Institut für Parallele und Verteilte Systeme

Universität Stuttgart
Universitätsstraße 38
D–70569 Stuttgart

Diplomarbeit Nr. 3455

BYOD - Private Hardware in der Firma nutzen

Athanasios Panos

Studiengang:	Informatik
Prüfer/in:	Prof. Dr.-Ing. habil. Bernhard Mitschang
Betreuer/in:	Dipl.-Inf. Christoph Stach
Beginn am:	2013-02-18
Beendet am:	2013-07-23
CR-Nummer:	D.4.6, K.4.1

Kurzfassung

BYOD (Bring-Your-Own-Device), auch bekannt als BYOT (Bring-Your-Own-Technology), ist ein neuartiger Trend, welcher die letzten Jahre immer mehr an Zuspruch gewonnen hat. Wie der Name schon zu verstehen gibt, bringen Angestellte ihre privaten mobilen Geräte, in diesem speziellen Fall Android-Smartphones, in das Arbeitsumfeld mit und greifen damit auf sensible Daten des Unternehmens zu.

Vorteile dieses Konzeptes sind steigende Produktivität und größere Flexibilität der Mitarbeiter. Diese Art von Einsatz mobiler Geräte bringt allerdings auch Nachteile in Form von Risiken mit sich, die sich in der Datensicherheit beider Seiten bemerkbar macht. Auch die Privatsphäre kann durch den Einsatz privater Geräte eingeschränkt werden. Laut einigen Studien bringt der Einsatz einer BYOD-Policy weitere positive Nebenwirkungen in das Arbeitumfeld. Demnach entwickeln sich die Gemüter von Angestellten ins positive durch ein fröhlicheres und gemeinschaftlicheres Miteinander.

Auf Basis des mobilen Betriebssystems Android wurde an der Universität Stuttgart die Privacy Management Platform (PMP) entwickelt. Sie regelt den Zugriff auf Ressourcen anhand der vom Benutzer vorgenommenen Einstellungen. Ziel war es die Privatsphäre und die Sicherheit der Daten des Benutzers im Android-Betriebssystem zu erhöhen.

In dieser Diplomarbeit wurden grundlegende Anpassungen an der PMP vorgenommen, um ein Lösungsansatz für das BYOD-Problem zu konzipieren. Mithilfe der Zugriffsverwaltung von Applikationen auf Ressourcen und Funktionen des Smartphones lassen sich die nötigen Einschränkungen in Form von Richtlinien jedem Unternehmen individuell Anpassen. Auch erhält der Benutzer die Möglichkeit, seine Privatsphäre zu schützen, indem diesem noch eigene Anpassungen entsprechend der Firmenrichtlinien zugestanden werden. Um das Ergebnis in seiner abschließenden Form abzurunden, wurde eine Applikation zu Demonstrationszwecken entwickelt.

Des Weiteren werden in dieser Diplomarbeit weitere Ansätze und Lösungen des BYOD-Konzeptes vorgestellt und mit dem in dieser Arbeit entwickelten Konzept verglichen. Unter diesem Aspekt werden abschließend die Möglichkeiten zur sicheren Speicherung von sensiblen Daten auf mobilen Geräten untersucht.

Abstract

BYOD (Bring-Your-Own-Device), also known as BYOT (Bring-Your-Own-Technology), is a new trend, which expanded rapidly in the last few years. As described by the name itself, employees take their private mobile devices with them to work and use them for work purposes by accessing sensitive data of the company.

Advantages of this concept are rising productivity and higher flexibility of the employees. Indeed there are also disadvantages like the risk of data loss and privacy concerns. People using their own devices in the workplace are more likely to get together with their co-workers and develop a nice communality.

On the basis of the mobile operating system android a privacy management platform (PMP) has been developed at the University of Stuttgart. This platform controlls access to ressources on the basis of the user settings. The goal was to secure private data and keep the users privacy save.

In this thesis fundamental changes have been made to the PMP, to develop a solution the BYOD-problem. With the help of the settings for controlling access to certain data and packages we can use rules to regulate the separation of user and corporal data. In addition to the BYOD-platform a demo-app has been developed to demonstrate the functionality.

Furthermore many other approaches have been reviewed in this thesis to compare different aspects of a BYOD-platform. Especially saving sensitive data to a device will be an important topic.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Verzeichnis der Listings

1 Einleitung

1.1 Motivation

Durch den immer größer werdenden Anteil an Smartphones im Bereich der Mobiltelefone, werden diese aufgrund des großen Funktionsumfang immer gerne und häufiger in der Arbeitsumgebung genutzt. Dies gilt vor allem auch für private Geräte. In den meisten Fällen jedoch entsprechen die Sicherheitsvorkehrungen an besagtem Gerät nicht den Sicherheitsstandards des Unternehmens in welchem der Einsatz erfolgt. Infolgedessen setzen sich immer mehr IT-Abteilungen mit der Thematik des „Bring-Your-Own-Device"-Systems auseinander und versuchen die breite Masse an unterschiedlichsten Geräten und mobilen Betriebssystemen in die eigene IT-Infrastruktur mit einzupflegen.

Nachdem Smartphones immer populärer wurden, entwickelt sich nun parallel der Tablet-Markt mit rasanter Geschwindigkeit. Sowohl Besitzer von Smartphones, als auch von Tablets, sehen den Nutzen, welcher durch den Einsatz dieser Geräte entstehen kann. Werden von einem Unternehmen keine Richtlinien bezüglich der Nutzung privater Geräte festgelegt, so muss immer damit gerechnet werden, dass Mitarbeiter sich die Arbeit mit den privaten, kleinen Helfern vereinfachen. Aus diesem Grund müssen sich Unternehmen zwangsläufig mit dieser Thematik auseinandersetzen um Sicherheitsrisiken zu vermeiden.

Fließen mobile Geräte in einem Unternehmen in den Entwicklungsprozess mit ein, so werden sich Mitarbeiter irgendwann mal in dem Szenario wiederfinden, in dem sie neben ihrem privaten Smartphone zusätzlich ein Gerät zum geschäftlichen Zweck mit sich tragen müssen. Findet sich ein Mitarbeiter in dieser Situation, so bevorzugt dieser es nur ein Gerät bei sich tragen zu müssen. Ob es nun eine Möglichkeit gibt oder nicht, viele Mitarbeiter finden eine Möglichkeit dieses Problem zu umgehen und greifen mit privaten Geräten auf Firmenressourcen zu. Deshalb ist es ratsam, sich in einem Unternehmen mit dieser Thematik zu befassen.

1.2 Aktuelle Daten und Fakten

Durch BYOD sind laut Gartner Kosteneinsparungen von bis zu 40 Prozent möglich. Sowohl Anschaffungs- als auch Unterhaltungskosten sind ein großer Einsparungsfaktor. Durch das Einbinden von privater Hardware ist die ständige Erreichbarkeit von Mitarbeitern ein netter Nebeneffekt für Unternehmer [Rat11].

Durch sogenannte Diensthandys sind im Moment sowieso bereits ein Drittel der Beschäftigten jederzeit für den Chef erreichbar. In den meisten Fällen geschieht dies auf Wunsch des Vorgesetzten. Bitkom stellt außerdem eine Statistik auf, die besagt, dass ebenso ein Drittel aller Arbeitnehmer mindestens ein mal in der Woche von zu Hause arbeitet. Wer oft unterwegs ist und außerhalb des Büros arbeitet erledigt Aufgaben sowohl im Auto, in der Bahn als auch in Cafés und Restaurants [Spi13].

1.2.1 Umfragen

Laut einer Umfrage von Avanade hatten Ende 2011 von 600 befragten Unternehmen 80 Prozent der Mitarbeiter, welche mit privaten Geräten auf Firmendaten zugriffen [Hae12]. Dies zeigt umso mehr, dass der Bedarf an BYOD-Lösungen da ist, und sich Unternehmen mit diesen Thema auseinander setzen müssen.

Auch die BITKOM-Studie vom April 2013 bestätigt den Verdacht. Das Meinungsforschungsinstitut ARIS hat im Auftrag des Branchenverbandes BITKOM 854 Unternehmen und etwa 500 Beschäftigte zu ihrer Erfahrung mit dem Thema BYOD befragt [Thy13]. Im folgenden die Ergebnisse:

- 71 Prozent der Berufstätigen in Deutschland nutzen im Moment private Geräte im Arbeitsumfeld.
- 27 Prozent der Unternehmen in Deutschland erlauben die Nutzung privater Geräte für den Zugriff auf Ressourcen des Unternehmens.

Unterschiede gibt es allerdings auch bei der Art der verwendeten Geräte wie die folgende Statistik zeigt:

- 35 Prozent nutzen das private Notebook für den Beruf.
- 32 Prozent erledigen ihre Arbeit an eigenen PCs.
- 31 Prozent telefonieren mit dem Handy zu Arbeitszwecke.
- Nur 19 Prozent nutzen die umfangreichen Fähigkeiten ihres Smartphones für den Beruf.
- Immerhin 8 Prozent nehmen ihre Tablets mit für den Einsatz im Unternehmen

Durch die Nutzung privater Geräte, vor allem Handys und Smartphones, im Unternehmen erhöht sich allerdings auch die Erreichbarkeit eines jeden Arbeitnehmers. Arbeitgeber haben oftmals strenge Richtlinien an die sich Mitarbeiter halten müssen. BITKOM fragte ebenso nach der Erreichbarkeit von Angestellten, allerdings unabhängig vom Einsatz privater Geräte im Unternehmensumfeld. Demnach sind 77 Prozent der Befragten noch nach Feierabend für den Chef erreichbar. Dabei lesen beispielsweise 62 Prozent noch außerhalb der Arbeitszeiten ihre geschäftlichen E-Mails [Wen13].

Das Problem hier, das ebenfalls für das BYOD-Konzept gilt, ist die nicht klar geregelten Rahmenbedingungen. 62 Prozent der Unternehmen haben dazu gar keine Regelungen zur Erreichbarkeit außerhalb der Geschäftszeiten. In nur 12 Prozent ist dies individuell in den Arbeitsverträgen niedergeschrieben. In 20 Prozent der Unternehmen sind immerhin mündliche Vereinbarungen getroffen worden [Was13, Hul13a].

Ohnehin machen die meisten Angestellten das, was sie wollen. Aufgrund der zu kurz kommenden Absprache und Vereinbarungen über diese Art des Arbeitens verwenden viele Arbeitnehmer auch private Geräte für geschäftliche Zwecke. In den meisten Fällen ist dies vom Unternehmen auch nicht einfach zu verhindern. Deshalb denken sich viele Unternehmen, wenn man es schon nicht verhindern kann, sollte man es fördern. Für die Arbeitgeber stellt das allerdings hohe Anforderungen an die Sicherheit des IT-Infrastruktur. Mitarbeiter dagegen erfreuen sich an einem Produktivitätsgewinn [Vie12, Kle13].

Im Rahmen einer Studie [Tec12] von Techconsult wurden mittelständische und Großunternehmen mit mehr als 250 Computern am Arbeitsplatz befragt. Laut Techconsult machen allerdings Großunternehmen mit mehr als 2000 Computern den größten Anteil aus. Befragt wurden allerdings nur Arbeitnehmer, welche täglich mit mobilen Geräten wie Notebooks, Smartphones oder Tablet-PCs arbeiten und somit potenziell die Möglichkeit haben, private Geräte mit in den Arbeitsprozess mit einzubeziehen [Saw12].

Abbildung 1.1 zeigt das Ergebnis der Umfrage wie viele der Mitarbeiter private Endgeräte im Unternehmen nutzen, und wenn ja, welche Art von Gerät. Mehr als zweidrittel der Angestellten haben demnach schon mindestens einmal ihr privates Gerät für die Arbeit zur Hilfe genommen. Dabei führen Smartphones mit 67 Prozent die Liste an. Gefolgt von Laptops mit 53 Prozent sind diese bei mehr als der Hälfte der Befragten zum Einsatz gekommen. Handys (32 Prozent), Tablet-PCs (15 Prozent) und weitere private Geräte (5 Prozent) werden allerdings ebenso genutzt.

1.2.2 In der öffentlichen Verwaltung

Dass die Thematik über die Nutzung privater Geräte im Beruf besonders umstritten ist, bestätigt die Äußerung von Berlins Landesdatenschützer, Alexander Dix. Im März 2013 stellte der Berliner Datenschutzbeauftragte in seinem Jahresbericht für das Jahr 2012 klar, dass die Nutzung privater Geräte in der öffentlichen Verwaltung weiter verboten bleibt [Dix13]. Genauer heißt es darin: "Meist entziehen sich (...) die Privatgeräte dem IT-Management, mit dem der Arbeitgeber die eigenen informationstechnischen Geräte - auch aus Gründen des Datenschutzes - verwaltet und kontrolliert. Ihr Einsatz birgt deshalb auch datenschutzrechtliche und technische Risiken. In der öffentlichen Verwaltung ist daher der Einsatz privater Datenverarbeitungsgeräte bisher prinzipiell untersagt".

Als Grund für die Nutzung privater Geräte nennen Angestellte die veralteten und zu langsamen Geräte in den Räumlichkeiten der öffentlichen Verwaltung. Die meisten Angestellten sind im Besitz von leistungsfähigeren und benutzerfreundlicheren Geräten, die viel lieber eingesetzt werden weil sie darüber hinaus persönlicher und individueller einsetzbar sind [Saw13].

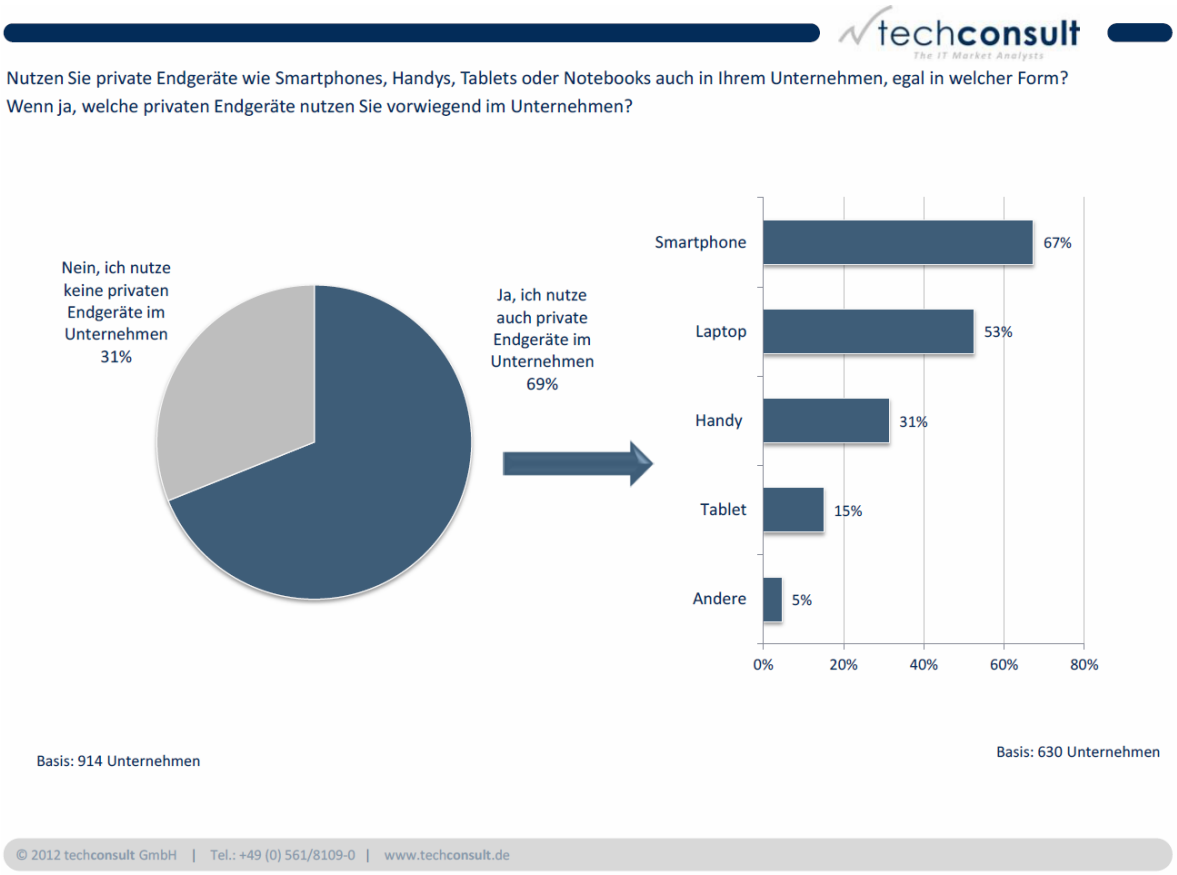

Abbildung 1.1: Auswertung einer Umfrage zur Nutzung privater Endgeräte im Unternehmen im Diagramm [Tec12].

1.3 Gliederung

Die Arbeit ist in folgender Weise gegliedert:

Kapitel 2 – Grundlagen: Hier werden die Grundlagen dieser Arbeit beschrieben.

Kapitel 3 – Konzept: Dieses Kapitel handelt über vorausgehende und verwandte Arbeiten.

Kapitel 4 – Implementierung und Anwendung: Hier wird auf das Gesamtkonzept eingegangen.

Kapitel 5 – Verwandte Arbeiten und Vergleich mit weiteren Lösungen: Dieses Kapitel zeigt die Umsetzung der Arbeit.

Kapitel 6 – Assessment: Abschließend die Beurteilung und Einschätzung.

Kapitel 7 – Zusammenfassung und Ausblick fasst die Ergebnisse der Arbeit zusammen und stellt Anknüpfungspunkte vor.

1.4 Ziele

Unternehmen lassen immer mehr ihre Mitarbeiter entweder von zu Hause arbeiten oder schicken sie direkt zum Kunden vor Ort. Damit die IT-Infrastruktur diesen Gegebenheiten stand halten kann, braucht es neuartige Ansätze zur Verwaltung relevanter Daten zentral im Unternehmen. Auch ist es notwendig dem Mitarbeiter eine Möglichkeit zu bieten auf diese Daten von überall aus zugreifen zu können. Durch den Einsatz mobiler Geräte durch den Mitarbeiter lassen sich diese Anforderungen erfüllen.

Allerdings sind es nicht nur portable Computer die der Angestellte im Außendienst nutzen kann. Auch moderne Smartphones sind mittlerweile technisch in der Lage viele Aufgaben zu lösen. Das Smartphone hat zwar zum großen Teil die Mobiltelefone ersetzt, allerdings wurde der Umfang an Funktionen im Smartphone um Vieles erweitert. Neben der Telefonie gehören E-Mails verfassen, Termine verwalten und eine Vielzahl an Sensoren zu den Ausstattungsmerkmalen eines modernen Smartphones. Diese Sensoren können zum Erfassen von Kontextbezogenen Daten genutzt werden, um ein auf den Nutzer abgestimmtes Arbeitserlebnis zu gestalten.

Werden Mitarbeiter nun mit solch moderner Technik vom Arbeitgeber ausgestattet, müssen Angestellte neben ihrem privaten Gerät ein ähnliches Gerät mit ähnlichen Funktionen jederzeit bei sich tragen. Oft sind diese Geräte an strenge Richtlinien gebunden und können von den eigenen Nutzungsgewohnheiten abweichen, was letztendlich zu Verwirrung beim Benutzer selbst führt. Aus diesem Grund befasst sich diese Arbeit mit der Nutzung privater Hardware im Unternehmen und dessen Vor- und Nachteile. Erfasst werden unter anderem die nötigen Sicherheitsvorkehrungen der entsprechenden IT-Abteilungen, als auch die Kosteneinsparungen auf Hardwareebene.

Ziel dieser Arbeit ist die Konzipierung und Entwicklung eines Lösungsansatzes für das ‘Bring-Your-Own-Device’-Problem. Als Grundlage dient das Android-Betriebssystem und dessen weitreichenden Funktionen und Sensoren. Durch Richtlinien werden jegliche Ressourcenzugriffe differenziert und entsprechend gewährt oder abgelehnt. Durch eine Beispielanwendung soll die Funktionalität der hier vorgestellten Lösung demonstriert und bewertet werden.

Im Anschluss sollen alternative Lösungen für das BYOD-Problem ermittelt und vorgestellt werden. Abschließend sollen diese mit der hier vorgestellten Lösung verglichen und bewertet werden.

2 Grundlagen

2.1 Vorangehende Modelle

BYOD ist im Grunde keine Neuerfindung, es ist mehr ein verändertes Modell des alten „use-what-you-are-told"-Konzeptes, kurz UWYT. Genau genommen ist es im direkten Vergleich das komplett gegensätzliches Modell, in welchem das Unternehmen die vollständige Kontrolle über die zur Verfügung stehenden Geräte hat. Allerdings stellt in diesem Fall der Arbeitgeber die Geräte und trägt somit die gesamten Hardwarekosten. Auch beim Einsatz einer BYOD-Lösung, kann vom Arbeitgeber ein Teil der Hardwarekosten übernommen werden. Bis zuletzt wurde vorwiegend auf den UWYT-Ansatz gebaut, allerdings folgen immer mehr Unternehmen dem Trend des BYOD-Konzeptes [Sin12].

2.1.1 Unterschiede

Die Unterschiede zwischen beiden Modellen lassen sich in verschiedenen Sektoren eines Unternehmens erkennen. Ein Vergleich, welcher zu Gunsten des BYOD-Ansatzes ausfällt, sind die Hardwarekosten, welche sich aufgrund der Einsparungen durch den Einsatz von persönlichen Geräten eines jeden Mitarbeiters drastisch senken lassen. Allerdings gibt es auf der anderen Seite Kostensteigerungen im Bereich der Sicherheit und Verwaltung für ein solch komplexes System wie es das BYOD-Modell ist. Personalkosten werden für Administratoren fällig, welche für die Einrichtung und den Support zuständig sein werden.

Auf Anwendungsebene wird einerseits durch eine limitierte Anzahl an Geräten und eingeschränkter Funktionalität nur das zur Verfügung gestellt, was wirklich benötigt wird und somit eine erhöhte Sicherheit vor jeglichem Missbrauch gewährleistet. Auf der anderen Seite muss eine Applikation auf unterschiedlichen Geräten lauffähig und somit an offene Standards angepasst sein. Auch muss zwischen Firmen- und Privatanwendungen unterschieden werden, um unerwünschten Zugriff beiderseits auszuschließen.

Auf Transaktionsebene werden Daten im UWYT-Modell zentral gesichert. Dabei gilt es Antivirensoftware auf dem aktuellen Stand zu halten und in gleichmäßigen Zeitintervallen eine Datensicherung auszuführen. Im Falle des BYOD-Modells müssen im gleichen Umfang Sicherheitsvorkehrungen getroffen werden, allerdings auf mehreren Systemen gleichzeitig. Voraussetzung für die strengen Sicherheitsvorkehrungen, die jeder Mitarbeiter zu befolgen hat, ist eine vorgeschriebene Sicherheitsklausel in der Firmenpolitik, sowie die benötigte Einverständniserklärung der Mitarbeiter für den nötigen Zugriff auf das private Gerät.

	UWYT	BYOD
Hardwarekosten	100 % Arbeitgeber	Arbeitnehmer (teilw. Arbeitgeber möglich)
Verwaltungskosten	niedriger	höher
Geräte	1 privat, 1 geschäftl.	1 für privat und geschäftl.

Tabelle 2.1: Vergleich der Konzepte UWYT und BYOD

Eine große Differenz lässt sich auch in der Anzahl der benötigten Mitarbeiter für den technischen Support festlegen. Zum einen muss mit mehr Aufwand für die Unterstützung verschiedener Geräte einkalkuliert werden als für eine festgelegte Menge. Des Weiteren müssen Mitarbeiter auf verschiedene Art und Weise abhängig vom eingesetzten System eingelernt werden, was zu höheren Personalaufwand führt.

