blackberry-10-2-1-neue-blackberry-version-enthaelt-abgespecktes-android-4-2-2-1401-104266.html.

[Time14] Time, „Android Gets a Malware Scanner for Google Play Store Apps“, 10.04.2014. http://time.com/57878/android-gets-a-malware-scanner-for-google-play-store-apps/.

[TrMi13] Trend Micro, „Trend Micro und BlackBerry kooperieren“, 5.2.2013. www.trendmicro.de/newsroom/pr/trend-micro-und-blackberry-kooperieren/.

[TrMi14] Trend Micro, „Mobile App Reputation Service – Datenblatt“, 15.10.2014. www.trendmicro.de/media/ds/mobile-app-reputation-service-datasheet-de.pdf,