2.2 Vorkehrungen

Bewegen sich Mitarbeiter in Einrichtungen des eigenen Unternehmens nutzen Sie nicht selten die Firmeneigene Infrastruktur um ihr Smartphone oder Notebook mit dem Internet zu verbinden. Die am häufigsten genutzte Methode dies zu bewerkstelligen ist die Verbindung über Wireless-LAN. Neben der Verschlüsselung durch den neuesten Standard, welcher aktuell WPA2 (Wi-Fi Protected Access 2) mit AES (Advanced Encryption Standard) ist, kann auch eine Tunnelung der Verbindungen zu jedem mobilen Gerät über VPNs (Virtual Private Networks) realisiert werden [Mer12].

Da die verwendeten Mittel im BYOD-Modell in erster Linie privat eingesetzt werden, ist der Verlust des Gerätes ein ernst zu nehmendes Risiko, welches ins Worst-Case Senario aufgenommen werden sollte. Dementsprechend sind Sicherheitsvorkehrungen für diesen Fall zu treffen. Kritische Situationen wie diese können auf verschiedene Arten gelöst werden. Durch entsprechende Maßnahmen kann schon das Ändern des Passwortes des verknüpften Email-Accounts für den benötigten Schutz ausreichen. Noch sicherer sind jedoch entsprechende Vorkehrungen zur Ferngesteuerten Sperrung oder sogar Löschung des jeweiligen Gerätes [Sha13].

Natürlich darf bei einem solchen Vorhaben nicht der Schutz vor Viren, Malware und Phishing-Attacken fehlen. Deshalb sollte stets auf aktuelle Antiviren und Malware Programme gesetzt und vor neuartigen Phishing Methoden gewarnt und informiert werden.

Um das Ereignis eines Ausfalls des Firmenservers oder dergleichen zu entgehen, müssen mit dem Wachstum der Anzahl von verbundenen Endgeräten auch die Kapazitäten der Server erweitert werden. Denn mit dem größer werdenden Zugriff auf die Firmeneigenen Ressourcen, steigt der Anspruch an die entsprechende Hardware und kann zu einem Ausfall der Infrastruktur führen.

2.3 Risiken

Die Risiken, welche BYOD mit sich bringt sind jedem Unternehmen bereits länger bekannt. Sie unterscheiden sich allerdings geringfügig in der Umsetzung. Früher war es ein Speichermedium in Form eines USB-Sticks, welches in einem unaufmerksamen Augenblick abhanden kommen kann. In diesem neuartigen Ansatz liegt die Gefahr ebenso in einem Speichermedium, allerdings in Gestalt eines Smartphones.

Durch die ständige Verbindung des mobilen Gerätes hat sich die Gefahr des Missbrauchs um ein vielfaches vergrößert. Sensible Daten könnten entweder während der Übertragung abgefangen werden, oder durch Manipulation des Gerätes umgeleitet werden.

Hinzu kommt das Problem des breiteren Spektrums an Betriebssystemen, welches nun unterstützt und überwacht werden muss. Wo früher auf ein oder wenige Betriebssysteme gesetzt wurde, um Kompatibilitätsprobleme aus dem Weg zu gehen, muss nun, da die Auswahl des Gerätes und damit auch des Betriebssystems meist dem Mitarbeiter überlassen wird, jedes Betriebssystem unabhängig voneinander auf Sicherheitslücken geprüft und verbessert werden.

Folgende typische Funktionen werden meist in die BYOD-Verwaltungssoftware mit eingepflegt um einige der Risiken zu entschärfen:

- Lokalisierung eines Gerätes
- Verschlüsselung der Daten und der Kommunikationswege
- Passwort- oder PIN-geschützter Zugriff
- Ferngesteuerte Verwaltung zum Sperren oder Löschen eines Gerätes bei Verlust
- Überwachung des Umgangs mit Firmendaten

Viele, der hier genannten Punkte erfordern jedoch die Einwilligung eines jeden Mitarbeiter, da diese einen gewaltigen Schnitt in die Privatsphäre bilden. Deshalb gilt es auch hier abzuwägen, welche Risiken können zum Teil in Kauf genommen oder anderweitig gelöst werden, und was kann das eigene Unternehmen seinen Mitarbeitern zumuten. Auch werden sich Mitarbeiter weniger über die Überwachung und des ferngesteuerten Löschen des eigenen Gerätes freuen.

2.4 Rechtliche Bedenken

Bei der Implementierung eines BYOD-Konzeptes in einem Unternehmen bekommen die Rechte der Angestellten auf Privatsphäre wenig Berücksichtigung. In erster Linie werden die eigenen Firmendaten auf die sicherste Weise verschlüsselt um Datenmissbrauch und Verlust zu verhindern, denn schließlich haftet bei solch einem ungünstigen Fall immer noch das Unternehmen. Nichtsdestotrotz gilt es auch die Daten und Aktivitäten der Angestellten auf

ihrem eigenen Gerät zu respektieren und eine Lösung zu finden, ohne die Rechte auf private Ressourcen zu verletzen [Hal13].

2.4.1 Mobile Police im Unternehmen

Um sich Rechtlich auf beiden Seiten abzusichern ist es ratsam einen Rahmenvertrag aufzusetzen. Dies gewährleistet auch eine erhöhte Vorsicht und Achtsamkeit beider Seiten auf die Daten und Privatsphäre der Gegenseite. In dieser Police werden alle enthaltenen Funktionen und Rahmenbedingungen für den Einsatz eines privaten Gerätes in einem Unternehmen dargelegt und mögliche Konsequenzen aufgeführt. Dazu gehört auch die Definition von Funktionen wie das oben angesprochene ferngesteuerte Löschen des Gerätes bei Verlust oder Missbrauch. Abhängig von Region und Land müssen unterschiedliche Gesetze beachtet und niedergeschrieben werden. Dazu gehört auch die explizite Zustimmung und Inkenntnisnahme eines jeden Mitarbeiters auf den möglichen Zugriff auf private Daten durch Installation der BYOD-Software auf dem eigenen Gerät. Damit es zu einer akzeptablen Lösung für beide Parteien kommt, müssen Kompromisse eingegangen werden. Diese sind zum Beispiel die freie Wahl eines jeden Mitarbeiters auf ein von ihm gewünschten Gerät und die Freigabe der nötigen Schritte auf Diesem. Ebenso werden von jedem Mitarbeiter eine dementsprechende Verantwortung und Pflicht gegenüber den Daten des Unternehmens eingeräumt.

Wie schon weiter oben erwähnt muss sich jedes Unternehmen mit den rechtlichen Gegebenheiten in der Region auseinandersetzen. Natürlich gelten je nach Land unterschiedliche Grundsätze. Folgende Verordnungen sind allerdings in den meisten Rechtsprechungen wie folgt verankert:

- Es muss eine eindeutige Zustimmung von jedem Mitarbeiter eingeholt werden bevor Zugriff auf persönliche Daten und Aktivitäten durch Unternehmenssoftware möglich ist
- Technische Sicherheitsvorkehrungen müssen jederzeit gewährleistet sein, um persönliche Daten zu schützen. Entsprechende Verfahren sind zum Beispiel Verschlüsselung der Daten und / oder des Kommunikationskanal.

2.5 Aspekte der Sicherheit

Zu jedem Problem, welches sich bei der Erstellung einer BYOD-Verwaltungsplattform stellt, müssen Entscheidungen abgewogen und ausgewählt werden. In diesem Abschnitt wird auf die einzelnen Themen eingegangen und mithilfe gezielter Fragestellungen die für diese Arbeit beste Umsetzung ausgewählt. Allerdings werden mehr Aspekte aufgeführt, als in dieser Arbeit sowohl aus Zeitgründen, als auch aufgrund der Tatsache, dass es sich hierbei um ein Beispiel handelt und es keine ausschlaggebende Datengrundlage gibt.

Um sensible Daten eines Unternehmens vor ungewollten Zugriff zu schützen muss generell ein wesentlicher Schutz gewährleistet sein. Dies kann ein einfacher PIN-Code sein,

der bei falscher Eingabe den Zugriff verweigert, oder auch die Beeinträchtigung einiger Funktionen des Gerätes um Missbrauch vorzubeugen. Diese Arbeit verwendet eine Plattform auf Grundlage des Android-Betriebssystems, welches Anwendungen vordefinierte Zugriffsrechte auf die Funktionen des Mobiltelefons gewähren. Diese können unabhängig von jedem Unternehmen den Ansprüchen entsprechend angepasst und entwickelt werden. Zusätzlich schützt die individuelle Bearbeitung dieser Rechte ein zusätzlicher PIN-Code vor ungewollten Anpassungen [Deh13].

Bei Verlust oder Diebstahl kann ein Mechanismus zum Einsatz kommen, welcher sensible Daten oder sogar den kompletten Inhalt des Gerätes löscht. Dies kann entweder ferngesteuert oder bei mehrmaliger falscher Eingabe vordefinierter Sicherheitscodes vonstatten gehen. In dieser Arbeit wird kein solches Verfahren eingesetzt und kann, bei Bedarf, individuell in weiteren Arbeiten ergänzt werden. Auch kann nur eine vorübergehende Sperre bei falscher PIN-Eingabe erfolgen, welcher auf ungewollte Weise ausgelöst werden kann.

2.5.1 Privatsphäre der Mitarbeiter

Bei dem Einsatz von jeglicher BYOD-Software und der genutzten Sicherheitsvorkehrungen auf einem privaten Gerät eines Mitarbeiters sollte sich jedes Unternehmen über die Rechte dieser auseinandersetzen. Diese sind von der Region abhängig, in der das Unternehmen wirtschaftet, allerdings fallen diese größtenteils ähnlich aus wodurch auf folgende Aspekte beim Einsatz von BYOD-Software Wert gelegt werden sollte:

- Jedem Mitarbeiter sollte bewusst sein, welche Daten und Anwendungen auf dem eigenen Gerät überwacht werden. Dazu gehören auch die implementierten Vorgehensweisen bei Verlust des Gerätes oder Austritt aus dem Unternehmen. Dies wird im besten Fall schriftlich festgehalten und beiderseits unterzeichnet.
- Auch wenn eine BYOD-Software darauf ausgelegt ist, private Daten bei der Erfassung auszuschließen, gilt es seine Mitarbeiter darüber in Kenntnis zu setzen, dass die Möglichkeit besteht auf private Daten Zugriff zu erhalten. Dies sichert das Unternehmen für die Zukunft ab, um rechtliche Klagen aus dem Weg zu räumen.
- Bevor der Einsatz einer BYOD-Software bei einem Mitarbeiter Verwendung findet, sollte die Bestätigung, meist in Form einer Unterschrift, von diesem eingeholt werden. Auch kann durch eine Art Bestätigung in der BYOD-Software selbst die Zustimmung der Richtlinien eingeholt werden. Hierzu sollten allerdings unbedingt die rechtlichen Möglichkeiten aufgrund der Gesetzeslage im jeweiligen Land beachtet werden.

Diese Punkte sind alle speziell auf Unternehmensebene einzubinden und haben in dieser Arbeit keine Umsetzung gefunden.

2.5.2 Einsatzgebiet

Nicht jeder Mitarbeiter eines Unternehmens wird sich in dem Szenario wiederfinden, sein privates, mobiles Gerät im Arbeitsumfeld nutzen zu wollen oder müssen. Vor allem wenn es keinen positiven Effekt nach sich zieht, kann es dem einen oder anderen zur Last fallen und das Arbeiten mit negativer Wirkung beeinflussen. Somit gilt es zu unterscheiden in welcher Position und Abteilung der Einsatz einer BYOD-Umgebung Sinn ergibt. Darüber hinaus muss festgelegt werden, ob sich für verschiedene Positionen und Abteilungen in einem Unternehmen unterschiedliche Funktionen und Berechtigungen innerhalb der BYOD-Software ergeben.

In dieser Arbeit werden Regelungen in Form einer Config-Datei über den Firmenserver geladen, welche vom Mitarbeiter in geringer und festgelegter Weise angepasst werden dürfen, um auch den eigenen Bedürfnissen gerecht zu werden.

2.5.3 Gesetzeslage

Am 25. August 2010 wurde von der Bundesregierung das Gesetz zur Regelung des Beschäftigungsdatenschutzes beschlossen als Ergänzung zum bereits bestehenden Bundesdatenschutzgesetz. Damit soll die Mitarbeiterüberwachung in Deutschland eingedämmt und erschwert werden. Arbeitgeber erwarten sich damit Geschäftsprozesse zu optimieren und Verhalten der Mitarbeiter zu kontrollieren.

Allerdings zeigen Fälle aus vergangenen Jahren, dass die Privatsphäre der Mitarbeiter dadurch gefährdet wird. Im Jahr 2009 veröffentlichte das Magazin Stern einen Artikel über Mitarbeiterüberwachung bei der Deutschen Bahn AG. Sowohl Mitarbeiter als auch dessen Partner wurden mittels einer Detektei überwacht um Korruption in den eigenen Reihen auszuschließen. Unter den Beschatteten, welche sich auf mehr als 1000 Personen beziffern, waren größtenteils Mitarbeiter aus dem oberen Managements. Laut der Deutschen Bahn AG versprachen Sie sich mit diesem Vorgehen falsche Angaben bezüglich Nebeneinkünfte und Beteiligungen an anderen Firmen aufzudecken.

Im Falle der privaten Nutzung von Telefon und Internet im Unternehmensumfeld, sofern diese erlaubt sind, ist der Arbeitgeber auf Gesetzesebene als Telefon- und Internet-Dienstleistungsanbieter anzusehen und steht somit unter der Einhaltung des Datenschutzgesetzes. Diese sagen aus, dass die Überwachung des privaten Email- und Telefonverkehrs untersagt ist, sollte die Trennung zum geschäftlichen Verkehr nicht eindeutig identifizierbar sein. Dies ist oft der Fall bei Geräten auf denen sowohl private als auch geschäftliche Aufgaben erledigt werden. Deshalb gilt bei der Umsetzung und Anwendung von BYOD-Technologien eine erhöhte Beachtung der Gesetzeslage.

2.6 Speichermethoden

Zur Speicherung von Programm- und Nutzerdaten gibt es auf einem Gerät viele Möglichkeiten. In der Regel existiert neben dem frei zugänglichen Speicher auch ein Programmspeicher, auf welchem nur ausgewählte Pakete Zugriff haben. Die Verwendung des jeweiligen Speichers hängt immer von der Art der Nutzung und des zu speichernden Inhalts ab. Sensible Daten werden im Gegensatz zu öffentlich Zugänglichen als privat gekennzeichnet und dementsprechend an einem sichereren Ort abgelegt. Auch kommt es auf die Größe der jeweiligen Datenmengen an. Meist fällt der Programmspeicher im Vergleich zum gemeinsam genutzten Externen Speicher kleiner aus und hat nur ein begrenztes Volumen. In den folgenden Abschnitten werden die Möglichkeiten dargelegt, welche sich für die Datenspeicherung auf einem Android-Mobiltelefon anbieten [Goo12a].

2.6.1 Programmspeicher

Der Programmspeicher wird individuell von jeder Applikation erstellt und verwaltet. Alle Dateien in diesem Bereich sind privat und somit nicht aus anderen Programmpaketen abrufbar. Bei der Deinstallation einer Applikation werden die Daten im dazugehörigen Programmspeicher gelöscht und somit sensible Daten vernichtet.

Um eine Datei in den Programmspeicher zu erstellen und darin zu schreiben benötigt es eine Instanz des `java.io.FileOutputStream`. Listing 2.1 zeigt den Ausschnitt, welcher für das Schreiben eines Strings in den Programmspeicher verantwortlich ist. Bei der Erstellung des FileOutputStreams wird neben dem Dateinamen ein weiterer Parameter übergeben. Der zweite Parameter konnte noch vor dem Android API-Level 17 mehrere Werte annehmen, wie zum Beispiel `MODE_WORLD_READABLE` um einzelne Dateien außerhalb des Programms zur Verfügung zu stellen. Mittlerweile ist der Programmspeicher standardmäßig privat und macht den Wert `MODE_PRIVAT` fast überflüssig. Einziger Unterschied zum Wert `MODE_APPEND` ist, dass die Datei mit dem angegeben Dateinamen, falls vorhanden, überschrieben wird, statt der zu schreibende Wer angehängt wird.

Damit von einer Datei aus dem Programmspeicher gelesen werden kann, wird ein Objekt des `java.io.FileInputStream` benötigt. Diesem wird der Name der zu lesenden Datei mittels der Funktion `openFileInput()` übergeben, und anschließend mit `read()` gelesen. Sowohl nach dem Lesen, als auch nach dem Schreiben ist es nötig den Stream mit `close()` zu schließen, um den Zugriff auf die Datei und den Speicherplatz wieder freizugeben.

Die Dateien, welche im Programmspeicher abgelegt werden, lassen sich im Hauptspeicher eines Android-Mobiltelefons unter `/data/data/paketname/files/` wiederfinden. Der Paketname entspricht immer dem entsprechenden Namen der Applikation.

Listing 2.1 Schreiben in den Programmspeicher.

```
FileOutputStream fos = openFileOutput(FILENAME, Context.MODE_PRIVATE) ;
fos.write(STRING.getBytes());
fos.close();
```

2.6.2 Shared Preferences

Das Interface `android.content.SharedPreferences` stellt ein Framework in der Android API zur Verfügung, welches zur Speicherung von primitiven Daten genutzt werden kann. Dabei werden zum Schreiben und Auslesen Schlüsselwort-Wert Paare zur eindeutigen Identifikation verwendet. Die gespeicherten Daten werden ebenfalls in den Internen Speicher des Gerätes geschrieben und in einem Unterordner mit dem Names `sharedprefs` des entsprechenden Paketes abgespeichert. Somit sind die Daten der Shared Preferences persistent und bleiben, auch nach Neustart der Anwendung oder des Gerätes, erhalten.

Für den Abruf einer Instanz der Shared Preferences wird die Funktion `getSharedPreferences()` zur Verfügung gestellt. Dieser werden zwei Parameter übergeben. Der erste Parameter dient zur Identifikation des Preference-Objektes und ist vom Typ String. Das heißt es können mehrere Preference-Objekte einer Applikation gleichzeitig existieren. Wird nur ein Preference-Objekt benötigt so kann alternativ die Funktion `getPreferences()` verwendet werden, wobei der Activity-Klassenname als Preference-Name standardmäßig verwendet wird. Ähnlich wie in Abschnitt 2.6.1 wird mit dem zweiten Parameter der Modus beschrieben, welcher jedoch mittlerweile in Version 17 der Android API nur noch einen gültigen Wert annehmen kann. Dieser Wert lautet `MODE_PRIVATE` bzw. 0.

Folgende Datentypen können in den Shared Preferences abgespeichert werden: `Boolean`, `Float`, `Int`, `Long` und `String`. Die entsprechenden Funktionen für das abspeichern lauten: `putBoolean()`, `putFloat()`, `putInt()`, `putLong()` und `putString()`. Listing 2.2 zeigt den Ablauf für das Abspeichern in die Shared Preferences. Dazu wird ein Objekt vom Typ SharedPreferences.Editor, welcher über die Funktion `edit()` des Preference-Objektes abgerufen werden kann. Der Speicherfunktionen werden zwei Parameter übergeben. Zuerst gilt es mit dem ersten Parameter, welcher vom Typ String ist, den Wert eindeutig zu identifizieren. Der zweite Parameter beinhaltet dann den zu schreibenden Wert. Abgeschlossen wird das Speichern mit dem Aufruf von `commit()` auf das Editor-Objekt.

Analog zu den Schreibfunktionen heißen die Auslesefunktionen wie folgt: `getBoolean()`, `getFloat()`, `getInt()`, `getLong()` und `getString()`. Im Vergleich zum Schreibaufwand ist das Auslesen eines Wertes aus den Shared Preferences um einiges schneller wie im Listing 2.3 zu sehen ist. Ist das Objekt der Shared Preferences erst mal instanziiert, so kann über die entsprechende Funktion der gewünschte Wert unter Angabe des Identifikations-String und einem Default-Wert ausgelesen werden. Ist kein Wert hinter dieser Schlüsselwort abgespeichert, so wird der Default-Wert angenommen.

Listing 2.2 Speichern von primitiven Datentypen in Shared Preferences.

```
SharedPreferences settings = getPreferences(MODE_PRIVATE);
SharedPreferences.Editor editor = settings.edit();
editor.putBoolean("expertmode_enable", true);
editor.commit();
```

Listing 2.3 Auslesen aus Shared Preferences.

```
SharedPreferences settings = getPreferences(MODE_PRIVATE);
boolean expertmode = settings.getBoolean("expertmode_enable", false);
```

2.6.3 SQLite Datenbank

Eine weitere Möglichkeit Daten für eine Applikation auf einem Android-Mobiltelefon zu speichern ist die Verwendung von Datenbanken. Hierzu bietet Android den vollen Funktionsumfang der SQLite Datenbanken. Die Datenbanken werden ebenso wie die Shared Preferences in einem Unterordner des Applikationsspeichers geschrieben und sind nur aus den Anwendungen dieses Programmpaketes abrufbar.

Die empfohlene Methode zur Erstellung einer neuen Datenbank ist in Listing 2.4 beispielhaft dargestellt. Dabei wird eine Unterklasse von `android.database.sqlite.SQLiteOpenHelper` erstellt und die beim Instantiieren ausgeführte Methode `onCreate()` überschrieben. Dieser Funktion wird vom System ein Objekt der Klasse `android.database.sqlite.SQLiteDatabase` übergeben, auf welches anschließend durch `execSQL()` SQLite-Befehle ausgeführt werden kann. Über den Konstruktor (hier: `DatabaseHelper()`) kann in weiteren Klassen der Applikation ein Objekt des `SQLiteOpenHelper` instantiiert werden um durch `getWritableDatabase()` und `getReadableDatabase()` weitere SQLite-Operationen auszuführen.

Listing 2.4 Erstellen einer SQLite Datenbank

```
public class DictionaryOpenHelper extends SQLiteOpenHelper {

    private static final int DATABASE_VERSION = 2;
    private static final String DICTIONARY_TABLE_NAME = "BYOD";
    private static final String DICTIONARY_TABLE_CREATE =
                "CREATE TABLE " + DICTIONARY_TABLE_NAME + " (" +
                KEY_WORD + " TEXT, " +
                KEY_DEFINITION + " TEXT);";

    DictionaryOpenHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(DICTIONARY_TABLE_CREATE);
    }
}
```

Queries auf SQLite Datenbanken können über die Funktion `query()` ausgeführt werden. Bei komplexeren Anfragen kann auf den `android.database.sqlite.SQLiteQueryBuilder` zurückgegriffen werden. Dabei werden alle Ergebnisse als `android.database.Cursor` zurückgegeben. Der Cursor ist ein Zeiger mit welchem durch die einzelnen Reihen durch-navigiert werden kann.

2.6.4 Externer Speicher

Der externe Speicher eines Android-Gerätes befindet sich entweder ebenfalls wie der interne Speicher fest im Smartphone verbaut, oder kann zusätzlich über ein externes Speichermedium erweitert werden. Dieser Umstand hat zur Folge, dass ein Teil oder sogar der komplette externe Speicherplatz bei Entnahme des Speichermediums nicht jederzeit zur Verfügung steht.

Auch aufgrund der Möglichkeit zur Entnahme des Speichermediums ist der Externe Speicher ein geteilter Speicher, der von jeder Anwendung und jedem Nutzer ausgelesen werden kann. Somit sind alle auf diesem Medium gespeicherten Daten nicht vor unrechtmäßigem Zugriff geschützt. Da der Interne Speicher eine eingeschränkte Größe hat, weichen viele Applikationen auf den externen Speicher aus. Ist dies bei empfindlichen Daten erforderlich, so lässt sich nur durch Verschlüsselungsmethoden eine gewisse Sicherheitsstufe aufbauen.

Bei jeder Interaktion mit dem externen Speicher ist es sicherzustellen, ob das Medium bereit, eingelegt oder beschreibbar ist. Dies lässt sich über die Funktion `getExternalStorageState()` feststellen, welche sich über das Objekt `android.os.Environment` ausführen lässt. Zurückgegeben wird ein String, welcher einem der vielen hinterlegten Strings der Klasse Environment entspricht. Hier ein Auszug aus den am häufigsten auftretenden Rückgabewerte:

- `MEDIA_MOUNTED` - Medium ist bereit und am gemounteten Ort beschreibbar.
- `MEDIA_MOUNTED_READ_ONLY` - Medium ist bereit und am gemounteten Ort nicht beschreibbar.
- `MEDIA_REMOVED` - Externer Speichermedium nicht eingesetzt.
- `MEDIA_SHARED` - Speicher eingelegt, allerdings nicht gemountet, da über die Funktion USB-Massenspeicher, der Speicher einem angeschlossenem Computer zugänglich gemacht ist.
- `MEDIA_NOFS` - Medium vorhanden, allerdings wird das Dateisystem nicht unterstützt.
- Alle weiteren Rückgabewerte sind unter [Goo12a] zu finden.

Seit Android API Level 8 kann über die Funktion `getExternalFilesDir()` der Activity auf Dateien zugegriffen bzw. abgespeichert werden. Der Methode wird ein Parameter vom Typ String übergeben, welcher den Pfad zu diesen Dateien definiert. Wird der Funktion `null` übergeben, so wird automatisch der Unterordner im externen Speicher erstellt und

übermittelt. Wird die API Level Version 7 oder vorher verwendet, so ist nur das Root-Verzeichnis des externen Speichers über die Funktion `getExternalStorageDirectory()` abrufbar. Um nun die Daten der Applikation in dem dazugehörigen Unterordner zu finden, muss zum entsprechenden Unterordner wie folgt `/Android/data/<package_name>/files/` navigiert werden. Zu beachten ist allerdings, dass alle Dateien, auch wenn Sie sich im dazugehörigen Unterordner befinden, von allen Programmpaketen aus veränderbar und auslesbar sind. Wird ein Programm vom Gerät entfernt, so werden die dazugehörigen Daten im Unterordner des entsprechenden Paketes gelöscht.

Sollen Bilder, Musik, Videos oder sonstige Dateien nicht in dem jeweiligen Programmordner abgelegt werden, so kann der Funktion einer der vielen vordefinierten Werte übergeben werden, welche die Dateien anschließend in den dazugehörigen Ordner abspeichert. Für Musikdateien oder Bilder können die Werte `DIRECTORY_MUSIC` oder `DIRECTORY_PICTURES` verwendet werden, welche den Ordner `/Music/` oder `/Pictures/` auf dem externen Speichermedium angibt. Analog gilt dies für die Werte `DIRECTORY_PODCASTS`, `DIRECTORY_RINGTONES`, `DIRECTORY_ALARMS` oder `DIRECTORY_MOVIES`. Diese Dateien sind von der Löschung nach Deinstallation des entsprechenden Programms geschützt, da Sie sich nicht im Unterordner der Applikation befinden.

Entsprechend des Ordners für Dateien einer Applikation kann der Ordner namens `/Android/data/<package_name>/cache/` zum Zwischenspeichern genutzt werden. Ab dem API Level 8 kann die Funktion `getExternalCacheDir()` behilflich sein. Für die Versionen davor muss wie gewohnt über `getExternalStorageDirectory()` dorthin navigiert werden.

2.7 Zugangskontrolle

Auch wenn viele Mechanismen zum Schutz von privaten als auch firmeneigenen Daten auf einem mobilen Gerät eingesetzt werden können, helfen diese Maßnahmen meist nicht unmittelbar nach dem Abhanden kommen des Gerätes. Einzig eine Zugangskontrolle zu den Daten kann hier Abhilfe schaffen.

Beim Einsatz von zusätzlicher Software für eine Datenverwaltung ist es mittlerweile öfters der Fall, dass nach einem Passwort gefragt wird. Gelangt ein Fremder allerdings Zugriff auf ein eingeschaltetes Gerät, so kann dieser in den meisten Fällen alle privaten Daten einsehen und durchsuchen. Aus diesem Grund besitzen Laptops und Smartphones beim Start eine Passwort- oder PIN-Code-Abfrage. Diese soll in erster Linie den Zugang Dritter zum Gerät verhindern und so die persönlichen Daten des Nutzers schützen.

2.7.1 Sperrbildschirm

Android besitzt neben der PIN-Code-Abfrage beim Start einen Sperrbildschirm welcher sich unmittelbar nach dem Aufwecken des Gerätes aus dem Standby und nach der PIN-Abfrage zeigt. Dieser kann in den Sicherheitseinstellungen von Android angepasst und zwischen

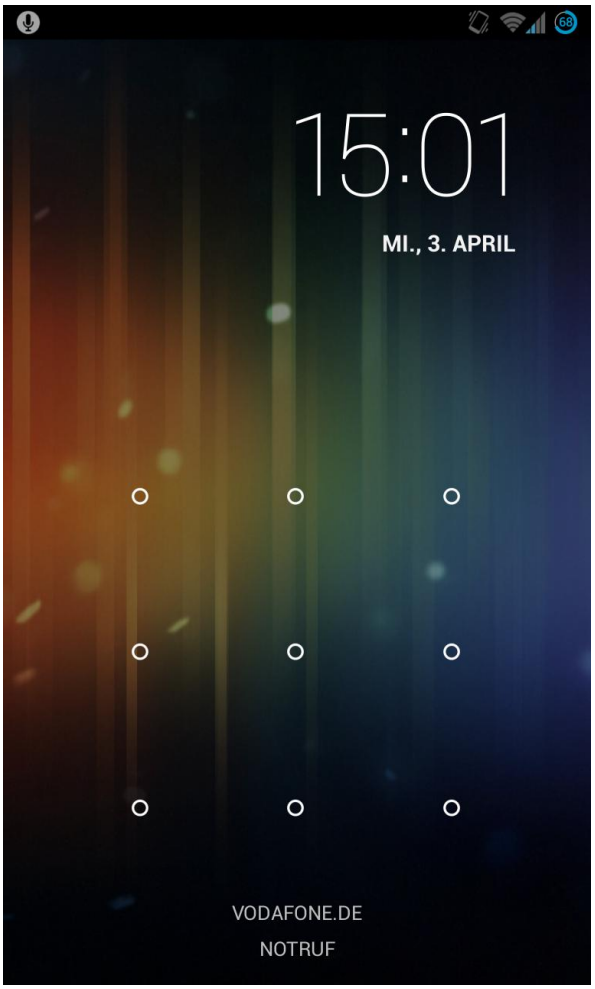

Abbildung 2.1: Sperrbildschirm mit dem Einsatz eines Musters.

verschiedenen Techniken und Funktionen gewählt werden. Auch kann diese sofern der Nutzer sich nicht bedroht oder keine Daten zu schützen hat komplett deaktiviert werden.

Die erste Sperrmethode ist die PIN-Code-Methode, wie sie in Abbildung 2.1 zu sehen ist. Diese hat den gleichen Funktionsumfang wie die Abfrage der PIN einer SIM-Karte, und kann unabhängig davon einen anderen Code beinhalten. Wird das Gerät nun aus dem Aus-Zustand hochgefahren, so muss nach dem Startvorgang zuerst die PIN der SIM-Karte eingegeben und anschließend der Entsperr-Code bestätigt werden. Bei jedem erwecken des Gerätes aus dem Standby-Betrieb ist nur noch der Entsperr-Code nötig.

Ein ähnliches Verfahren ist das Sperren mittels eines Passwortes. Dabei kann der Benutzer eine beliebige Zeichenkette aus Buchstaben, Zahlen und Sonderzeichen mit der Mindestlänge 4 wählen.

Mit dem Muster als Entsperr-Technik hat Google ein neuartiges und innovatives Verfahren für Touchscreens vorgestellt. Dabei fährt der Nutzer mit dem Finger in einem imaginären Quadrat mindestens vier der neun abgebildeten Punkte nacheinander ab, ohne den Finger zu heben. Die Punkte sind gitterartig angeordnet und sind gleichmäßig verteilt. Zusätzlichen Schutz bietet das Deaktivieren der Funktion 'Muster sichtbar machen', welche die Spur beim abfahren nicht anzeigt.

Mit dem Release von Android 4.0 'Ice Cream Sandwich' wurde Ende 2011 eine weitere Methode zur Nutzung des Sperrbildschirmes hinzugefügt. Dabei wird während der Einrichtungsphase ein Portrait-Foto vom Nutzer mittels der Frontkamera geschossen und anschließend beim Zugriffsversuch mit der Person vor dem Smartphone verglichen. Laut Google wird hierbei die neueste Technologie für Gesichtserkennung eingesetzt. Allerdings wurde schnell eine Sicherheitslücke entdeckt, welche es zulässt, die Funktion Face-Unlock mit einem Foto der Zielperson zu überlisten. Dabei wird das Foto beim Entsperren dem Smartphone gegenüber gehalten und somit dem Gerät suggeriert, es wird von der erkannten Person bedient.

Mitte 2012 wurde dann mit dem Erscheinen von Android 4.1 'Jelly Bean' die Sicherheitslücke vermeintlich geschlossen indem die Funktion um den 'Liveness Check' erweitert wurde. Diese Funktion erweitert den Entsperrprozess um die Erkennung einer Menschlichen Geste, und zwar dem Blinzeln. Hat das Gerät die Person vor dem Smartphone erkannt, so wird er nun aufgefordert seine Augen kurz zu schließen und wieder zu öffnen. Allerdings ließ sich diese Methode ebenfalls austricksen. Durch Verwendung von zwei Bildern, lässt sich eine Person so darstellen, als hätte Sie geblinzelt. Dabei muss beim zweiten Bild die Augen unkenntlich gemacht werden und Wimpern eingezeichnet werden, um geschlossene Augenlider zu suggerieren. Anschließend könne die Portrait-Bilder abwechselnd dem Gerät vorgehalten werden.

Sollte die Gesichtserkennung aus verschiedenen Gründen die Person nicht erkennen, so kann auf ein anderes Entsperr-Verfahren ausgewichen werden, welches vorher während der Einrichtung gewählt und eingestellt wurde. Einer dieser Umstände, aufgrund welchem keine Gesichtserkennung möglich sein kann, sind die Lichtverhältnisse. Bei schlechtem Licht oder bei Dunkelheit ist es generell nicht Möglich eine Person zu erkennen und somit ist das Ausweichen auf eine andere Technik nötig.

In den Sicherheitseinstellungen von Android finden sich weitere Einstellmöglichkeit für die Feinjustierung. Eine davon ist die sofortige Sperre des Gerätes bei Betätigung des Power-Knopfes um das Gerät in den Standby-Modus zu schicken. Ist dies nicht gewünscht, so kann der Haken weggeklickt werden. In diesem Fall wird das Gerät erst gesperrt, sobald die Zeit, welche unter 'Automatisch sperren nach' steht, abgelaufen ist.

2.7.2 Zugangsschutz einer Applikation

Wie schon in Abschnitt 2.7 beschrieben, bieten nur sehr wenige Sicherheitsmaßnahmen kompletten Schutz vor unrechtmäßigem Zugriff auf die Daten eines Nutzers. Aus diesem Grund werden des öfteren weitere Schutzvorrichtungen in einzelnen Applikationen eingerichtet.

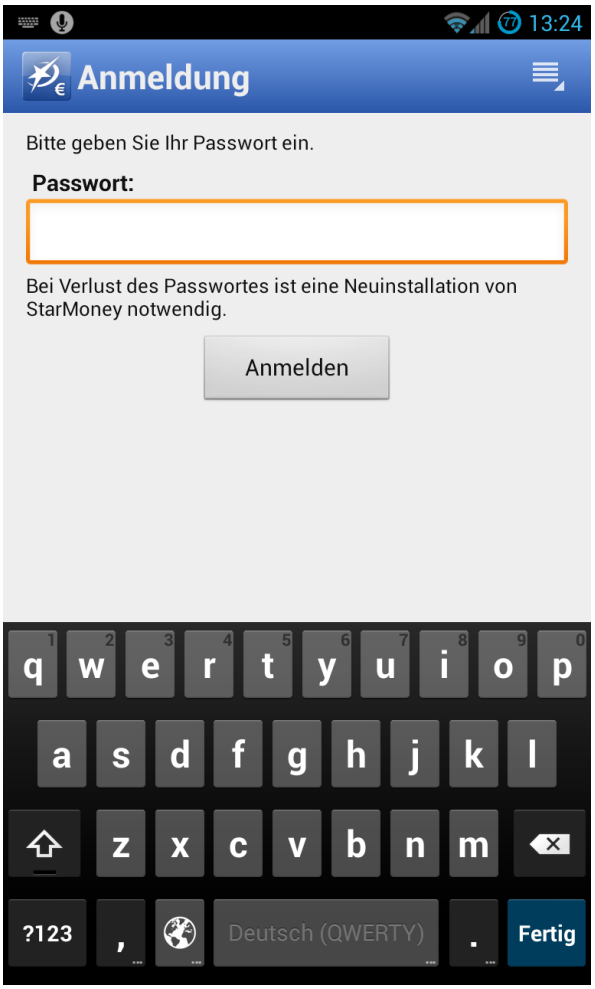

Abbildung 2.2: Passwortschutz der Applikation StarMoney.

Zumeist werden ähnliche Methoden wie auf dem Sperrbildschirm genutzt. Starmoney, eine Anwendung zur Bankkontenverwaltung, schützt zusätzlich zur üblichen Passwortabfrage eines jeden Online-Banking Portals die Applikation durch eine weitere Passwortabfrage. Abbildung 2.2 zeigt den Bildschirm des Gerätes nach Start der Applikation.

Auch Dropbox, ein weltweit agierender Cloud-Speicher-Dienst, schützt die Daten seiner Nutzer durch einen extra Schutzmechanismus in Form einer PIN-Code-Abfrage. Wie Abbildung 2.3 zeigt ist der Nutzer im Vergleich zu einer Passwortabfrage eingeschränkter durch die Festlegung auf eine Zahlenkombination mit genau vier Ziffern.

Durch die eingeleiteten Sicherheitsmaßnahmen versuchen Unternehmen sensible und private Daten vor Fremdeingriff zu schützen. Im Fall von Dropbox werden sämtliche Nutzerdaten nach mehrmaliger falscher PIN-Eingabe aus Sicherheitsgründen gelöscht.

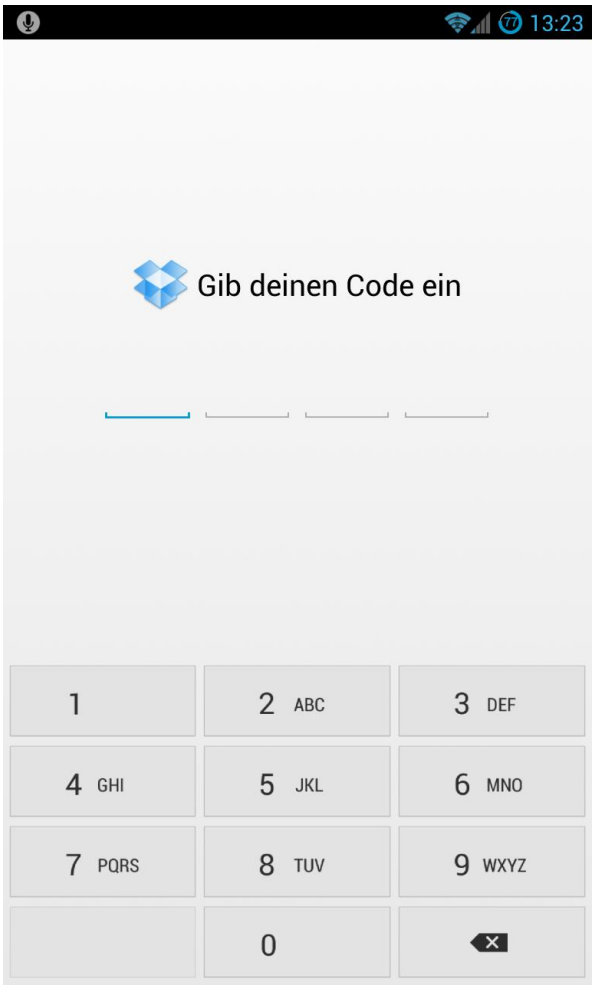

Abbildung 2.3: PIN-Zugangsschutz der Applikation Dropbox.

2.8 Sicherheitskonzept

Um die Sicherheit in einem offenen System, wie es das Betriebssystem Android ist, zu gewährleisten, braucht es ein robustes Sicherheitskonzept und eine gute Umsetzung. In der Architektur von Android finden sich mehrere Sicherheitsschichten, welche die nötige Flexibilität eines offenen und weit verbreiteten Systems liefert.

Neben dem Nutzer helfen die integrierten Sicherheitsmaßnahmen auch dem Entwickler. Android bietet ein Interface für die Anpassung verschiedener Sicherheitsmaßnahmen einer Applikation. Werden keine Veränderungen vorgenommen, so wird für weniger erfahrene Entwickler die Standardeinstellung für Sicherheit übernommen [Goo13b].

Die wichtigsten Sicherheitsaspekte von Android sind die Sicherheit von Benutzerdaten, Systemressourcen und die Isolation von Applikationen. Um diese Ziele umzusetzen sind

mehrere Vorkehrungen zu treffen. Gelöst wurde dies in dem von Google entwickelten Betriebssystem unter anderem durch die Sicherheitsmaßnahmen im Linux-Kernel, welcher dem Android Betriebssystem zugrunde liegt. Außerdem werden alle Applikationen in einer Sandbox gehalten und haben so keinen Einfluss auf fremde Daten. Des Weiteren werden Zugriffe auf erweiterte Funktionen für eine Applikation gesondert gekennzeichnet und vom Benutzer die Erlaubnis zur Ausführung eingeholt.

2.8.1 System und Kernel

Neben der Sicherheitsaspekte des Linux-Kernels kommt in Android eine sichere Interprozesskommunikation zum Einsatz, welche dazu dient, eine stabile und sichere Kommunikation zwischen verschiedenen Prozessen zu gewährleisten. Da dies auf der Betriebssystemebene stattfindet, werden nicht nur Applikationen sondern auch nativer Code isoliert ausgeführt und sind keinem gefährlichem Angriff ausgesetzt.

Die jahrelange Entwicklung und ständige Verbesserungen zeichnen den Linux-Kernel als ein ausgezeichnetes Fundament aus. Seit Jahren ist der Linux-Kernel in unterschiedlichen Umgebungen erfolgreich im Einsatz. Über die Jahre wurde der Linux-Kernel stets durch die Forschung und Angriffe auf diesen von mehreren Tausend Entwicklern verbessert. Viele der Schlüsselfunktionen, welche sich in Android wiederfinden, wurden zum Teil oder komplett übernommen. Dazu gehören unter anderem die Prozessisolation, der Mechanismus für die sichere Interprozesskommunikation und auch das Genehmigungsverfahren zur Rechteverwaltung. Des Weiteren bietet der offene Linux-Kernel die Möglichkeit unnötige Teile zu entfernen und mit eigenen Implementationen zu erweitern.

Mit der Version 4.2 wurde Android zu einem Mehr-Benutzer-System weiterentwickelt. Allerdings ist diese Funktionalität den Smartphones vorenthalten. Lediglich Tablets kommen in den Genuss dieser Technologie. Der Grund hierfür liegt an der Einsatzweise wie das Gerät täglich genutzt wird. Ein Tablet wird in der Regel zu Hause von der ganzen Familie verwendet um kurz im Internet zu surfen oder die Zeit zum Abendessen mit einem kleinen Spiel zu überbrücken. Oft wird allerdings auch auf diesen Geräten mit sensiblen Daten hantiert wie zum Beispiel der Email-Verkehr. Durch die Mehr-Benutzer-Funktionalität kann sich jeder Benutzer individuell ein- und ausloggen um Datenmissbrauch zu verhindern. Jedem Nutzer wird ein eigenes Email-Konto zugewiesen, sodass er beim Aufrufen des Email-Programms nur seinen Posteingang zu sehen bekommt.

Der Grundstein der Mehr-Benutzer-Funktion liegt wieder einmal im Kernel. Dieser unterstützt die Trennung und Isolierung personenbezogener Daten und verhindert somit den Zugriff auf Daten anderer Benutzer desselben Gerätes. Haben mehrere Benutzer parallel eine aktive Sitzung auf einem Android-Tablet laufen, so werden auch benötigte Ressourcen dieser Sitzungen voneinander isoliert. Dies gilt ebenso für Teile des Arbeitsspeichers, als auch für die zugewiesene Rechenleistung des Gerätes auf eine aktive Sitzung. Auch werden Bluetooth und GPS individuell vergeben und können nicht voneinander gestohlen werden.

Sandbox

Jeder Applikation auf einem Android-Gerät wird eine eindeutige UserID (UID) zugewiesen. Isoliert von allen anderen Diensten wird jede Applikation als individueller Benutzer in eigenen Prozessen ausgeführt. Die benötigten Rechte für die Ausführung der Applikation werden dem jeweiligen Benutzer, in welchem der Prozess läuft, gewährt. Viele andere Betriebssysteme, darunter auch Linux, handhaben dies anders. Dort werden mehrere Applikation vom selben Nutzer ausgeführt, welcher die nötigen Rechte besitzt.

Dieses Vorgehen des Android-Betriebssystems bildet die Sandbox, in welcher Applikationen ausgeführt werden. Es wird eine Sicherheitsschicht zwischen System und Prozesse zugunsten der Applikationen gebildet. Applikationen können von Haus aus nicht miteinander kommunizieren und auch der Zugriff auf Funktionen des Betriebssystems sind eingeschränkt und müssen individuell gewährt werden.

Da sich die Implementation der Sandbox im Kernel befindet, sind nicht nur die Applikationen des Betriebssystems davon betroffen. Auch nativer Code, Frameworks und Bibliotheken sind Bausteine, welche sich auf dem Kernel als Grundlage stützen.

2.8.2 Verschlüsselung

Für die Implementierung von kryptografische Verfahren stellt Android mehrere APIs zur Verfügung. Angefangen mit den standardmäßigen und am meisten verwendeten Verfahren AES (Advacned Encryption Standard), RSA (Rivest, Shamir und Adleman), DSA (Digital Signature Algorithm) und SHA (Secure Hash Algorithm) bis hin zu den übergeordneten Protokollen wie SSL (Secure Sockets Layer) und HTTPS (Hypertext Transfer Protocol Secure) werden alle Verschlüsselungsmethoden unterstützt. Eine weitere Sicherheitsmaßnahme, welche mit Android 4.0 hinzugefügt wurde, ist die Möglichkeit über die neu eingeführte Klasse KeyChain private Schlüssel zur Verschlüsselung in den Systemspeicher abzulegen, um den Zugriff für Fremde weiter zu erschweren.

Telefon-Verschlüsselung

Auch Google hat mit Android 3.0 eine eigene Verschlüsselungsmethode in das Betriebssystem eingebunden. Diese Option befindet sich in den Sicherheitseinstellungen unter Verschlüsselung. Die Funktion 'Telefon verschlüsseln' dauert laut Google bis zu einer Stunde und kodiert sämtliche Daten, Konten, Medien und sonstige heruntergeladene Informationen auf dem Gerät [Goo12b].

Gesichert werden die Daten durch ein benutzerdefiniertes Passwort, welches dem des Entsperrbildschirmes entspricht. Die Daten werden einmalig beim Start des Gerätes entschlüsselt und bieten deshalb keinen vollständigen Schutz vor Datendiebstahl. Kommt das Gerät im laufenden Betrieb abhanden, so sind sämtliche Daten bereits entschlüsselt und für jeden

Einsehbar, sofern der Sperrbildschirm umgangen werden kann. Der Sperrbildschirm gilt nicht als sehr sicher und wurde schon des öfteren ausgehebelt.

Die Daten werden im Kernel verschlüsselt durch den Einsatz von dmcrypt AES128 mit CVC und ESSIV:SHA256. Der Schlüssel, welcher für die Verschlüsselung eingesetzt wird, wird vom Passwort des Benutzers abgeleitet. Um systematische Angriffe in Form von brute force oder Ähnliche abzuwehren, wird das Passwort mit einem zufälligem String kombiniert und wiederholt mit SHA1 und dem Standart PBKDF2 Algorithmus gehasht.

2.8.3 Rooting

Ein Benutzerkonto mit Root-Rechten besitzt die höchste Sicherheitseinstufung und ist mit allen möglichen Zugriffsrechten ausgestattet. Ihm erlaubt es das System jegliche Anpassung vorzunehmen und Daten anderer Benutzer und Applikationen einzusehen. Die Berechtigung mit welchen das Root-Benutzerkonto ausgestattet ist sind nicht für den täglichen Gebrauch empfohlen. Besagte Rechte sind nur für spezielle Verwaltungsaufgaben von Nöten und nicht an Benutzer weiterzugeben. Gelangt ein nicht berechtigter Nutzer an nur wenige dieser Berechtigungen, so ist das mit einen erheblichen Sicherheitsrisiko verbunden.

In der Standardausführung von Android besitzt nur der Kernel und einige Systemapplikationen die besagten Rechte zur globalen Verwaltung. Weist ein Benutzer Root-Rechte auf, so ist das Android-Gerät höheren Risiken ausgesetzt. Weitere Applikationen können sich nämlich über den Root-Benutzer sämtliche Rechte aneignen und somit bösartige Software wie Malware installieren oder sogar das System vorübergehend komplett unbenutzbar machen. Außerdem haben Benutzer und Applikation anschließend uneingeschränkte Einsicht auf sämtliche Daten des Systems.

Android bietet die Möglichkeit alternative oder angepasste Betriebssysteme zu installieren. Um dies zu bewerkstelligen muss der Benutzer an Root-Rechte gelangen. Dazu ist es notwendig den Bootloader zu öffnen. Aus Sicherheitsgründen wird bei dieser Prozedur das Gerät auf Werkseinstellungen zurückgesetzt, um Datenmissbrauch vorzubeugen.

Durch die Root-Berechtigungen auf einem Android-Gerät sind selbst durch einen Schlüssel, welcher sich auf im geschützten Bereich des Gerätes befindet, gesicherte Daten nicht sicher. Eine alternative Möglichkeit wäre es den Schlüssel über eine passwortgeschützte Server-Verbindung einzuholen. Allerdings bietet dieser Ansatz auch keine 100-prozentige Sicherheit, da der Schlüssel irgendwann einmal auf dem Gerät landet.

Eine höhere Sicherheit um seine Daten zu schützen bietet ein weiterer Ansatz durch den Einsatz von zusätzlicher Hardware. Eine Möglichkeit ist es einen trusted Storage mittels der Near-Field-Communication (NFC) Technologie einzusetzen, auf welchem bei Bedarf zugegriffen werden kann.

2.8.4 User Security Features

Device Administration

Mit Android 2.2 wurde die Android Device Administration API eingeführt und seither weiterentwickelt und verbessert. Diese erlaubt es administrative Vorkehrungen schon auf Systemebene zu implementieren. Durch die Einführung dieser API ist es sogar möglich, gestohlen oder verloren gegangene Geräte auf Werkseinstellungen zurück zu versetzen und somit alle personenbezogenen Daten ein für alle Mal aus der Ferne zu löschen.

Über die Android Device Administration API ist es zum Beispiel möglich Passwortrichtlinien für bestimmte Applikationen, wie die eingebaute Android Email-App, festzulegen, und so aus der Ferne noch Einfluss auf die Sicherheit seiner Nutzer nehmen [Goo13a].

Credential Storage

Android unterstützt von Haus aus eine Reihe von Zertifikate, welche für die Nutzung sicherer Verbindungen verwendet werden, wie zum Beispiel SSL in einem Browser. Um die Sicherheit zu erhöhen kann jeder Benutzer seit der Veröffentlichung von Android 4.0 bereits im System hinterlegte Zertifikate manuell abwählen. Die Nutzung von weiteren Zertifikaten ist nach Eingabe des Systempassworts und anschließender Installation des Zertifikats möglich.

Virtual Private Network

Die Nutzung eines Vitual Private Network (VPN) ist in Android schon immer möglich gewesen. Dazu steht ein eingebauter VPN-Client bereit, welcher viele Protokolle wie PPTP, L2TP und IPsec unterstützt. Des Weiteren wurde mit Android 4.0 die Klasse android.net.VpnService eingeführt um Drittanbietern die Einbindung eigener VPN-Lösungen zu bieten. Die Funktion ‘always on’, welche kurze Zeit später in Android 4.2 auftauchte, gibt dem Benutzer die Möglichkeit, Applikationen nur über diesen VPN die Verbindung zum Netzwerk herzustellen.

2.8.5 Application Security

Permission Model

Wie bereits weiter oben erwähnt laufen sämtliche Applikationen in einer eigenen Applikation-Sandbox und haben somit nur begrenzen Zugriff auf Systemressourcen. Erweiterter Zugriff auf bestimmte Funktionen und Ressourcen für eine Applikation werden vom System verwaltet. Dabei gelten Regeln und Einschränkungen für verschiedene Ressourcen und Funktionen.

Es existieren allerdings auch Funktionen, auf welche Applikationen keinen Zugriff haben aufgrund nicht vorhandener APIs. Dies kann in erster Linie sicherheitstechnische Hintergründe haben, damit mutmaßliche Manipulation verhindert wird.

Sämtliche Funktionen folgender hardwarespezifischen APIs sind nach der dazugehörigen Erlaubniserteilung für eine Applikation nutzbar:

- Kamera - `android.hardware.Camera - permission.CAMERA`
- Ortung - `android.location - permission.ACCESS_FINE_LOCATION / ACCESS_COARSE_LOCATION`
- Bluetooth - `android.bluetooth - permission.BLUETOOTH / BLUETOOTH_ADMIN`
- Telefon - `android.telephony - permission.READ_PHONE_STATE`
- SMS - `android.telephony.SmsManager - permission.READ_SMS / WRITE_SMS / SEND_SMS`
- Internet - `android.net.Socket - permission.INTERNET`
- eine Liste aller Permissions sind unter [Goo13c] zu finden.

Um die jeweiligen Ressourcen und dessen Funktionen nutzen zu können, muss die dazugehörige Permission in der Manifest-Datei der Applikation deklariert sein. Ist diese nicht in der Manifest-Datei im Permission-Block zu finden, bleibt die Funktion in der jeweiligen Anwendungen ohne Wirkung. Alle angeforderten Permissions werden während der Installation einer Applikation angezeigt und als Ganzes entweder komplett akzeptiert oder durch Abbruch der Installation abgelehnt. Es ist nicht möglich einzelne Permissions abzuwinken und andere nicht.

Ist eine Applikation erst einmal installiert und somit die Zugriffsrechte erteilt, können diese jederzeit von der Applikation genutzt werden, vorausgesetzt die Anwendung bleibt installiert. Um erteilte Berechtigungen aus welchen Gründen auch immer zu widerrufen ist es notwendig die Applikation zu deinstallieren. Sofern die Applikation wieder installiert werden sollte, werden die benötigten Berechtigungen nochmals angefordert.

Es gibt jedoch in einzelnen Fällen wie bei Bluetooth oder GPS die Möglichkeit über die globale Einstellung des Gerätes, den Zugriff auf diese Ressourcen kurzzeitig abzublocken. Dabei kann der Benutzer das GPS- oder Bluetooth-Modul in den Einstellungen komplett deaktivieren. Dies hat zur Folge, dass jegliche Applikationen weder die Position über GPS bestimmen, noch Verbindung über Bluetooth herstellen können. Entwickler können aus ihren Applikationen heraus eine individuelle Anfrage an den Benutzer starten, welche ihn dazu auffordert, die entsprechende Einstellung zu ändern. Diese Funktionalität gilt ebenso für die Datenverbindung über das Mobilfunknetz, das Wireless-LAN Modul und auch die Telefon-Funktion.

Missbrauchsprävention

Das Anzeigen und Einholen von Zugriffsberechtigungen hat in erster Linie das Ziel vor Missbrauch zu schützen. Zum einen kann sich ein jeder Nutzer vor der Installation einer Applikation darüber informieren, welche Berechtigungen für die Funktionalität der Anwendung benötigt wird, und individuell abschätzen, ob der versprochene Funktionsumfang der Applikation mit den angeforderten Berechtigungen vereinbar ist.

Zusätzlich informiert Android bei empfindlichen Berechtigungen über Möglichkeiten zur Veränderung und Manipulation des Betriebssystem, welche die Applikation durch diese erhält.

3 Konzept

Dieses Kapitel befasst sich mit dem Konzept der zu entwickelnden Plattform, die einen Lösungsansatz für das 'Bring-Your-Own-Device'-Problem bieten soll. Im weiteren Verlauf dieser Arbeit wird diese als BYOD-Plattform bezeichnet. Ebenso geht dieses Kapitel auf Projekte ein, welche dieser Arbeit vorangehen und als Grundlage dienen.

3.1 Annahme

In dieser Arbeit wird davon ausgegangen, dass die zu entwickelnde BYOD-Plattform fester Bestandteil des Android-Betriebssystem ist. Die zugrundeliegende Arbeit der Privacy Management Platform, wie sie im Abschnitt 3.3 vorgestellt wird, wurde in ihrer ersten Version als eigenständige Applikation entwickelt. In einer parallel laufenden Arbeit werden Möglichkeit gesucht und abgewägt, diese in das Android-Betriebssystem fest einzubinden. Somit gilt für diese Arbeit, dass es sich hierbei immer noch um eine eigenständige Applikation handelt. Nichtsdestotrotz wird die PMP in naher Zukunft fester Bestandteil des Betriebssystems und hier ebenso behandelt.

3.2 Vorarbeit

Das von Google bei seinem Betriebssystem eingeführte 'Permissions Konzept' sind eine der vielen Sicherheitsvorkehrungen, welche sich in Android finden lassen. Allerdings lässt sich bis heute noch keine 100 prozentige Sicherheit gewährleisten. Auch aufgrund dieser Lücken arbeiten immer mehr Entwickler an unabhängigen Lösungen und versuchen dem Benutzer neuartige und bessere Ansätze aufzuzeigen.

Das angesprochene 'Permissions Konzept' zeigt bei der Installation einer Applikation, die für diese Anwendung benötigten Zugriffsrechte. Diese müssen vom Entwickler vorher in der Manifest-Datei der Anwendung explizit angegeben werden. Sind 'Permissions' in der Manifest-Datei nicht angegeben, so steht die entsprechende Funktionalität auch nicht zur Verfügung, auch wenn sie möglicherweise vollständig Implementiert wurde. Dies soll die Nutzung von mächtigen Funktionen verhindern, welche die Einwilligung des Nutzers benötigen.

Allerdings hat auch dieses Konzept einen kleinen Haken. Ist man mit einer der vielen 'Permissions' nicht einverstanden so existiert für den Nutzer keinerlei Möglichkeit diese

anzupassen oder zu deaktivieren. Entweder man installiert die Applikation und gibt sich mit der Berechtigungsvergabe zufrieden, oder man verzichtet gänzlich auf diese Anwendung und bricht den Installationsvorgang ab. Dabei ist es egal ob die fragwürdige Berechtigung für die benötigte Funktionalität der Applikation erforderlich ist oder nicht.

Eine mächtige Berechtigung im Android-Betriebssystem ist zum Beispiel der Zugriff auf das Telefonbuch. Deklariert ein Entwickler aus einem unbekannten Grund die Berechtigung zum Lesen des Telefonbuches auch ohne irgendeine Funktion des Telefonbuchs implementiert zu haben, so wird dies bei der Installation angezeigt und in den meisten Augen als gefährlich eingestuft. Dagegen ist bei einer Messaging-Applikation das Lesen des Kontaktbuches unumgänglich und wird dementsprechend von den meisten Nutzern für nötig befunden und akzeptiert.

Die Berechtigungen einer Applikation kann im Android Betriebssystem auch nach der Installation noch im Einstellungsmenü unter Apps eingesehen werden.

3.3 Privacy Management Platform

Das oben angesprochene Problem des 'Permission Konzept' wurde von der Universität Stuttgart in einem Studienprojekt angenommen und weitreichend behandelt. Es galt eine Plattform zu entwickeln, die das Prinzip der Permissions, um die Möglichkeit zum abwählen einzelner Funktionen, erweitert. An diesem Projekt nahmen 10 Studenten im Umfang von einem Jahr teil, welches im April 2011 startete. Beendet wurde das Studienprojekt dann im April 2012 mit einem Zeitaufwand von etwa 4500 Stunden.

Das Ziel wurde erfolgreich erreicht. Es entstand die Privacy Management Platform, kurz PMP, die das 'Permission Konzept' ersetzt und dem Benutzer erlaubt, Berechtigungen nach eigenem Ermessen zu gewähren und zu verbieten. Hierdurch soll die Sicherheit erhöht und die Privatsphäre des Nutzers geschützt werden.

Zum weiteren Funktionsumfang der PMP gehört:

- Automatische Anpassung der Berechtigungen an Ort und Zeit.
- Import- und Exportfunktionen für voreingestellte Berechtigungsgruppen.

Im Rahmen des Studienprojektes wurde eine Android Applikation entwickelt, die den Zugriff auf Ressourcen steuern und kontrollieren kann. Ressourcen bestehen aus Funktionen und Berechtigungen, die in eine Ressourcengruppe zusammengefasst werden. Verwaltet werden diese über sogenannte 'Presets' innerhalb der Applikation.

Abbildung 3.1 zeigt den Ablauf der Funktionen der PMP. Während der Installation einer Applikation wird der Registrierungsprozess der PMP gestartet. Dieser hinterlegt einen Eintrag in der PMP und informiert diese über die Präsenz dieser Applikation. Sollte die PMP nicht installiert sein, so lässt sich die Applikation auch nicht ausführen.

Nach dem Registrierungsprozess wird die Applikation unter 'Apps' gelistet und die dazu benötigten Ressourcen vermerkt. Sind benötigte Ressourcengruppen nicht vorhanden, so

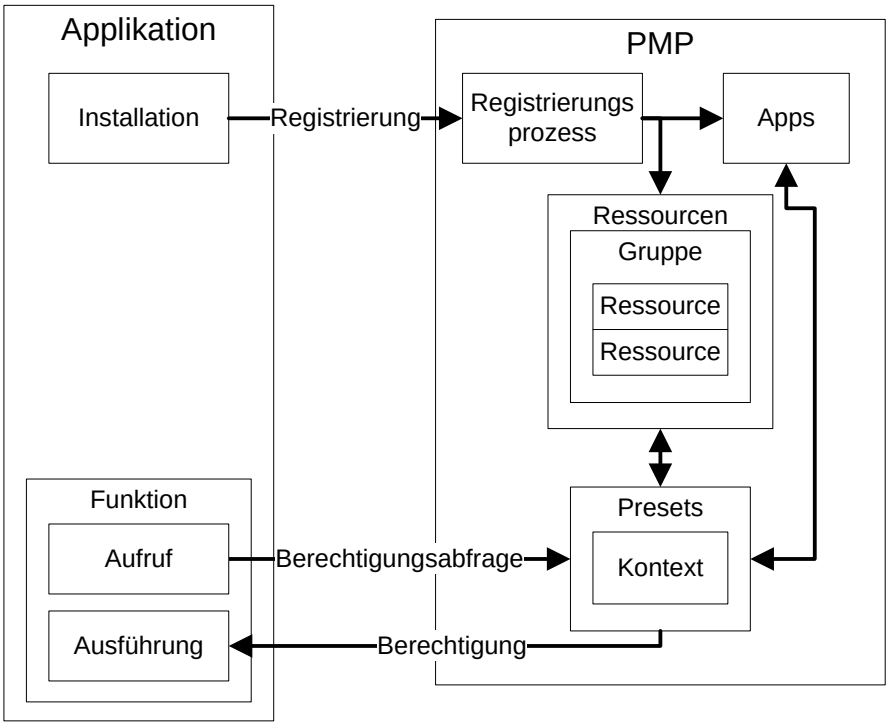

Abbildung 3.1: Kommunikation zwischen Applikation und PMP.

können sie über den eingestellten Server im Programm, auch noch während des Registrierungsprozess, nachinstalliert werden.

In einer weiteren Arbeit, welche ebenso die PMP als Grundlage nutzt, wird versucht diese in das Android Betriebssystem fest einzubinden. In diesem Fall wird die PMP als vorinstallierter Bestandteil des Betriebssystems angesehen. Dazu mehr in Abschnitt 3.1.

3.3.1 Berechtigungsrichtlinienmodell

Richtlinien und Berechtigungen werden in der PMP von vier wesentlichen Komponenten gesteuert [Sta13b]. In Abbildung 3.2 ist das Zusammenspiel zwischen einer Policy Rule und den vier Komponenten grafisch dargestellt.

Einzelne Funktionen umfangreicher Applikationen können in sogenannte 'Service Feature' gepackt werden. Diesem 'Service Features' kann anschließend vom Benutzer individuell eine Berechtigung zur Ausführung erteilt werden oder nicht. Die Auswahl ist jederzeit anpassbar. Die Berechtigung wird ausschließlich der Funktion erteilt und nicht der Applikation.

Ressourcen werden für besondere Funktionen und Daten des Systems genutzt. Um diese einer Applikation zur Verfügung zu stellen, ist es notwendig Ressourcengruppen für die erforderlichen Funktionen anzubieten. Die PMP enthält bereits viele vordefinierte Ressourcen,

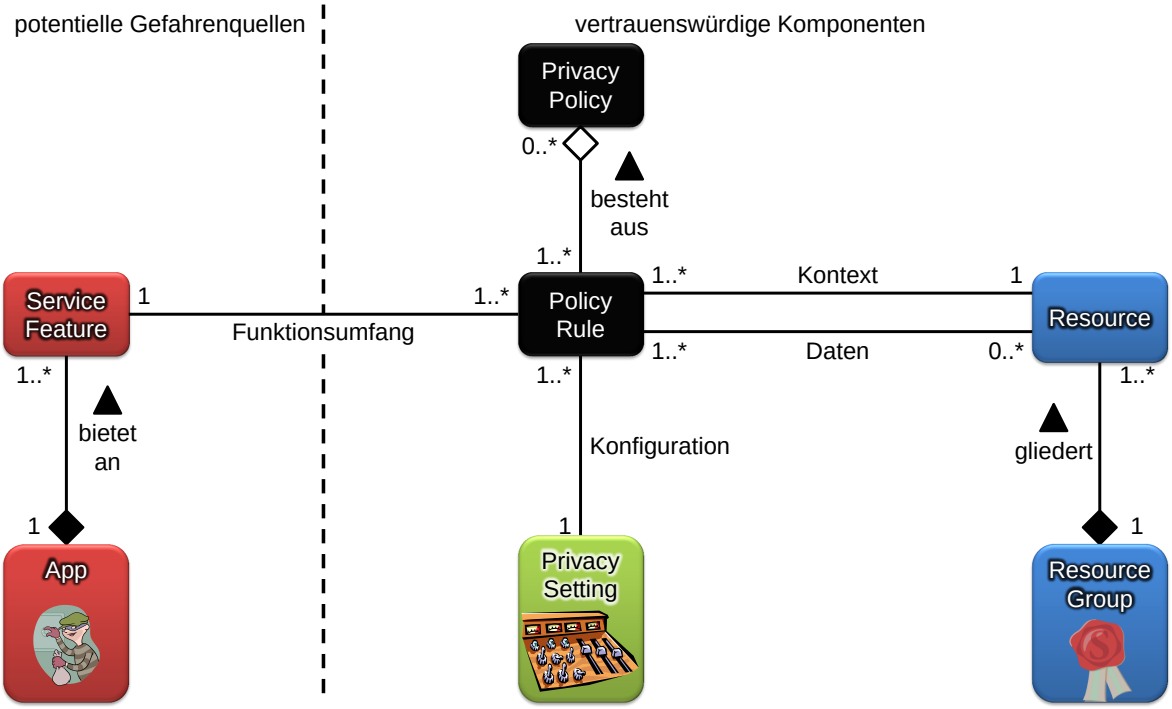

Abbildung 3.2: PMP-Modell für Berechtigungen und Richtlinien [Sta13b].

welche bei Bedarf erweitert oder weiterentwickelt werden können. Fehlt eine Ressource bei Aufruf einer Funktion, so kann diese, falls vorhanden, von der PMP nachinstalliert werden.

Die 'Privacy Setting' gibt an, wie die eingestellte Ressourcen verwendet werden. Dabei kann der Nutzer zu Testzwecken randomisierte Koordinaten anstelle der echten Koordinaten über das GPS-Modul zurückgeben lassen. Auch können voreingestellte Koordinaten festgelegt werden. Verschiedene Ressourcen können zusätzlich besondere Funktionalitäten bieten. Für das GPS-Modul kann die Genauigkeit auf eine feste Anzahl von Metern eingestellt werden.

Durch die Datenschutzeinstellungen, die der Nutzer vornimmt und erteilt, können bestimmte Service Features auf zugeteilte Ressourcen zugreifen. Dies entspricht einer Datenschutzregel in der PMP. Des Weiteren können diese, wie schon in Abschnitt 3.3 erwähnt, an bestimmte Orte oder Zeiten gekoppelt werden [Sta13b].

3.3.2 Privacy Policy

In Abbildung 3.3 ist das Zusammenspiel von 'Privacy Setting', 'Service Feature' und 'Ressource' grafisch dargestellt. Die 'Privacy Policy', in welchem diese Wechselbeziehung zusammen läuft, hält alle wichtigen Informationen für die Autorisierung von Zugriffen auf geschützte Daten.

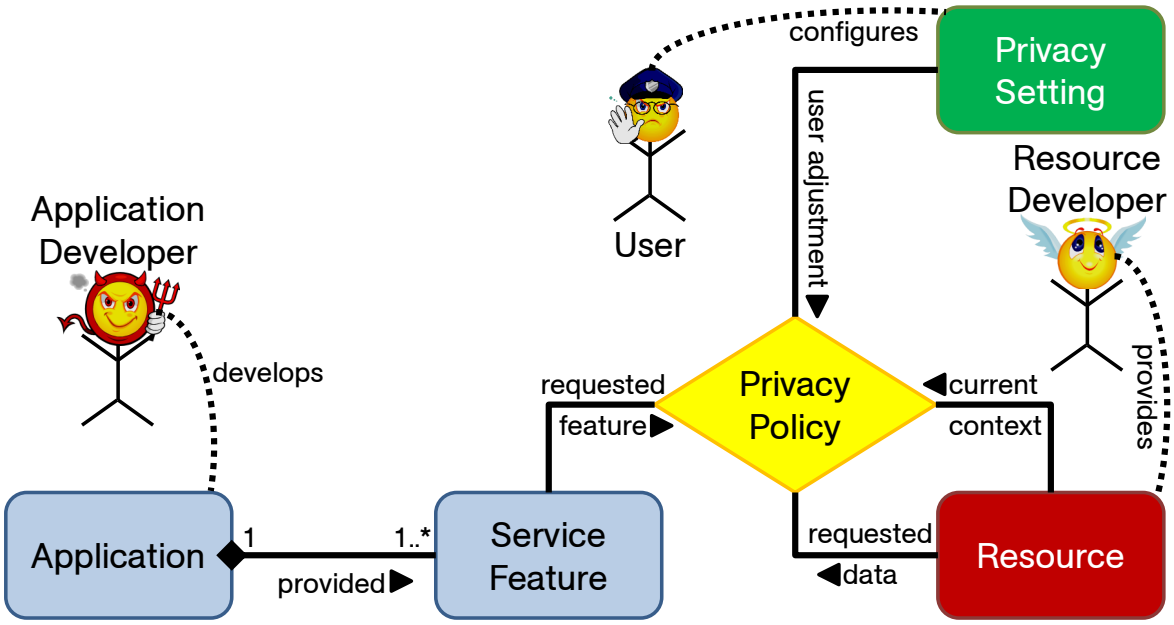

Abbildung 3.3: Wirkung einzelner Komponenten auf die Datenschutzrichtlinien [SM13].

Es wird angenommen, dass nur Ressourcen von vertrauenswürdigen Entwicklern [SM13] in die PMP aufgenommen werden. So soll in erster Linie verhindert werden, dass bösartige Funktionen auf dem Gerät ausgeführt werden können. Die Funktion der Ressourcen, ist die Weitergabe der angeforderten Daten.

Auf der anderen Seite stehen die Entwickler der Applikationen, welche nicht kontrolliert und überwacht werden können. Diese könnten die Absicht verfolgen Malware oder Spyware auf einem Gerät zu platzieren. Tiefgreifende Funktionen, die allerdings für einen solchen Eingriff nötig sind, müssen durch das PMP-Modell autorisiert werden. Neben der entsprechenden Ressource ist auch die Autorisierung durch den Benutzer notwendig.

Letzterer ist schließlich das Sicherheitspersonal der PMP. Dem Nutzer ist es erlaubt jedes angebotene Service Feature einer Applikation individuell abzuwählen und so vor unbefugtem Zugriff zu schützen. Ist der Benutzer auf eine Applikation angewiesen, aber hat bei einigen Service Features bedenken, so kann er diese abwählen [Sta13a].

3.4 BYOD-Plattform

3.4.1 Kommunikation

Abbildung 3.4 zeigt den Kommunikationsablauf zwischen einer Applikation und der BYOD-Plattform. Im Vergleich zur Abbildung 3.1 wurde diese um das Konfigurations-Objekt

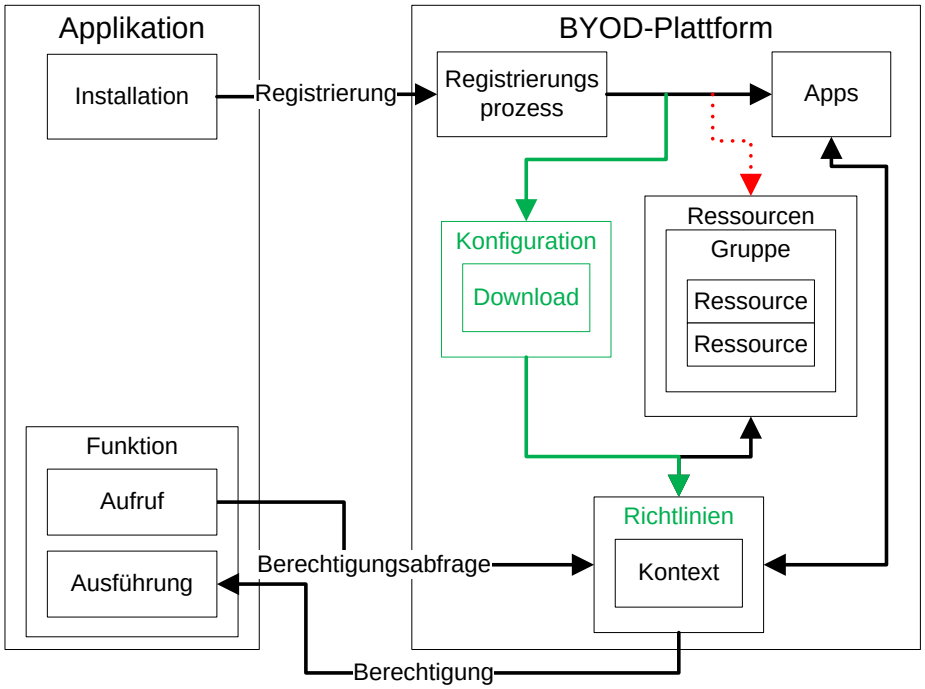

Abbildung 3.4: Kommunikation zwischen Applikation und BYOD-Plattform.

erweitert. Des Weiteren wurden die Presets an sich übernommen, jedoch in ihrer Funktion angepasst und umbenannt.

Durch die Anpassung der Presets werden diese nun nicht mehr vom Benutzer bei der Installation gesetzt, sondern primär über die Konfigurationsdatei verwaltet. Dies soll verhindern, dass geschützte Applikationen und Ressourcen nicht vom Benutzer selbst bearbeitet werden können.

Weitere Richtlinien lassen sich weiterhin vom Benutzer über das Menü erstellen und bearbeiten. Ausgeschlossen von der Modifikation sind allerdings firmeneigene Applikationen, welche dazu dienen, die Sicherheit des Unternehmens zu gewähren.

3.4.2 Applikation und Funktionen

Damit die Konfiguration und Verwaltung von Applikationen über die BYOD-Plattform verlaufen soll, müssen diese dementsprechend angepasst sein. Aus diesem Grund wird eine Programmierschnittstelle (API) zur Verfügung gestellt werden, mit welcher Applikationen und dessen Funktionen der Plattform angepasst werden können. Bei der Programmierung einer Applikation muss die bereitstehende API mit eingebunden werden.

Greifen Funktionen einer Applikation auf gerätespezifische Funktionalitäten zu, so müssen diese anschließend nicht wie gewohnt über das Hardware-Paket von Android eingefordert

werden, sondern mittels der API der Plattform eingebunden. Auch können selbst definierte Funktionen über die Plattform eingebunden werden und somit in die Richtlinien aufgenommen werden. Dazu muss allerdings für die eingeforderten Funktionen eine dementsprechende Ressource zur Verfügung stehen. Ist dies nicht der Fall kann allerdings auch gleichzeitig mit der Applikation auch eine entsprechende Ressourcengruppe erstellt werden, welche die benötigten Ressourcen beinhaltet. Hierzu steht ebenso wie für die Anpassung der Applikation die selbe API zur Verfügung.

3.4.3 Registrierungsprozess

Ähnlich wie bei der PMP sollen sowohl androidspezifische als auch firmeneigene Applikationen zur Nutzung bei der entsprechenden Plattform registriert werden. Ohne Registrierung ist die Verwendung nicht möglich. Allerdings soll der Registrierungsprozess keine Hürde für Entwickler von Applikationen sein, sondern lediglich auf die Zuständigkeit zur Plattform hindeuten.

Während bei der PMP im Zuge des Registrierungsprozesses die Zugriffsberechtigungen der jeweiligen Applikation vom Benutzer konfiguriert werden, soll die BYOD-Plattform die entsprechenden Richtlinien aus der Konfigurationsdatei lesen und anwenden. Da die Plattform Richtlinien von Applikationen, die nicht installiert sind, nicht verarbeiten kann, werden diese beim Einlesen der Konfigurationsdatei schlicht übersprungen. Deshalb soll gegen Ende des Registrierungsprozesses entweder die Konfigurationsdatei neu vom Konfigurationsserver des Unternehmens geladen oder eine zwischengespeicherte Version neu eingelesen werden.

3.4.4 Konfiguration

Wohingegen die Konfiguration der PMP logischerweise dem Benutzer überlassen wird, muss im Fall der BYOD-Plattform das Unternehmen aktiv werden. Das Ziel der PMP ist es, die Sicherheit und Privatsphäre des Nutzers zu gewährleisten. Dazu konfiguriert der Benutzer selbstständig die Zugriffsrechte und Ressourcenzugriffe sämtlicher Applikationen. Auch kann dieser alle vorgenommenen Einstellungen jederzeit an seine Bedürfnisse anpassen.

Im Gegensatz dazu muss in der BYOD-Plattform auf die Wünsche und Bedürfnisse zweier Parteien eingehen und diese kombinieren. Auf der einen Seite steht das Unternehmen, welches wichtige Daten vor Missbrauch schützen will. Auf der anderen Seite möchte der Besitzer und tägliche Begleiter des Gerätes sowohl Unternehmensdaten für seine Arbeit verwenden, als auch private Tätigkeiten über sein Smartphone abwickeln. Aus diesem Grund gilt es eine Konfigurationsmöglichkeit zu finden, die es erlaubt beide Seiten den entsprechenden Spielraum zu überlassen.

In dieser Arbeit soll eine Lösung implementiert werden, die anhand ihrer Konzeption der einen Partei eine höhere Priorität gewährt, um die Wünsche dieser Seite besser umsetzen zu

können. Dieser Schritt ist nötig, denn schließlich wird das private, mobile Gerät im Unternehmen eingesetzt und mit sensiblen Daten in Kontakt kommen. Somit soll in erster Linie das Unternehmen ihre eingesetzte Software eigenständig auf dem Gerät verwalten können. Des Weiteren soll der Benutzer das Recht erhalten, alle Standardfunktionen des Smartphones selbstständig zu konfigurieren. Optional soll das Zeitintervall für die Konfiguration des Benutzers vom Unternehmen eingeschränkt werden können, um vorgeschriebene Richtlinien während der Arbeitszeiten geltend zu machen.

Um dieses Vorgehen zu realisieren soll eine Konfigurationsdatei vordefinierte Richtlinien in die Plattform einspeisen. Um neue und verbesserte Richtlinien mit der Zeit einpflegen zu können soll durch den Download über einen Firmenserver die Konfiguration update-fähig sein.

4 Implementierung und Anwendung

4.1 Konfiguration

4.1.1 Remote-Konfiguration

Die Plattform wird zentral über eine Konfigurationsdatei vom jeweiligen Unternehmen verwaltet und wie im folgenden Abschnitt erklärt eingepflegt und verwaltet.

Download

Der Download der Konfigurationsdatei findet von dem Firmeneigenen Server oder einer anderen beliebigen Serverquelle statt. Zum Einsatz kommt das im Internet standardmäßige Hypertext-Übertragungsprotokoll (HTTP). Unter den erweiterten Einstellungen kann nach PIN-Eingabe die Adresse zur Konfigurationsdatei geändert oder hinzugefügt werden. Das Speichern der Konfigurationsdatei ist dabei nötig, um bei inaktiver Verbindung zum Server die Konfiguration nach dem zuletzt bekannten Zustand wiederherzustellen. Hierzu wird die Datei im Speicher der Applikation abgelegt, zu welchem im Normalfall nur das entsprechende Programm Zugriff hat. Es gibt allerdings Fälle in denen sich Dritte Zugriff zu dieser Konfigurationsdatei verschaffen können, und die Plattform nach eigenen Wünschen anpassen können. Weiteres dazu in Abschnitt 4.5.2.

Inhalt

Listing 4.1 zeigt den Aufbau einer Konfigurationsdatei am Beispiel der nötigen Ressourcenfreigaben für die Anwendung, welche Geschäftskundendaten anzeigt und kontaktieren lässt.

- `<config>` - Definiert das Root-Element der Konfigurationsdatei, welches das einzig existierende Root-Element sein darf, und vorkommen muss.
- `<compname>` - Hier wird ein beliebiger Name gewählt, meist des agierenden Unternehmens, welches an verschiedenen Stellen der Plattform auftaucht.
- `<timeoff>` - Der 'Time-off' beschreibt die Zeit, in welcher Benutzer selber eine eingeschränkte Kontrolle über die Platform haben, meistens außerhalb der Arbeitszeiten. Dazu mehr unter 4.1.2.

Abbildung 4.1: Die erweiterten Einstellungen sind mittels einer PIN-Abfrage geschützt.

- `<guidelines>` - Unter `guidelines` werden die firmeneigenen Richtlinien deklariert, jeweils untergeordnet in einem eigenen `<guideline>`-Blockelement. Diese sind nach dem Einlesen der Konfigurationsdatei unter dem Menüpunkt 'Richtlinien' wieder zu finden. Der Unterschied zu den benutzerdefinierten Richtlinien wird in Abschnitt 4.1.2 erläutert.
- `<guideline>` - Jede Richtlinie welche definiert werden soll, wird in einem eigenem Element deklariert. Zu jeder Richtlinie wird ein Name <name> und eine Beschreibung <description> angelegt, welche die jeweilige Richtlinie von anderen Unterscheidet.
- `<apps>` - Unter dem Element `apps` werden alle Apps aufgeführt, welche unter den Bedingungen der jeweiligen Richtlinie ausgeführt werden können. Dazu wird der Paketname der jeweiligen Anwendung in ein Element mit dem Namen <app> deklariert.

- `<privacySettings>` - Als letzten Punkt werden die Privacy Settings aufgeführt, welche ausgewählte Ressourcen unter bestimmten Umständen für die Richtlinien und deren Applikationen freigeben.
- Die Elemente `<name>` und `<value>` beinhalten den genauen Namen der Ressource, wie er auch in der dazugehörigen Ressourcengruppe definiert ist, und den Wert, welcher angenommen werden soll.
- `<resourceGroup>` - Hier wird der Paketname der Ressourcengruppe angegeben, damit die dazugehörige Ressource gefunden werden kann.
- `<contextAnnotation>` - Unter welchen Umständen die Richtlinie für diese Ressource greift, wird in diesem Abschnitt definiert.
- `<context>` - Der Kontext kann entweder zeitabhängig oder ortsabhängig sein. Hierfür wird der Wert dementsprechend auf '0' oder '1' gesetzt.
- `<contextCondition>` - Bei einem Zeitkontext baut sich der Wert des Elements aus Startzeit, Endzeit, dem Wiederholungszyklus und weiteren Werten für die Tage auf. Der Wiederholungszyklus kann dabei den Wert für Woche (W), Monat (M) oder Jahr (Y) einnehmen. Die dazugehörigen Zahlen definieren die Wochentage: 2 für Montag, 3 für Dienstag, usw. Für die Wiederholung am 26. Juni eines jeden Jahres nimmt die Variable den Wert 'Y5,26' ein.
- `<overrideValue>` - Dieses Element kann den Wert 'true' oder 'false' annehmen und beschreibt die Gültigkeit des Kontextes.

Installation

Die Datei zur Konfiguration der Plattform wird wie schon in 4.1.1 über eine aktive Internetverbindung von einem Remote-Server geladen. Alternativ kann über einen Link aus dem Intranet, sofern sich der Mitarbeiter in einem Firmennetzwerk befindet, die Konfigurationsdatei geladen werden. Da nicht jederzeit vom Benutzer oder vom Netzbetreiber eine vorhandene Verbindung zum Server gewährleistet werden kann, entscheidet die Plattform in Abhängigkeit der Konnektivität den Ablauf der zu ladenden Konfiguration.

Um die Aktualität der Konfiguration zu gewährleisten, wird zuallererst versucht die aktuellste Konfigurationsdatei vom angegebenen Server abzurufen. Dazu wird mittels des `Android.net.ConnectivityManager` überprüft ob eine aktive Internetverbindung besteht. Ist dies der Fall, so wird anschließend überprüft ob die Adresse, welche im Programm hinterlegt ist, erreichbar ist. Sollte die Adresse Fehler aufweist oder zu dieser Zeit nicht erreichbar ist, wird überprüft ob eine ältere Konfigurationsdatei bereits existiert. Abhängig vom auftretenden Fall wird wie Abbildung 4.3 zeigt, der Prozess für den entsprechenden Abschnitt eingeleitet. Im Falle von 'Nichts tun' wird die Plattform im vorherigen Zustand samt Einstellungen belassen. Der abgebildete Ablauf wird in der Klasse `de.unistuttgart.ipvs.pmp.service.config.ConfigService` als Service (`android.app.Service`) alle 24 Stunden ausgeführt. Sollte die BYOD-Anwendung vor dem

Listing 4.1 Inhalt der Konfigurationsdatei.

```
<?xml version="1.0" encoding="utf-8"?>
<config>
  <compname>B.Y.O.D. GmbH</compname>
  <timeoff>17:00:00-07:00:00</timeoff>
  <guidelines>
    <guideline>
      <name>Test</name>
      <description>Test Description</description>
      <apps>
        <app>de.unistuttgart.ipvs.pmp.apps.businessclients</app>
      </apps>
      <privacySettings>
        <privacySetting>
          <name>callPhone</name>
          <value>true</value>
          <resourceGroup>de.unis...groups.businessclients</resourceGroup>
          <contextAnnotations>
            <contextAnnotation>
              <context>0</context>
              <contextCondition>07:00:00-17:00:00-W2,3,4,5,6,</contextCondition>
              <overrideValue>true</overrideValue>
            </contextAnnotation>
          </contextAnnotations>
        </privacySetting>
        <privacySetting>
          <name>sendMail</name>
          <value>true</value>
          <resourceGroup>de.unis...groups.businessclients</resourceGroup>
          <contextAnnotations>
            <contextAnnotation>
            <context>0</context>
              <contextCondition>06:00:00-20:00:00-W2,3,4,5,6,</contextCondition>
              <overrideValue>true</overrideValue>
            </contextAnnotation>
          </contextAnnotations>
        </privacySetting>
      </privacySettings>
    </guideline>
  </guidelines>
</config>
```

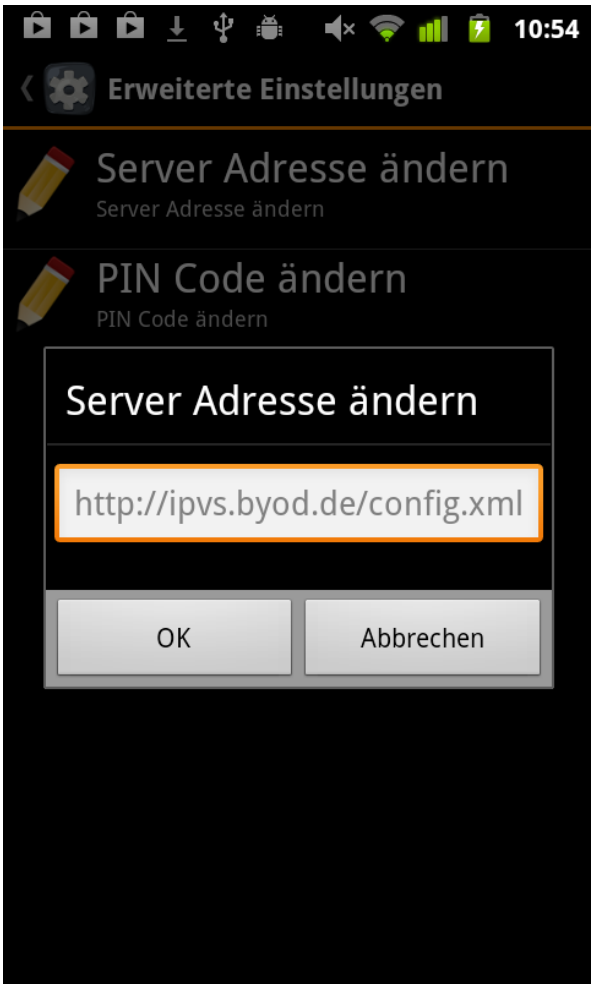

Abbildung 4.2: Konfigurationsmaske zur Anpassung der Serveradresse für den Download der Konfigurationsdatei.

Ablauf der 24 Stunden wiederholt gestartet werden, so beginnt der Aktualisierungsvorgang sofort und das Intervall wird auf volle 24 Stunden zurückgesetzt.

Auch nach Installation und Registrierung einer neuen Anwendung bei der BYOD-Plattform, wird der Service gestartet, um jene Applikation zur vordefinierten Richtlinie hinzu zu fügen. Denn sollte eine Applikation, welche in der Konfigurationsdatei definiert und in einer Richtlinie enthalten ist, im Moment der Einbindung in die Plattform nicht installiert sein, so wird diese übersprungen und nicht hinzugefügt.

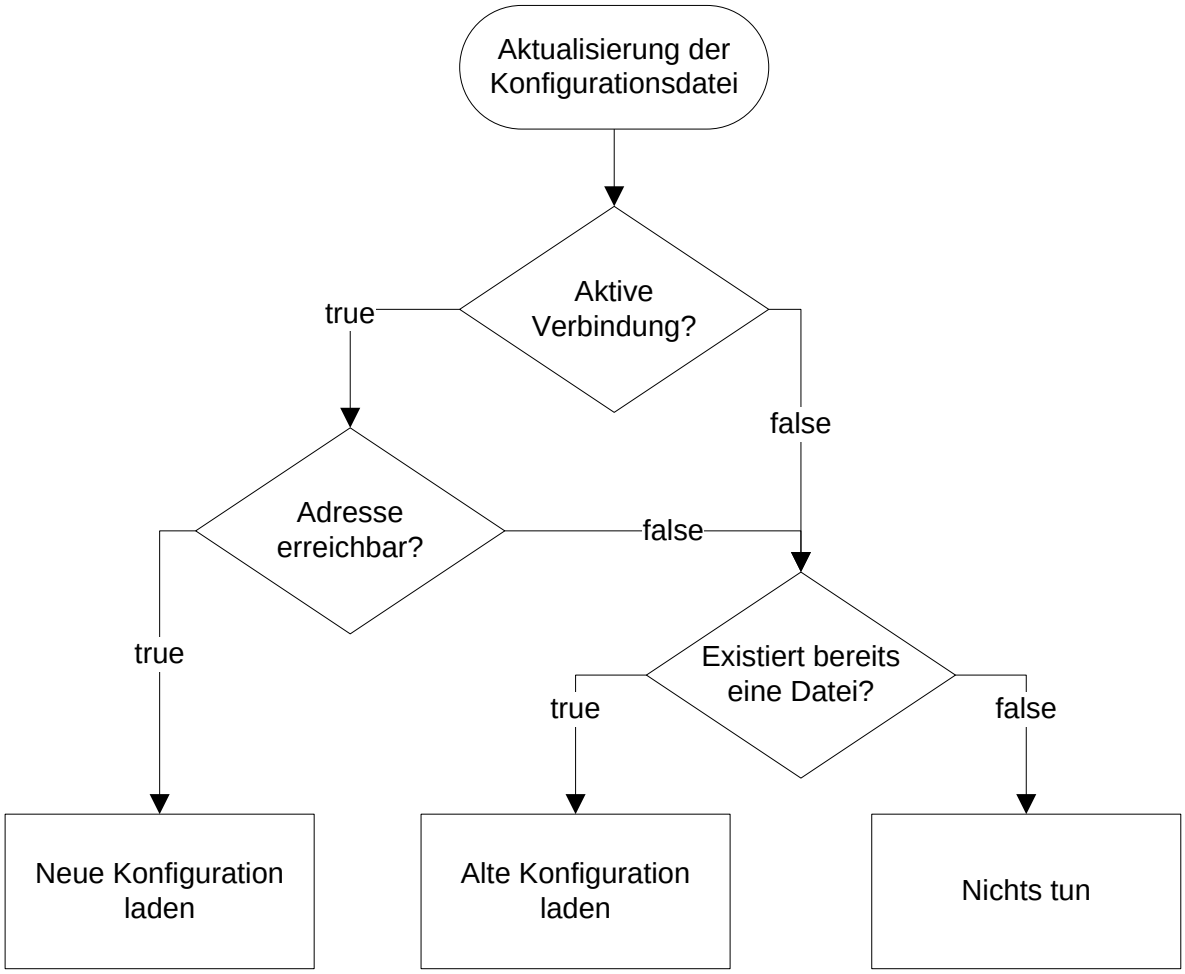

Abbildung 4.3: Ablaufschema des Entscheidungsprozesses für die Aktualisierung der Konfigurationsdatei.

4.1.2 Benutzerdefinierte Konfiguration

Während die Konfigurationsdatei die Richtlinien während der Arbeitszeiten regelt, muss vom Benutzer die Einstellung für die Zeit nach der Arbeit selbst konfiguriert werden.

Richtlinien

Unter dem Menüpunkt 'Richtlinien' stehen dem Benutzer folgende Konfigurationsmöglichkeiten zur Verfügung:

Hinzufügen und Bearbeiten einer Richtlinie Damit ein Nutzer sein Gerät auch nach Feierabend nach seinen Vorstellungen nutzen kann, ist es notwendig benutzerdefinierte Richtlinien zu erstellen und nach eigenen Wünschen anzupassen. Dazu gilt es erst einen beliebigen Namen und eine passende Beschreibung zu finden.

Zuweisen von Applikationen Jede Richtlinie bildet eine Gruppe von gewährten Funktionen für alle in dieser Richtlinie definierten Applikationen. Hierzu muss für jede Richtlinie individuell die gewünschten Apps hinzugefügt werden.

Hinzufügen von Ressourcen aus Ressourcengruppen Ressourcengruppen enthalten meist mehrere Funktionen bzw. Ressourcen, welche für bestimmte Abläufe einer Applikation nötig sind. Wird zum Beispiel aus einer Applikation heraus ein Anruf getätigt, so muss dieser Anwendung die Möglichkeit gewährt werden eine Nummer zu wählen.

Kontextbezogene Einschränkungen zu der Verfügbarkeit der ausgewählten Privacy Setting Zu jeder gewährten Ressource muss ein Zeitraum gewählt werden, in welchem die entsprechende Funktion ausgeführt werden darf. Ausgeschlossen von diesen Zeiten sind die vom Unternehmen in der Konfigurationsdatei gewählten Arbeitszeiten. Nur nach bzw. vor diesen Zeiten sind benutzerdefinierte Richtlinien möglich. Wird kein Zeitraum gewählt so werden allen Anwendungen dieser Richtlinie automatisch die Zeit außerhalb der Arbeitszeit hinzugefügt. In Abbildung 4.4 sind beispielhaft zwei Zeitkontexte für die Ressourcen ‘eMail’ und ‘Telefon’ voreingestellt.

Einsicht auf vordefinierte Richtlinien Um seine eigene Privatsphäre zu schützen, ist die Einsicht in die Richtlinien des Unternehmens gestattet. Eingriffe lassen sich allerdings nur über den Kontakt zum internen Support diskutieren.

Einschränkungen

Folgende Funktionen sind aus Sicherheitsgründen nicht verfügbar:

Bearbeitung und Löschung von vordefinierten Richtlinien Aus Schutz vor widerrechtlichem Zugriff auf Firmendaten ist die Bearbeitung von vordefinierten Richtlinien nicht gestattet. Es könnten Applikationen Dritter hinzugefügt oder die Zugriffzeiten verändert werden um an sensible Daten des Unternehmens zu gelangen.

Keine Nutzung von Ressourcen aus firmeneigenen Ressourcengruppen Ressourcengruppen erlauben Zugriff auf geschützte Funktionen. Aus diesem Grund ist das verwenden von Ressourcen aus Ressourcengruppen, welche von Unternehmen für den Einsatz in Firmenanwendungen vorgesehen sind, nicht gestattet und in der Plattform unterbunden.

Abbildung 4.4: Einschränkung in der Verfügbarkeit der Privacy Settings durch den Faktor Zeit.

4.1.3 Erweiterte Einstellungen

Unter dem Menüpunkt ‘Einstellungen’ kann im Abschnitt der erweiterten Einstellungen der PIN-Code für den Zugang geändert werden. Standardmäßig ist der Code auf ‘1111’ gesetzt. Abbildung 4.1 zeigt die Login-Maske für das Untermenü.

Des Weiteren kann und muss in diesen Untermenü die Serveradresse angegeben werden, von welcher die Konfigurationsdatei geladen wird. Abbildung 4.2 zeigt das Eingabefeld für die Serveradresse. Im Auslieferungszustand beinhaltet die Plattform keine Adresse und somit keine Konfigurationsdatei. Das heißt es sind keine vordefinierten Richtlinien und auch keine Konfigurationsdatei vorhanden. Sobald eine Adresse hinzugefügt wird, wird diese bei bestehender Verbindung zum Server abgerufen und installiert.

4.2 Ressourcen

Ressourcen sind der wichtigste Bestandteil der Plattform. Ohne sie können Funktionen einzelner Applikationen nicht ausgeführt werden und schränken ihre Funktionalität somit stark ein. In der BYOD-Plattform können alle installierten Ressourcengruppen eingesehen werden wie in Abbildung 4.5 zu sehen ist. In Untermenüs finden sich, wie in in Abbildung 4.6 zu sehen ist, Beschreibungen zum Funktionsumfang einzelnder Ressourcen und Service Features.

In einer Applikation werden Service Features definiert, welche Funktionen bestimmter Ressourcen für diese Anwendung festlegen. Dabei werden diejenigen Ressourcen ausgewählt, welche für die Ausführung der Dienste in der jeweiligen Applikation benötigt werden. In der Regel besitzt eine Applikation mehrere Service Features.

Jedes Service Feature legt fest, welche Ressourcen und dessen Funktionen benötigt werden. Umso mehr Service Features der Applikation gewährt werden, desto größer ist der Umfang der Funktionalität.

Sind zwei oder mehr Ressourcen inhaltlich oder aufgrund ihrer Herkunft miteinander verwandt oder verknüpft, können diese vom Entwickler zu einer Ressourcengruppe zusammengefasst werden. Durch den Verbund in einer Gruppe können diese einfacher und als Ganzes ausgeliefert und auf einem Gerät installiert werden.

4.2.1 Erstellen der Ressourcen

Zur Erstellung einer Ressource muss die Basisklasse `de.unistuttgart.ipvs.pmp.resource.Resource` erweutert werden, welche im Paket der BYOD-Plattform wieder zu finden ist. Anschließend müssen drei Methoden zur Fallunterscheidung implementiert werden. Im Einzelnen sind dies `getAndroidInterface()`, `getMockedAndroidInterface()` und `getCloakedAndroidInterface()`. Jede einzelne Methoden ruft entsprechend ihrer Bestimmung eine andere Implementierung der Ressource auf. Näheres dazu weiter unten. Listing 4.2 zeigt die Klasse 'MailResource', welche von der Beispielanwendung verwendet wird, um E-Mails aus der Applikation heraus zu versenden.

Zusätzlich zur Klasse muss im Paket der Ressourcengruppe ein AIDL (Android Interface Definition Language) erstellt werden, welches die Funktionen der Ressource beinhaltet. Diese werden durch die Einbindung des Interfaces für entsprechende Applikationen sichtbar. Listing 4.3 zeigt den Aufbau des Interfaces der Ressource Mail, welche zum Versand einer E-Mail aus der Applikation heraus benötigt wird.

Das Android-SDK erstellt aus der `AIDL`-Datei automatisch ein `Stub`, welches bis zu dreimal implementiert werden kann. Jede einzelne Implementation gestattet der Plattform unterschiedliche Werte abhängig vom eingestellten Modus zurück zu liefern. In der ersten Ausführung werden die normalen Werte zurückgegeben. Im Falle des GPS-Moduls würden die exakten Koordinaten des Gerätes bestimmt und ausgeliefert werden. Die zweite Implementierung kann genutzt werden, um abgespeicherte Werte zurück zu geben, ohne dass

Abbildung 4.5: Ressourcenmenü zeigt alle installierten und verfügbaren Ressourcengruppen an.

diese der Wahrheit entsprechen. Dies bietet zusätzliche Möglichkeiten für das Debuggen. Als letzte Implementierung ist es gestattet zufällige Werte zu berechnen und zurück zu geben.

In jeder der drei Ausführungen ist es anschließend zwingend erforderlich die in der AIDL-Datei deklarierten Funktionen einzubinden. Listing 4.4 zeigt die Implementierung der korrekten Ausführung der Ressource. Zu Beginn der Funktion `sendMail()` wird mittels der Methode `verifyAccessAllowed(..)` die Berechtigung der Applikation abgefragt.

Listing 4.2 Erweiterung der Basisklasse 'Resource' zur Erstellung einer Ressource.

```
public class MailResource extends Resource {
	private BusinessClientsResourceGroup busiClRG;

	public MailResource(BusinessClientsResourceGroup busiClRG) {
		this.busiClRG = busiClRG;
	}

	public IBinder getAndroidInterface(String appPackage) {
		BusinessClientsResourceGroup srg = (BusinessClientsResourceGroup)
		    getResourceGroup();
		return new MailImpl(appPackage, this, srg.getContext(appPackage));
	}

	public IBinder getMockedAndroidInterface(String appPackage) {
		return new MailMockImpl(appPackage, this);
	}

	public IBinder getCloakedAndroidInterface(String appPackage) {
		return new MailCloakImpl(appPackage, this);
	}

	boolean verifyAccessAllowed(String appPackage, String privacySetting) {
		BooleanPrivacySetting bpl = (BooleanPrivacySetting)
		    getPrivacySetting(privacySetting);
		try {
			return bpl.permits(appPackage, true);
		} catch (PrivacySettingValueException e) {
			e.printStackTrace();
			return false;
		}
	}
}
```

Listing 4.3 Aufbau des AIDL-Interface.

```
interface IMail {
	void sendMail(String mailaddy);
}
```

4.2.2 Erstellen der Ressourcengruppe

In der für diese Arbeit entwickelten Applikationen werden wichtige Funktionen über die BYOD-Plattform aufgerufen. Hierzu müssen wie oben erwähnt, die entsprechenden Ressourcen in einer Ressourcengruppe erstellt und zusammengefasst werden.

Um eine Ressourcengruppe zu erstellen wird eine neue Klasse erstellt, die um die Klasse 'ResourceGroup' erweitert wird. In unserem Fall fasse ich sämtliche für die Beispielapplikation erforderlichen Ressourcen entsprechend ihrem Verwendungszweck inhaltlich zusammen. Listing 4.5 zeigt den Aufbau der entsprechenden Ressourcengruppe.

Listing 4.4 Implementierung der 'MailRessource' zur Rückgabe korrekter Werte.

```
import de.unistuttgart.ipvs.pmp.resourcegroups.businessclients.IMail.Stub;

public class MailImpl extends Stub {

	private String appIdentifier;
	private MailResource resource;
	private Context context;

	public MailImpl(String appIdentifier, MailResource resource, Context context) {
		this.appIdentifier = appIdentifier;
		this.resource = resource;
		this.context = context;
	}

	public void sendMail(String mailaddy) throws RemoteException {
		if (!this.resource.verifyAccessAllowed(this.appIdentifier,
			BusinessClientsResourceGroup.PS_SEND_MAIL)) {
				throw new SecurityException();
		}

		Intent intentMail = new Intent(Intent.ACTION_SENDTO);
		String uriText = "mailto:" + Uri.encode(mailaddy) + "?subject="
			+ Uri.encode("") + "&body="
			+ Uri.encode("");
		Uri uri = Uri.parse(uriText);
		intentMail.setData(uri);
		intentMail.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
		context.startActivity(Intent.createChooser(intentMail, "Send mail..."));
	}

}
```

Listing 4.5 Klasse der Ressourcengruppe.

```
public class BusinessClientsResourceGroup extends ResourceGroup {
	public static final String PACKAGE_NAME =
	    "de.unistuttgart.ipvs.pmp.resourcegroups.businessclients";

	public static final String R_CELL = "cellResource";
	public static final String R_MAIL = "mailResource";

	public static final String PS_CALL_PHONE = "callPhone";
	public static final String PS_SEND_MAIL = "sendMail";

	public BusinessClientsResourceGroup(IPMPConnectionInterface pmpci) {
		super(PACKAGE_NAME, pmpci);
		registerResource(R_CELL, new CellResource(this));
		registerPrivacySetting(PS_CALL_PHONE, new BooleanPrivacySetting());
		registerResource(R_MAIL, new MailResource(this));
		registerPrivacySetting(PS_SEND_MAIL, new BooleanPrivacySetting());
	}
}
```

Abbildung 4.6: Ressourcengruppe für die Applikation zur Kontaktaufnahme mit Geschäftskunden.

Damit die Ressourcengruppe von Applikationen gefunden werden kann ist es nötig der erstellten Klasse einen Identifikationsstring zuzuweisen. Hierzu wird ein Konstruktor angelegt, der ein Objekt vom Typ `IPMPConnectionInterface` übergeben bekommt. In diesem Konstruktor wird dann der Konstruktor der Basisklasse aufgerufen und die Identifikation zusammen mit der Referenz des erhaltenen IPMPConnectionInterface-Objekts übergeben.

Mittels der Funktion `registerResource(..)` werden im Konstruktor erstellte Ressourcen der entsprechenden Ressourcengruppe zugewiesen. Als Parameter werden der Funktion ein Identifikationsstring und das Objekt der Ressource übergeben.

Um anschließend die benötigten Privacy Settings der Ressourcen auf die entsprechende Ressourcengruppe zuzuweisen, wird die Methode `registerPrivacySetting(..)` verwendet.

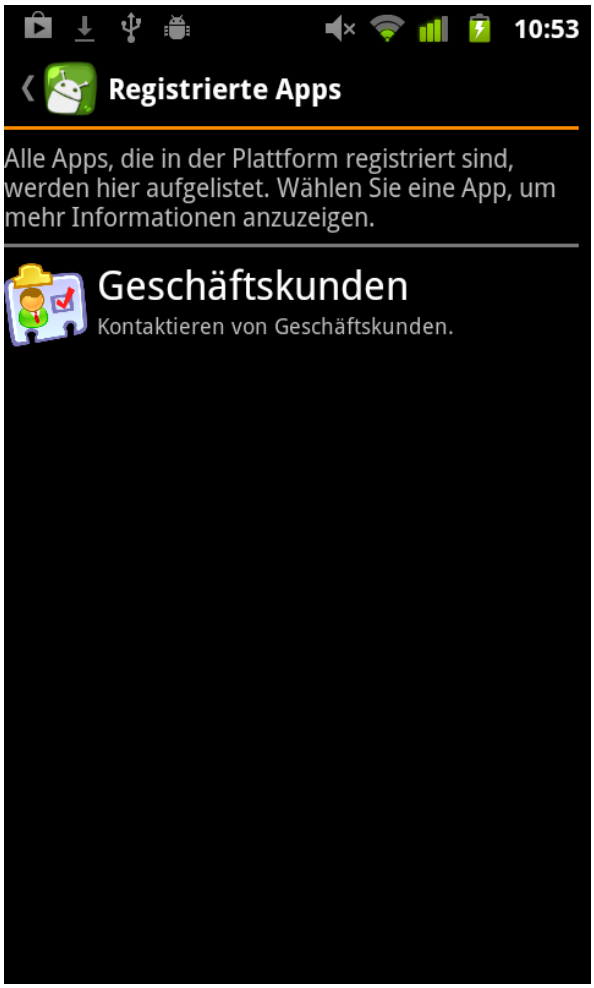

Abbildung 4.7: Der Menüpunkt 'Apps' zeigt installierte Anwendungen, die in der Plattform registriert sind.

Übergeben werden als Parameter ein entsprechender Identifikationsstring und der Typ der Privacy Setting.

4.3 Installation von Firmen-Anwendungen

Die zur Installation freigegebenen Applikationen werden von der jeweiligen Firma bereitgestellt. Bei der Installation und anschließenden Ausführung werden diese in der Verwaltungsplattform registriert und die zulässigen Service Features entsprechend der Richtlinien in der Konfigurationsdatei, welche zuvor vom Firmenserver geladen wurde, automatisch eingerichtet. Installierte Applikationen können im Menüpunkt 'Apps', wie in Abbildung 4.7 zu sehen ist, eingesehen werden.

4.4 Anwendungsbeispiel

Um die Fähigkeit des hier umgesetzten BYOD-Konzeptes zu testen und besser darzustellen, wurde abschließend eine App für das Android-Betriebssystem entwickelt, welche einige Ressourcen der BYOD-Verwaltungssoftware nutzt. Folgende Funktionen sind Bestandteil dieser Applikation:

- Verschlüsseltes Herunterladen und Auslesen einer Kundendatenbank
- Kontaktaufnahme mit Firmenkunden (Anruf, Email, etc)
- Zugriff auf Kundendaten nur während des Aufenthalts innerhalb des Firmengeländes
- Kontaktaufnahme nur zu Arbeitszeiten

4.4.1 Erster Start

Wie schon im Abschnitt 3.4.3 und erklärt, durchläuft jede neu installierte Applikation den Registrierungsprozess der BYOD-Plattform. Wrid dieser nicht erfolgreich abgeschlossen, kann die Applikation nicht verwendet werden. Nach einer fehlerhaften Registrierung kann die Applikation neu gestartet werden und der Registrierungsvorgang beginnt von vorn.

Abbildung 4.8 zeigt den Registrierungsprozess der Beispielapplikation. Folgende Punkte werden abgehandelt:

- `Prepare & detect plattform` - hierbei wird die Plattform auf dem Gerät gesucht und falls gefunden die überprüft, ob diese Lauffähig ist.
- `Check registration` - Ist die Applikation bereits registriert, startet die Applikation, falls nicht, geht es im nächsten Schritt weiter.
- `Registration at plattform` - Der eigentliche Registrierungsprozess.
- `Set initial Service Features` - Hierbei wird im Vergleich zur PMP, nicht der Benutzer nach den Zugriffen gefragt, sondern aus der Konfigurationsdatei geladen.
- `Open the App` - Anschließend kann durch einen Klick auf die Schaltfläche die Applikation gestartet werden.

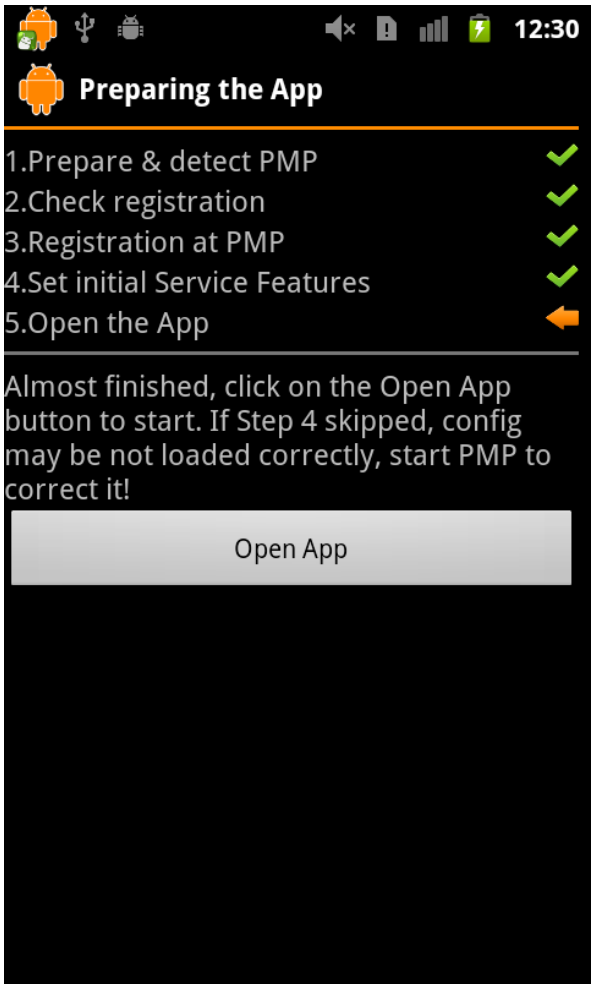

Abbildung 4.8: Registrierugsprozedur der Beispielapplikation.

4.4.2 Funktionen der Applikation

Hat die Applikation den Registrierungsprozess überstanden und wurde anschließend vom Benutzer gestartet, begrüßt ihn ein Firmenkontaktbuch wie es in Abbildung 4.9, welches geschmückt mit Geschäftskunden ist. Das Programm lässt die Kontaktaufnahme mittels Telefon und Email zu. Mit dem entsprechenden Klick auf das Telefon oder den Briefumschlag neben jener Person, die man kontaktieren möchte, öffnet sich entweder der Telefondialog, oder das E-Mail-Programm.

Für die entsprechenden Funktionen wurden Ressourcen in einer Ressourcengruppe erstellt. Die Ressource `Mail` in der Ressourcengruppe `BusinessClientsResourceGroup`, ist für das 'Service Feature' zur Nutzung der E-Mail-Berechtigung nötig. Analog dazu existiert die Ressource `Call`, welche für das 'Service Feature' zur Nutzung der Telefonfunktion benötigt wird.

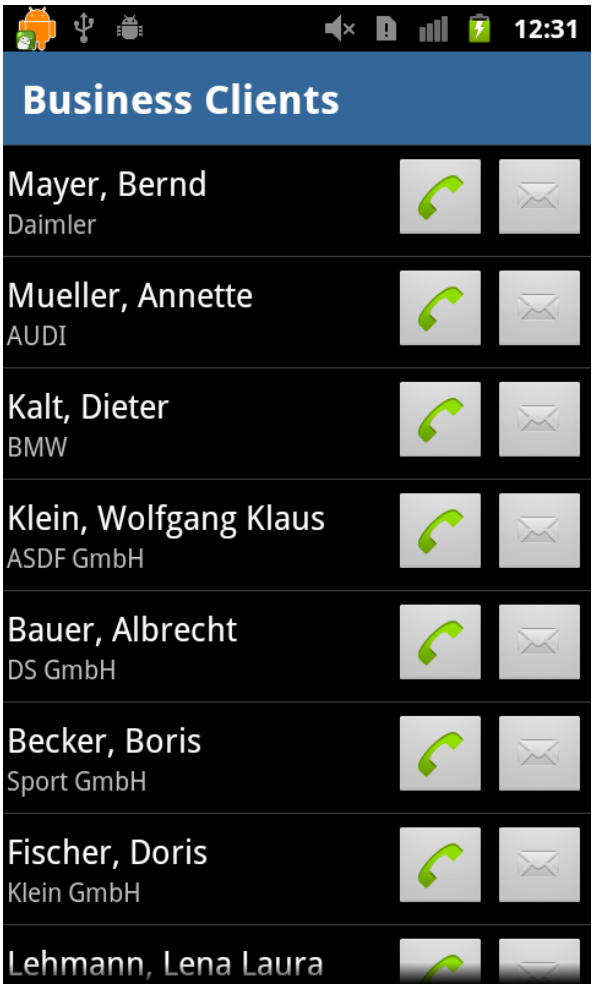

Abbildung 4.9: Beispielapplikation zur Kontaktaufnahme von Firmenkunden und -kontakte.

In der Beispielkonfiguration unseres Beispielunternehmens sind die 'Service Features' für die Funktionen zeitlich begrenzt. Die Telefonfunktion ist dem Mitarbeiter nur für den Zeitraum während seiner Arbeitszeit erlaubt. Die E-Mail-Funktion entsprechend etwas länger, allerdings nicht über Nacht. Gleichzeitig ist es dadurch nicht möglich Telefonnummern oder E-Mail-Adressen aus dem Programm zu extrahieren um nach Feierabend mit Geschäftskunden zu kommunizieren.

4.5 Verbesserungen

4.5.1 Sicherheit von Programmdaten

Alle gespeicherten Daten werden aus Sicherheitsgründen in einem Bereich auf dem Android-Gerät gespeichert, auf welchen Dritte keinen Zugriff haben. Dazu wird vom Android-Betriebssystem jeder Anwendung ein eigener Programmspeicher in einem Unterordner unter `/data/data/` zugeordnet. Differenziert werden Zugriffe und Namen der Unterordner über den Paketnamen der Applikation. Im Fall der BYOD-Plattform lautet der komplette Pfad zum genannten Ordner `/data/data/de.unistuttgart.ipvs.pmp/`. In diesem Ordner finden sich neben den Dateien, welche von der Applikation gespeichert werden, auch Datenbanken, Cache-Dateien und Einstellungen der Shared Preferences.

4.5.2 Root-Zugriffsrechte

Wie bereits erwähnt kann dieser Bereich nur aus derjenigen Applikation aus abgerufen werden, bei welchem der Paketname mit dem Pfad übereinstimmt. Über einen Datei-Manager können diese Dateien ebenso lokalisiert werden, allerdings von Haus aus nicht geöffnet oder bearbeitet werden. Sind hingegen Root-Rechte auf dem Gerät freigeschaltet worden, so kann er ohne Einschränkungen sicher geglaubte Dateien und Datenbanken ungehindert bearbeiten, auslesen oder sogar herunterladen. Deshalb gilt es sicherzustellen, dass Anwender der BYOD-Plattform kein gerootetes Gerät besitzen. Dies ist Möglich indem das Ausführen von Applikationen über die BYOD-Plattform generell verhindert wird, sollten Root-Rechte auf dem Gerät freigeschaltet sein.

Eine andere Methode wäre die Kodierung der Daten. Allerdings muss dabei die Verschlüsselungsmethode auf Android-Ebene implementiert werden, um den anwendungseigenen Speicher zu chiffrieren. Derartige Eingriffe in das Betriebssystem erfordern einen höheren Aufwand und sind demnach weniger empfehlenswert.

Eine weitere Verschlüsselungsmethode wäre das Kodieren auf Programmebene. Dazu werden die Daten vor dem Abspeichern verschlüsselt, und beim Auslesen entschlüsselt. Hierzu ist es jedoch erforderlich einen anderen Speicherort für die Programmdaten zu wählen, da der von Android zugeteilte Programmspeicher nicht jede beliebige Dateiarten und Speichermethoden unterstützt. Diese Methode ist aufgrund des geringeren Aufwands und ähnlichem Sicherheitslevel zu bevorzugen.

Beide Vorgänge erfordern einen größeren Aufwand und sind deshalb nicht Bestandteil dieser Arbeit. In dieser Ausführung wird davon ausgegangen und festgelegt, dass die BYOD-Plattform aus Sicherheitsgründen nur auf nicht-gerooteten Geräten ausgeführt werden darf.

5 Verwandte Arbeiten und Vergleich mit weiteren Lösungen

In diesem Kapitel werden Forschungskonzepte und weitere Lösungen für das BYOD-Problem näher erläutert. Für eine bessere Darstellung und leichtere Verständlichkeit werden diese abschließend mit der in dieser Arbeit entwickelten Lösung verglichen und abgegrenzt.

5.1 Forschungskonzepte

Dieser Abschnitt befasst sich mit Ansätzen, die für die Nutzung in BYOD-Lösungen in Betracht gezogen werden können. Diese bieten keine komplette Lösung, können aber als wichtige Bestandteile für ein fertiges Produkt genutzt werden.

5.1.1 TrustDroid

TrustDroid ist ein Analysetool, welches basierend auf 'Taint Tracking' [ZO12] den Verlust von sensiblen Daten über Smartphones in Unternehmensnetzwerken verhindern soll. Dazu untersucht das Tool fertig-compilierte Android-Applikationen, welche standardmäßig in einer APK-Datei vorliegen. Durch semantische Analyse kann erkannt werden, ob sensible Daten durch jene Applikation verloren gehen können.

Es existieren zwei Modi, in welchen TrustDroid eine Analyse durchführen kann. Findet eine Analyse 'offline' statt, so werden mögliche Applikationen einmalig bei Aufruf analysiert. Findet eine Analyse allerdings in Echtzeit statt, so mindert dies die Leistungsfähigkeit und Batteriestatus des Gerätes.

Mithilfe der 'Dalvik virtual machine', welche beim Android-Betriebssystem zum Einsatz kommt, wird die semantische Analyse um einiges erleichtert. Alle Dalvik-Befehle werden durch die virtuelle Maschine der Semantik nach angeordnet und kategorisiert, so dass die Taint-Regeln für das Analysewerkzeug TrustDroid sehr einfach eingerichtet werden können.

5.1.2 2TAC

'2-Tier Access Control' (2TAC) [CCE+12] ist ein Cloud-basiertes Sicherheitstool, das mittels verschiedener Sicherheitsmechanismen sowohl auf dem Gerät als auch in der Cloud sicherstellt, dass nur autorisierte Nutzer und Applikationen Zugang zu sensiblen Daten eines Unternehmens gelangen.

In der ersten Schicht befindet sich auf dem Gerät ein Kontrollmechanismus, der verschiedene Profile für den Benutzer enthält. Jedes Profil hat dabei nur einen begrenzten Funktionsumfang. Die Profile werden anhand von Zeit und Ort gewechselt, um den Nutzer die für ihn brauchbaren und erlaubten Funktionen anzubieten.

Des Weiteren bietet die erste Schicht einen ersten Anti-Malware scanner. Durch die begrenzte Akkulaufzeit und Leistung der Smartphones handelt es sich hierbei um eine 'Light'-Version einer Anti-Malware Software. Das Ausführen der Software kann entweder von Hand durch den Benutzer, oder ferngesteuert über die Cloud erfolgen.

In der Cloud, und somit in der zweiten Schicht, können im 'Profile Management System' Profile angelegt, bearbeitet oder entfernt werden. Diese werden wiederum von den Geräten in Schicht eins verwendet. Die 'Anti-Malware'-Software in der Cloud kopiert bei Ausführung ein Abbild des laufenden Gerätes in die Cloud und scannt es anschließend. Weitere Maßnahmen können anschließend von der Cloud aus vorgenommen werden. 'Access Records' speichern Bewegungsdaten eines jeden Gerätes. Diese können beliebig von Administrationen geprüft und ausgewertet werden. Das 'Trust Management System' ist für den Zugriff auf die sicheren Daten und die Vernetzung der Geräte untereinander verantwortlich.

5.2 Lösungskonzepte aus der Industrie

Dieser Abschnitt befasst sich mit weiteren Projekten, welche sich sowohl mit den verbesserungswürdigen Sicherheitskonzepten bestehender Betriebssysteme auseinandersetzen, als auch mit Lösungen für das 'bring-your-own-device'-Problem.

5.2.1 Security Enhanced Android

Security Enhanced Android (SEAndroid) ist eine Anpassung des Android Betriebssystem, welche sich mit den Lücken im Sicherheitskonzept von Android befasst. SEAndroid entstand aus SELinux, welches wiederum eine Modifikation für Linux-Systeme auf Kernel Ebene darstellt. Somit wird mithilfe von SEAndroid der Funktionsumfang von SELinux auf Android, welches auf Linux basiert, portiert. SEAndroid bietet darüber hinaus eine Referenz-Implementierung für die Einbindung in den Android-Kernel mit zusätzlichen Demonstrationen für gestopfte Sicherheitslücken. Eine dieser Sicherheitslücken beschreibt der meist praktizierte „Root"-Exploit, welcher dem Benutzer Zugriff auf alle versteckten Dateien und Funktionen gewährt.

Sowohl SELinux als auch SEAndroid werden maßgeblich von der NSA entwickelt und eingesetzt. Allerdings bietet die NSA keine fertig kompilierten Versionen an, sondern stellt ausschließlich den Quellcode zur Verfügung, welcher mit dem offiziellen Android Open Source Project verschmelzt werden kann. Dadurch lässt sich eine individuelle Android-Version erstellen zur Weiterentwicklung.

5.2.2 Samsung Knox

Knox ist eines dieser Projekte, welches mithilfe von SEAndroid entstanden ist. Knox ist eine BYOD-Lösung von Samsung, welche im Februar 2013 auf dem Mobile World Congress in Barcelona vorgestellt wurde. Samsungs Kurzfassung für diesen Dienst „A New Solution for Work and Play" suggeriert ein Mittel um Arbeit und Privates auf einem Smartphone unter einen Hut zu bringen. Wie auch bereits SEAndroid kommt dieses Projekt ausschließlich Android Smartphones zu Gute.

Neben SEAndroid, welches zur Unterscheidung der Daten zwischen privat und geschäftlich dient, wird eine Technologie namens Secure Boot eingesetzt. Mittels Secure Boot können auf dem Smartphone nur noch verifizierte und authorisierte Applikationen installiert und ausgeführt werden. Zusätzlich schützt der Dienst TIMA (TrustZone-based Integrity Measurement Architecture) vor unrechtmäßigem Eingriff in den Android-Kernel oder Bootloader. Im Falle einer Intervention werden dementsprechend voreingestellte Aktionen ausgeführt. Eine der möglichen Handlungen ist das Abschalten und Deaktivieren des Gerätes.

Knox ist hauptsächlich ein Dienst entwickelt für Arbeitgeber und soll demnach empfindliche Daten des Unternehmens schützen. Dazu trennt es nicht nur Privates von Geschäftlichem, sondern verschlüsselt und versiegelt die Daten des Unternehmens zusätzlich.

Für die Speicherung der Daten verwendet Samsung bei seinem Dienst Knox einen Container, welcher eine isolierte und sichere Umgebung innerhalb des Smartphones darstellt. Es handelt sich dabei um einen zweiten Launcher bzw. Homescreen, ähnlich der Android Oberfläche, in welcher die geschäftlichen Applikationen und Daten ausschließlich nutzbar sind. Für den Start des Containers soll lediglich ein Klick auf das entsprechende Icon in der üblichen Android-Oberfläche möglich sein. Die Nutzung der Ressourcen des Unternehmens in einer anderen Umgebung vermeidet das Vermischen von privaten mit geschäftlichen Daten. Dies soll Abbildung 5.1 nochmals grafisch darstellen.

Für die Verschlüsselung kommt das Advanced Encryption Standard (AES) System zum Einsatz mit einem 256-Bit Schlüssel. Um schließlich die Verbindung zwischen Firmenserver und Smartphone zu schützen wird ein Virtual Private Network (VPN) aufgebaut, indem über den Firmenserver ein VPN-Profil auf das Gerät ge'push't wird. Dies beinhaltet zusätzlich die Möglichkeit, individuelle Anpassungen und Restriktionen für jeden Mitarbeiter vorzunehmen. Diese und weitere Sicherheitskonzepte von Samsung Knox sind in Abbildung 5.2 grafisch dargestellt und zeigen die detailierte Erweiterung der Konzepte des Standard-Android-Betriebssystems.

Abbildung 5.1: Kozept der zwei unabhängigen Oberflächen von Samsung Knox [Sam13].

Samsung bietet mit Knox ein für den Nutzer unabhängig voneinander getrenntes Erlebnis der Android Software mit scheinbar zwei Platformen für den privaten und geschäftlichen Gebrauch. Laut Samsung erfreuen sich Unternehmen an der dazu gewonnen Sicherheit ihrer Daten und Applikationen, und der einfachen Anbindung an jede Mobile-Device-Management-Software. Außerdem verspricht Samsung eine einfache und schnelle Anbindung in das Android-Betriebssystem auch für neue Geräte.

Kurzgefasst soll Samsung Knox dafür Sorgen, dass sich aus einem einzigen Android-Gerät zwei Smartphones machen lassen. Sowohl Daten als auch Kontakte werden durch die Software in zwei unabhängig voneinander getrennten Bereiche abgespeichert. Im zweiten, untergeordneten Bereich, lassen sich nur Applikationen starten, welche vom Administrator freigegeben wurden. Dies kann der Chef, ein Systemadministrator im Unternehmen oder der Besitzer selbst sein, je nach Einsatzort. Nicht nur Kontakte sondern auch Fotos der Kamera werden extra abgespeichert und verschlüsselt. Somit sollen unberechtigte Zugriffe auf die entsprechenden Bilder ausgeschlossen werden [Sam13].

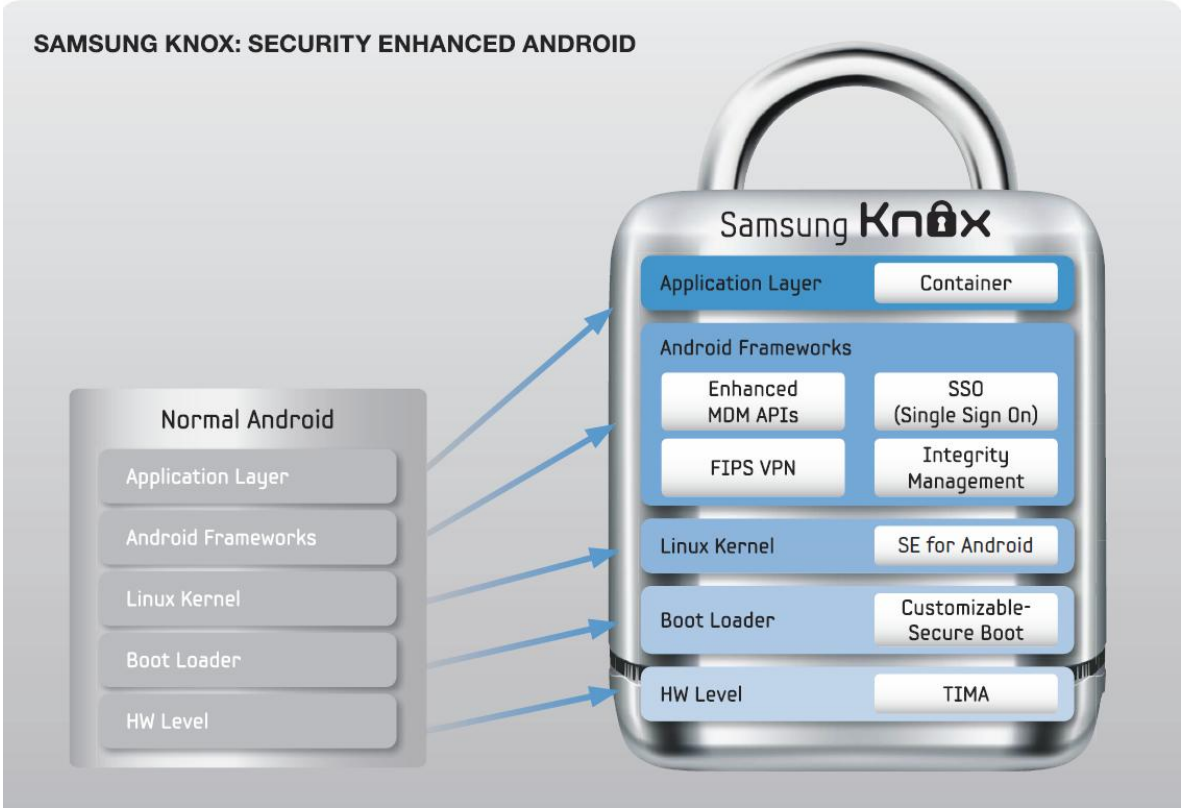

Abbildung 5.2: Sicherheitsaspekte von Samsung Knox [Sam13].

Um zwischen den Beiden Oberflächen hin und her zu wechseln reicht ein einfacher Klick auf das entsprechende Icon. Zum Schutz vor unberechtigtem Zugriff dient eine Passwortabfrage wie in Abbildung 5.3 zu sehen ist. Bei dreimaliger Passworteingabe lassen sich automatisch alle Daten löschen. Das entsprechende Profil kann auch an den Ort gebunden werden und automatisch gewechselt werden. So kann eingestellt werden, dass sobald sich der Benutzer in der Nähe seiner Arbeitsstelle aufhält, das Profil geändert wird. Auch kann vom Administrator das Kameramodul für die Zeit auf dem Firmencampus deaktiviert werden.

Samsung Knox kann allerdings auch für andere Gebiete eingesetzt werden. Durch die zusätzliche Oberfläche im Betriebssystem lässt sich auch schnell ein Spiele-Bereich für Kinder einrichten. Hier kann dann vom Elternteil der Zugriff auf Apps und Daten eingeschränkt werden um einerseits die Sicherheit der Kinder zu gewährleisten, und andererseits die Daten des Besitzers vor unbeabsichtigtem Löschen zu schützen [Her13].

Im Mai 2013 erklärte das amerikanische Department of Defense die Business Variante Samsung Knox als sicher. Dies gestattet nun den Einsatz von Samsung Geräten mit der Sicherheitssoftware Knox im geheimen Dienst der Vereinigten Staaten von Amerika. Bisher waren nur BlackBerrys für eine sichere Kommunikation zertifiziert. Allerdings wird auch

Abbildung 5.3: Log-In Anzeige beim Eintreten in den geschützten Bereich.

erwartet, dass die neueste Version des mobilen Betriebssystems iOS von Apple ebenfalls die gleiche Freigabe erhalten soll [Bru13, Ale13].

Nachdem die Malware-Attacken auf Android-Geräte in den letzten Jahren exponentiell gestiegen sind, ist das vom Department of Defense ausgezeichnete Gütesiegel für Android ein wichtiger Meilenstein. Es ist zudem gleichzeitig ein Anzeichen für den potenziellen Einsatz bei professionellen Kunden weltweit. Der Standard, welcher vom US-Verteidigungsministerium vorausgesetzt wird ist bei der gewerblichen Nutzung von mobilen Geräten in den meisten Unternehmen ein guter Richtwert. Viele Arbeitgeber orientieren sich in diesem Bereich an den Anforderungen des DoD [Ser13, Hen].

Ursprünglich sollte Samsung Knox als Bestandteil des im April 2013 auf den Markt gekommenen Gerätes, dem Samsung Galaxy S4, erscheinen. Allerdings wurde kurz vor Veröffent-

lichung zurückgerudert und das Gerät vorerst ohne die Sicherheitssoftware ausgeliefert. Unabhängig voneinander bestätigten Mitarbeiter von Samsung, dass sich die Software verspätet, da sowohl intern als auch mit verschiedenen Providern ausgiebige Tests noch bevorstehen [Che13, Eng13].

5.2.3 AppSense MobileNow

AppSense präsentiert mit seinem Produkt MobileNow eine komplette Lösung für das BYOD-Problem, welches, laut AppSense, das einzige Konzept als Software-as-a-Service umgesetzt ist. MobileNow bietet Unterstützung sowohl für Applikationen als auch für Daten von Unternehmen und Benutzer. Unternehmen bekommen für den Einsatz von MobileNow einen Service, welcher die Sicherheit und Kontrolle in Echtzeit unterstützt. Der Ansatz dieser Lösung benötigt im Vergleich zu anderen Konzepten nicht die komplette Kontrolle der Geräte seiner Mitarbeiter. Des Weiteren wird durch MobileNow der Umfang der unterstützten Applikationen nicht eingeschränkt. Das Anpassen schon existierender Anwendungen auf die BYOD-Plattform ist in diesem Fall nicht nötig und wird von Haus aus unterstützt.

MobileNow ist ein Dienst für Unternehmen und dessen Mitarbeiter, welcher vom Entwickler AppSense verwaltet und zur Verfügung gestellt wird. Nach der einfachen Installation auf verschiedene Android- oder iOS-Versionen fügt sich das Programm nahtlos in das Betriebssystem ein. MobileNow unterstützt sowohl die Administration von mobilen Geräten, als auch die Verwaltung der verwendeten Applikationen. Die Echtzeitunterstützung von MobileNow mindert den Verwaltungsaufwand gewaltig und bietet hohe Flexibilität.

MobileNow unterstützt außerdem:

- Erkennung von Malware und anderer schädlicher Software.
- Erkennung von Jailbreak (iOS) und Root (Android).
- Verschlüsselung der anwendungsspezifischen Daten.
- Erweiterte Sicherheit für native Email-Programme der Betriebssysteme.

Die Technik, welche bei MobileNow zum Einsatz kommt. wurde ursprünglich von RAPsphere entwickelt. Nach der Übernahme durch AppSense wurde die Verwaltungssoftware für mobile Geräte weiterentwickelt und schließlich zum Endprodukt MobileNow abgeschlossen. Die Weiterentwicklung umfasst zum einen die erweiterte Administration von Benutzern und Geräte, welche mit dem gemeinsamen Service kommunizieren. Die zweite wichtige Erweiterung ist die Konfigurationsmöglichkeit von installierten Applikationen, die zu Arbeitszwecken eingesetzt werden. Dabei ist der Zugriff des Administrators auf die unternehmensrelevanten Applikationen und Daten beschränkt. Private Daten der Benutzer und Eigentümer der Geräte bleiben unantastbar.

Dabei kommt eine Wrapping-Technik zum Einsatz, welche die Applikationen abkapselt. Zusätzliche Policies, welche es durch MobileNow zu vergeben gibt, schränken leiten die

gewünschten Rechte jeder Applikation individuell weiter. Die Vergabe der Zugriffsberechtigungen kann entweder für das gesamte Unternehmen, auf Gruppen und Abteilungen oder sogar auf einzelne Benutzer individuell eingerichtet werden [Hul13b].

AppSense bietet für sein Produkt zwei Preismodelle an. Wird nach der Anzahl der Benutzer gezahlt, so fallen 3,83 € pro Monat je Benutzer an Kosten an. Entscheidet sich ein Unternehmen für das Preismodell nach der Anzahl der Geräte, werden 2.67 € pro Gerät jeden Monat in Rechnung gestellt. Die Mindestvertragslaufzeit beträgt 12 Monate.

5.2.4 BlackBerry Enterprise Service

BlackBerry hat Anfang 2013 mit dem Enterprise Service mit der Versionsnummer 10 eine Plattform vorgestellt, die es erlaubt mobile Geräte einfach und organisiert zu verwalten. Auch wenn die Versionsnummer anderes vermuten lässt, ist dies die erste Software von BlackBerry mit einem solchen Funktionsumfang. Die Ziffer der Version wurde lediglich vom Betriebssystem BlackBerry 10, auch BlackBerry X (BBX) genannt, des kanadischen Herstellers übernommen, welches in etwa zeitgleich veröffentlicht wurde. Mit der Einführung der letzten Ausgabe des Betriebssystem wurde das Unternehmen entsprechend ihrem Produkt in BlackBerry umbenannt.

Mithilfe des Enterprise Service 10 lassen sich über eine einheitliche Plattform sowohl BlackBerry-Smartphones, BlackBerry-Tablets als auch Geräte mit Apples iOS und Googles Android OS verwalten. Zum Umfang des Enterprise Service gehören:

- BlackBerry Management Studio - eine Web-Applikation zu Administrationszwecken. Verwaltung von mobilen Geräten, Benutzer, Konfigurationen, Richtlinien uvm.
- BlackBerry Enterprise Server - zur Verwaltung der firmeneigenen BlackBerry-Geräte.
- BlackBerry Enterprise Server Express - zur Verwaltung der privaten BlackBerry-Geräte von Mitarbeitern im und für das Unternehmensumfeld.
- BlackBerry Device Service - BlackBerry-Playbook Tablets werden über diesen Teil verwaltet.
- Universal Device Service - bietet Funktionen zur Verwaltung von Geräten, die unter Googles Android OS oder Apples iOS laufen.

Alle Funktionen lassen sich über eine einzige Konsole ansteuern und verwalten. Benutzer können erstellt und zu Gruppen hinzugefügt werden. Ebenfalls können Benutzerprofile angepasst werden und den entsprechenden Geräten zugewiesen werden.

Wird ein BlackBerry Playbook zur Verwaltung in den BlackBerry Enterprise Service eingetragen, so lässt sich dieses mittels dem Device Service einem Benutzer zuordnen. Über die zentralisierte Verwaltungsplattform lässt sich dieses anschließend für den Einsatz im Unternehmen konfigurieren. Es lassen sich Passwortanforderungen konfigurieren, Verschlüsselung der geschäftlichen Daten aktivieren und sowohl das Tablet aus der Ferne sperren, als auch die auf dem Gerät befindlichen Geschäftsdaten löschen.

Anzahl der Lizenzen	Einmalige Lizenzkosten
1 - 99	€ 78 ,-
100 - 499	€ 76 ,-
500 - 999	€ 75 ,-
1000+	€ 71 ,-
	Alle Preise exkl. Ust.

Tabelle 5.1: Preisstaffelung der Kosten von BES.

Ein weiteres Feature des BlackBerry Enterprise Service ist die bereitstellung von Updates und App-Installation Over-The-Air (OTA). BlackBerry-Geräte lassen sich somit noch sicherer machen und kritische Sicherheitslücken schneller schließen. Eingebundene BlackBerry-Geräte des Unternehmens können in einem weiteren Katalog der BlackBerry-AppWorld unter optionalen Applikationen wählen, welche vom Unternehmen vorher geprüft und individuell freigegeben wurden.

Für Geräte mit Apples iOS oder Googles Android OS steht der Universal Device Service zur Verfügung. Dieser ermöglicht eine ähnliche Benutzerverwaltung für Android- und iOS-Geräte. Hinzu kommt das Definieren von IT-Richtlinien und Verbindungseinstellungen. Außerdem lassen sich Geräte bei Verlust lokalisieren und installierte Anwendungen auf dem Gerät verwalten.

Die wohl interessanteste Technologie für den parallelen Einsatz im privaten und beruflichen Umfeld ist BlackBerry Balance, welche schon vor dem BlackBerry Enterprise Service als Bestandteil von BlackBerry Mobile Fusion die Runde machte. Wird ein Gerät mit dem Enterprise Service von BlackBerry gekoppelt, so wird automatisch ein verschlüsselter Bereich für die geschäftlichen Informationen eingerichtet. Dadurch werden private und geschäftliche Daten sicher gespeichert und streng voneinander getrennt.

Die Tabelle zeigt die einmaligen Lizenzkosten, die es zu entrichten gilt für den Einsatz des BlackBerry Enterprise Service, in Abhängigkeit von der Anzahl der zu verwaltenden Geräte. Hinzu kommen die monatlichen Gebühren je Gerät, welche üblicherweise an Reseller in Form von Mobilfunkanbieter abgetreten werden müssen. Diese sind generell in den Laufzeitverträgen inbegriffen. Des Weiteren bietet BlackBerry seit Einführung der neuesten Ausgabe eine kostenlose Testversion, die auf 60 Tage und 20 Geräte begrenzt ist.

Durch den angepriesenen Neustart BlackBerrys und die damit verknüpfte Namensänderung und Präsentation der neuesten Version des mobilen Betriebssystem hat sich allerdings nicht viel verändert. Auch der Trend des Bring-Your-Own-Device-Konzeptes, welches bei BlackBerry von Anfang an am ehesten Verwendung fand ist keine Hilfe. Trotz des Quartalsgewinn im vierten Geschäftsquartal 2012 werden weiterhin Stellen abgebaut [Klo13].

5.2.5 AppTec Enterprise Mobile Manager

Das Schweizer Unternehmen AppTec hat im Mai 2013 nach einem umfangreichen Test mit 700 Nutzern seine BYOD-Lösung mit dem Namen Enterprise Mobile Manager 2013 (EMM) veröffentlicht. Der EMM kombiniert die Verwaltung mobiler Geräte mit zusätzlichen Funktionen zur Verschlüsselung von E-Mails, Kontakt- und Kalendereinträge.

Neben der Konfiguration von WLAN, VPN und der eingebauten Kamera des Mobilgerätes, lassen sich vom Administrator Richtlinien für verwendete Passwörter einrichten. Ein vom Administrator eingerichteter App-Store bietet dem Nutzer eine Auswahl aus Applikationen, die vom Unternehmen freigegeben worden sind, für die Verwendung zu Arbeitszwecken. Bei Verlust eines Gerätes lässt sich dieses durch den Enterprise Mobile Manager 2013 aus der Ferne orten, sperren und, falls erwünscht, sogar komplett löschen. Ein weiterer Sicherheitsmechanismus ist der automatische Versand einer SMS an eine vorkonfigurierte Nummer bei unberechtigtem Austausch der SIM-Karte auf einem der registrierten Geräte [Sch13].

Der Enterprise Mobile Manager [App13] von AppTec stellt folgende Funktionen zur Verfügung:

- Nur authorisierte Geräte erhalten Zugriff auf Unternehmensdaten und E-Mails.
- Bei Verlust eines Gerätes, kann mit einem Klick vor Missbrauch geschützt werden.
- Automatische Installation von Firmenapplikationen Over-the-Air.
- Daten auf dem Gerät können zentral gesichert und wiederhergestellt werden.
- Fremder Zugriff ist nicht möglich.
- Zusammenfassender Bericht aller verbundene und registrierter Geräte auf einem Blick.

Zum Zeitpunkt der Veröffentlichung wurde vorerst nur die Unterstützung für iOS ab Version 3.0 und Android ab Version 2.3 eingeräumt. Allerdings lässt sich AppTec die Option offen für die Vorstellung einer Version mit der Einbindung von Windows Phone mit der Version 8.

Seit der Veröffentlichung (Stand: Mai 2013) kann der Dienst im Umfang von 90 Tagen kostenlos getestet werden. Firmenkunden mit bis zu 25 Geräten erhalten den kostenlos. Einzige Einschränkung ist der nicht enthaltene Support für die kostenlose Version. Bei Inanspruchnahme des Supports fallen Gebühren an.

Für die kostenpflichtige Version gibt es mehrere Preismodelle. Für den vollständigen Kauf der Software mit unbegrenzter Laufzeit verlangt AppTec einmalig 19,00 €Ḟür das kontinuierliche Upgrade und den Support werden 20 Prozent des Betrages fällig. Wird die Software nach Kauf allerdings nicht auf einem firmeneigenen Server gehostet, und somit der Hosting-Dienst von AppTec in Anspruch genommen, so fallen zusätzlich pro Gerät und Monat 0,49 € an Kosten an. Das Hosting-Angebot ist zusätzlich an eine Mindestvertragslaufzeit von 24 Monaten gebunden. Die Wartung ist inklusive.

Will man im Gegensatz dazu den Dienst als Service mieten, so fallen lediglich 0,99 € pro Gerät und Monat an Kosten an. Upgrade und Support sind im Gegensatz zum Kauf der Software kostenlos mit inbegriffen. Allerdings wird man auch hier an die Mindestvertragslaufzeit von 24 Monaten gebunden. Ebenso fallen bei Hosting über AppTec die üblichen monatlichen Kosten pro Gerät an.

Alle hier genannten Preise sind vom Mai 2013 und verstehen sich zzgl. der gesteztlichen MwSt.

5.3 Vergleich

5.3.1 Mobile Device Management

Unter Mobile Device Management (MDM) versteht man eine zentralisierte Verwaltung von mobilen Endgeräten. Dazu gehören sowohl Smartphones und PDAs als auch größere, portable Geräte wie Tablet-Computer oder Notebooks. Mit Hilfe einer Software können Administratoren Benutzerkonten einrichten und mit verbundenen Geräten verknüpfen und verwalten. Des Weiteren werden auf dem Gerät liegende Daten auf diese Weise besser geschützt [Lix13].

5.3.2 Mobile Application Management

Unter Mobile Application Management (MAM) ist die Verwaltung und Administration von installierten und verfügbaren Applikationen auf einem mobilen Gerät zu verstehen. Oft gehören dazu firmeneigene Applikation, welche ausschließlich innerhalb des Unternehmens zum Einsatz kommen und nicht für den Einsatz Dritter vorgesehen ist. Im Gegensatz zum Mobile Device Management ist der Verwaltungsaufwand auf die Applikationen begrenzt.

5.3.3 Einbindung in das Betriebssystem

Anhand des Grades der Einbindung in das jeweilige Betriebssystem kann die Sicherheit der entsprechenden Software eingeschätzt werden. Umso tiefer die jeweilige Software in das Betriebssystem eingebunden ist, desto schwerer ist die Aushebelung der Sicherheitsmechanismen.

	Knox	MobileNow	BES	EMM	BYOD-Pl.
Datentrennung	-	x	x	x	-
MDM	x	x	x	x	-
MAM	x	x	x	x	x
Verschlüsselung	x	x	AES-256	x	-
OS-Einbindung	x (Samsung)	Applikation	Applikation	Applikation	x (später)
Android	x	x	x	x	x
iOS	-	x	x	x	-
BlackBerry	-	-	x	-	-
Kosten	-	Service	Service	Service	Server

Tabelle 5.2: Vergleich aller vorgestellten Lösungskozepte des BYOD-Problems.

6 Assessment

Dieses Kapitel dient der Bewertung und Beurteilung des hier vorgestellten Lösungsansatzes entsprechend der Anforderungen des Datenschutzes und der Industrie an einer 'Bring-Your-Own-Device'-Lösung.

6.1 Anforderungen

Die Anforderungen an einer Lösung für das BYOD-Problem können in unterschiedliche Klassen gruppiert werden. Zum Einen müssen rechtliche Anforderungen wie Datenschutz und Privatsphäre geklärt und entsprechend mit eingebunden werden. Ein weiterer Punkt sind die Anforderungen der Industrie an einer Lösung um sich auf die Thematik BYOD einlassen zu können. Als letztes gilt es noch die Anliegen der Benutzer und somit der Mitarbeiter zu betrachten, um diese vom Einsatz eigener Hardware im Unternehmen einschließlich der eingesetzten Sicherheitsvorkehrungen zu überzeugen.

6.1.1 Rechtliche Anforderungen

Dem Datenschutz ist beim Einsatz von privater Hardware im Unternehmen nur schwer gerecht zu werden. Um der Administration des Unternehmens die nötigen Rechte gesetzeskonform zu gewähren benötigt es der ausdrücklichen Zustimmung und Einräumung durch den Mitarbeiter.

Außerdem gilt es aus rechtlicher Sicht private und geschäftliche Daten strikt zu trennen. Da das Unternehmen zu jeder Zeit die volle Verantwortung für dienstliche und personenbezogenen Daten hat, ist es zu garantieren, dass geschäftliche Daten wie E-Mails, Dokumente und Applikationen in jeder Lage unter der Kontrolle des Unternehmens sind. Sind für bestimmte Daten diese Anforderungen nicht erfüllt, so gilt es diese aus dem BYOD-Programm zu entfernen. Die Kontrolle der Daten des Mitarbeiters ist unantastbar und gehört ausschließlich dem Benutzer.

Des Weiteren ist es nötig die Daten durch Zugangskontrollen vor unbefugtem Zugriff zu schützen. Dabei kann ein eigener Schutz des Unternehmens für jegliche Daten und Applikationen genutzt oder ein globaler Schutz des Benutzers für das Gerät verwendet werden. Als unbefugter Zugriff zählt ebenso der Lebenspartner oder Kinder des Mitarbeiters und sind somit auszuschließen. Auch weitere Mitarbeiter sind in der Regel nicht befugt

von einem Gerät eines anderen Mitarbeiters auf die IT-Infrastruktur zuzugreifen. Es können Daten zum Nachteil der anderen Mitarbeiter missbraucht werden oder die Sicherheitsstufe der Mitarbeiter könnten nicht übereinstimmen und somit die Zugriffsrechte.

Betriebssystem-Modifikationen wie Jailbreak oder Rooting können Sicherheitsvorkehrungen vieler Applikationen gestört und umgangen werden. Deshalb ist es auszuschließen, dass Mitarbeiter derartig modifizierte Geräte für den Einsatz im Unternehmen nutzen.

Auch ist es aus rechtlicher Sicht notwendig Daten vor Verlust zu schützen und entsprechende Vorkehrungen zu treffen. Aus diesem Grund sollten Daten zentral gespeichert werden und nicht auf allen Geräten vervielfältigt werden. Wird ständig mit sensiblen Daten gearbeitet ist es auch möglich, für einen begrenzten Zeitraum Daten abzurufen und anschließend beim Beenden der Applikationen geänderte Daten wieder auf den Server zu laden und lokal zu löschen. Vorkehrung gegen Verlust oder Diebstahl könnte eine Fernlöschung des gesamten Speichers auf dem Gerät sein.

Außerdem sollten Unternehmens-Richtlinien aufgestellt und von jedem Mitarbeiter angenommen werden. Es müssen klare Regeln und Vorgaben aufgestellt werden an welche sich Mitarbeiter und Arbeitgeber eindeutig zu halten haben. Hierunter fallen auch die möglichen Geräte, welche für das BYOD-Programm eingesetzt werden können. Auch Software, die unbedingt auf dem Gerät installiert werden muss (z.B. Antivirensoftware), ist hier fest zu legen. Dies ist schriftlich fest zu halten und eindeutig von jedem Mitarbeiter in Kenntnis genommen worden sein.

6.1.2 Anforderungen der Industrie

In erster Linie liegt dem Unternehmen nahe die jegliche Hardwarekosten zu senken. Dies geschieht am einfachsten wenn alle Mitarbeiter ihre privaten Geräte im Unternehmen nutzen. Allerdings entstehen dadurch anderweitig Kosten für Verwaltung, Administration und Support. Diese Kosten gleichen sich in der Regel schlussendlich aus.

Die Nebeneffekte von BYOD fallen da für das Unternehmen schon mehr ins Gewicht. In der Regel arbeiten Menschen mit privaten Geräten schneller und einfacher als mit fremden Geräten. Sie finden sich schneller zurecht und müssen in den meisten Fällen auch keine dritte Person um Rat fragen. Außerdem lässt sich durch ein privates und portables Gerät von überall und jederzeit arbeiten. Ist schließlich das Gerät ein Smartphone so ist der Mitarbeiter auch jederzeit für den Chef erreichbar.

Auch das Unternehmen wünscht sich die Kontrolle seiner Daten, wie es auch rechtlich vorausgesetzt wird. Dabei sollen Daten und Applikationen zwar nebeneinander auf dem Gerät lauffähig sein, allerdings die Herrschaft über diese je nach Typ dem rechtmäßigem Besitzer zustehen. Um die Produktivität eines Mitarbeiters zu steigern wäre auch eine ‘Pseudo-Mischung’ der Daten wünschenswert. Dabei können Daten und Applikationen zwar vermischt auftreten und arbeiten, allerdings bleiben Daten sicher vor unbefugtem Zugriff.

Die Anforderungen der Industrie sind klar, ein zufriedener Mitarbeiter, der durch den Einsatz privater Hardware seine Arbeit schneller und effizienter erledigt und jederzeit erreichbar ist. Letzteres ist allerdings auch beim Einsatz von Smartphones nicht immer zu garantieren.

6.1.3 Wünsche des Benutzers

Durch den Einsatz von BYOD im Unternehmen erhoffen sich Mitarbeiter mehr Spielraum und die freie Auswahl der Geräte, mit welchen dieser zu arbeiten hat. In dieser Hinsicht verspricht sich das Unternehmen weniger Support-Aufwand, da sich Mitarbeiter in der Regel für sich bekannte Geräte und Software entscheiden.

Durch die Nutzung privater Geräte wollen sich die wenigsten Vorschriften machen lassen wie man mit dem Gerät umzugehen hat und welche Software installiert werden soll. Allerdings ist dies in den meisten Fällen unumgänglich. Trotz des Zwangs für bestimmte Software und Regeln ist dem Nutzer trotzdem wichtig, die Kontrolle über das Gerät oder zumindest seinen privaten Daten zu haben.

Durch die nötigen Maßnahmen, die es von Unternehmensseite zu treffen gilt, möchte der Mitarbeiter aufgeklärt und zu jederzeit Einsicht haben. Niemand will gerne mit einem Gerät durch die Gegend spazieren, dass Prozeduren und Funktionen ausführt, die nicht mit dem Besitzer vereinbart wurden.

6.2 Abschließende Bewertung

Jegliche Daten werden von der BYOD-Plattform wie erfordert strikt getrennt. Durch die Applikationen der Unternehmen gelangen alle verarbeiteten Daten in den jeweiligen Container der Applikation. Durch Zusammenspiel von mehreren Applikationen des Unternehmens ist es möglich, Daten auszutauschen, sofern die nötige Berechtigung erfolgt ist.

Ein Besonderes Merkmal der BYOD-Plattform, welches keine andere Lösung bietet, ist die Möglichkeit, in bestimmten Fällen, auch Berechtigungen zu nicht firmeneigenen Applikationen zu setzen, um bei nötiger Integration problemlos zwischen Applikationen zu interagieren. Dies alles setzt allerdings das Wissen des Benutzers und anschließende Freigabe der Berechtigungen voraus. Andere bekannte Lösungen für das BYOD-Problem liefern eine komplett abgekapselte Arbeitsumgebung für die Nutzung im Unternehmen und können in keinster Weise eine solch schnelles und praktisches Verfahren nutzen.

Die Zugangskontrolle zu den Funktionen der Plattform sind durch einen globalen Code oder Passwort geschützt. Dieser ist lediglich dem Benutzer selbst bekannt und kann jederzeit in den Einstellungen des Betriebssystems angepasst werden. Jeder Mitarbeiter ist dazu verpflichtet, diesen Code keiner dritten Person mitzuteilen, da durch den Zugang zum Gerät auch alle berechtigten Funktionen von Dritten ausgeführt werden können.

6.3 Future Work

Wie jede Arbeit kann auch die hier vorgestellte Lösung stets verbessert und um Funktionen erweitert werden. Im folgenden möchte ich auf einzelne mögliche Funktionserweiterungen eingehen, die entweder nicht Teil der Aufgabenstellung waren, jedoch während der Bearbeitung als Nützlich empfunden wurden, oder aber aufgrund der begrenzten Zeit nicht zu Bearbeiten waren aber wünschenswert gewesen wären.

6.3.1 Fernwartungsfunktionen

Durch die Kontrolle eines Gerätes per Fernwartung lassen sich Sicherheitsmaßnahmen schneller einleiten und können somit Daten vor Missbrauch oder unbefugtem Zugriff geschützt werden. Allerdings wird dabei die Privatsphäre des Nutzers stark eingeschränkt. Deshalb sind nur wichtige Funktionen für die Fernwartung einzuplanen.

In erster Linie ist es für das Unternehmen wichtig, bei Verlust oder Diebstahl eines Gerätes, dieses so schnell wie möglich von jeglichen sensiblen Daten zu befreien. Im einfachsten Fall kann dies durch einen ferngesteuerten Werksreset ermöglicht werden. Dabei gehen allerdings auch persönliche Daten des Benutzers verloren.

Das Konfigurieren der BYOD-Plattform geschieht in der hier vorgestellten Version zwar über einen zentralen Konfigurationsserver, das Gerät muss dazu allerdings einmalig zu Beginn mit dem Server von Hand verbunden werden. Auch weitere Anpassungen müssen direkt am Gerät vorgenommen werden. Lediglich durch die Veränderung des Inhalts der Konfigurationsdatei kann die Konfiguration der Plattform angepasst werden. Hier könnte man sich eine komplett, ferngesteuerte Konfigurationsmöglichkeit vorstellen, in welcher nach einmaliger Einrichtung, sowohl andere Pfade der Konfigurationsdatei vergeben werden können, als auch ein Update abgehandelt werden können.

Auch Applikationen und Ressourcengruppen der Unternehmen müssen in der aktuellen Version von Hand installiert und verwaltet werden. Ein integrierter Software-Market für neue Applikationen und Ressourcengruppe könnte die Administration der Geräte um einiges erleichtern. Insbesondere wenn die Anzahl der Geräte, die es zu Verwalten gilt, im dreistelligen Bereich oder sogar noch höher liegt. Eine automatische Installation würde dem Ganzen die Krone aufsetzen. Allerdings wäre eine Benachrichtigung vor der Installation für den Benutzer von hoher Wichtigkeit.

6.3.2 Erweiterte Verschlüsselung und Zugangskontrollen

In erster Linie sind alle Daten jeder Applikation im Android System geschützt und nur von derjenigen Applikation einsehbar, die diese Daten auch speichert. Unter bestimmten Voraussetzungen und Berechtigungen können andere Applikationen und Benutzer auf diese Daten zugreifen.

Um eine höhere Sicherheit zu gewährleisten können verschiedene Verschlüsselungsmethoden zu der standardmäßigen Verschlüsselung des Android-Betriebssystems ergänzt werden.

Ebenso kann auf eine Verschlüsselungstechnik gesetzt werden, die den Kommunikationskanal zum Firmenserver schützt. Diese Maßnahme kann das Abfangen von sensiblen Informationen in unsicheren Netzen verhindern.

Wie auch in Abschnitt 6.2 angesprochen, ist keine eigene Zugangskontrolle in der aktuellen Version der BYOD-Plattform enthalten. Es wird darauf gesetzt und vorausgesetzt, dass ein Passwort das Gerät vor unberechtigtem Zugriff schützt. Diese Methode wird vom Betriebssystem angeboten und sollte auch für diesen Zweck in Anspruch genommen werden.

Durch eine individuelle Zugangskontrolle könnte jeder Zugriff durch die Plattform über einen neuen Code geschützt werden. Dabei wird die Freigabe für die Nutzung verschiedener Funktionen erst nach Eingabe des entsprechenden Codes ausgeführt. Im besten Fall löst man dieses Problem durch eine 2-Faktor-Authentifizierung, die allerdings in den meisten Fällen extra Hardware benötigt, die zu jeder Zeit einen individuellen Code generiert.

7 Zusammenfassung und Ausblick

In diesem Kapitel wird eine Zusammenfassung der Ergebnisse dieser Arbeit präsentiert und Vorschläge für die Weiterentwicklung diskutiert.

7.1 Zusammenfassung

Im Rahmen dieser Arbeit wurde ein interaktiver Ansatz für die Nutzung privater Endgeräte im Unternehmen gesucht und eingeleitet. Im Detail wurde auf Basis der Privacy Management Platform eine Anwendung zu diesem Zwecke implementiert und auf einem Android-Smartphone ausprobiert und demonstriert. Sämtliche Funktionen des Mobilgerätes werden zentral über die konzipierte Plattform geregelt und verwaltet.

Parallel dazu hat der Benutzer die Möglichkeit, individuelle Anpassungen vorzunehmen, in einem Rahmen, welcher vom bestimmenden Unternehmen gesetzt wurde. Benutzerdefinierte Richtlinien können jederzeit angepasst und verändert werden, solange Sie nicht mit den Richtlinien des Unternehmens überschneiden. Die Einschränkungen des Unternehmens werden automatisch über einen Server eingespeist und bei Veränderungen erneuert.

7.2 Ausblick

Bring-Your-Own-Device wird in der Zukunft immer mehr an Zuspruch gewinnen und dementsprechend wird die Anzahl der Arbeitnehmer steigen, welche ihr privates Gerät für Arbeitszwecke nutzen. Laut dem Institut Gartner wird dies bis zum Jahr 2017 die Hälfte aller Unternehmen betreffen. Laut einer weltweiten Umfrage von Gartner erwarten 38 Prozent der Unternehmen ab 2016 keine Mittel mehr für Geräte ausgeben zu müssen, um diese dann den Mitarbeitern auszuliefern.

Wie auch schon zu Beginn erwähnt, bestätigt David Willis (Gartner), die steigende Arbeitnehmerzufriedenheit und die neuen Möglichkeiten für die Arbeiterschaft. Ein weiterer wichtiger Faktor für die Einführung eines BYOD-Konzeptes ist die Kosteneinsparungen für Geräte der Mitarbeiter [RM13].

BYOD bringt auch Risiken mit sich, die natürlich nicht außer Acht gelassen werden dürfen. Der wichtigste Aspekt für Unternehmen muss der Sicherheitsfaktor sein. Auf mobilen Geräten ist die Gefahr von Datenlecks ziemlich hoch und nicht zu vernachlässigen. Die Sicherheit der eigenen Daten auf firmeneigenen Geräten stufen mehr als die Hälfte aller

befragten Unternehmen selbst als hoch an. Ein ähnliches Ergebnis sollte auf Geräte von Mitarbeitern als Ziel gelten.

David Willis führt weiter an, dass das BYOD-Konzept besser erklärt und verstanden werden muss um Unternehmen einen Schritt weiter zu bringen. Viele Geschäftsführer verstehen laut Willis die Vorteile des Konzeptes nicht. Des Weiteren glauben nur 22 Prozent der Unternehmen an die Weiterentwicklung, die sie durch die Einführung von BYOD erfahren haben.

Am häufigsten findet BYOD in mittleren und großen Unternehmen Einzug. Allerdings ist es auch in kleinen Unternehmen möglich einem solchen Ansatz nachzugehen. Dazu können oftmals Services von Drittanbietern in Betracht gezogen werden und somit die Anschaffungskosten gering gehalten werden.

Im Vergleich zu Europa ist es in den USA doppelt so wahrscheinlich in einem Unternehmen zu arbeiten, in dem BYOD betrieben wird. Auch in Indien, China und Brasilien kommt der Einsatz von privaten Geräten zu Arbeitszwecken öfters vor.

Literaturverzeichnis

[Ale13] D. Alexander. Samsung, BlackBerry devices cleared for use on U.S. defense networks, 2013. URL http://uk.reuters.com/article/2013/05/03/us-usa-defense-smartphones-idUKBRE94204E20130503. (Zitiert auf Seite 70)

[App13] AppTec. Bring your own Device - Damit private Geräte nach den Unternehmensrichtlinien eingesetzt werden können., 2013. URL http://www.apptec360.com/pdf/Whitepaper_EMM_2013.pdf. (Zitiert auf Seite 74)

[Bru13] C. Bruggemann. Samsung Knox bekommt Freigabe vom amerikanischen Department of Defense, 2013. URL http://allaboutsamsung.de/2013/05/samsung-knox-bekommt-freigabe-vom-amerikanischen-dod/. (Zitiert auf Seite 70)

[CCE+12] S. Chung, S. Chung, T. Escrig, Y. Bai, E.-P. Barbara. 2TAC: Distributed Access Control Architecture for Bring Your Own Device Security, 2012. URL http://www.computer.org/csdl/proceedings/biomedcom/2012/4938/00/4938a123-abs.html. (Zitiert auf Seite 66)

[Che13] B. X. Chen. Samsung Delays Release of Security Software for Galaxy Phones, 2013. URL http://bits.blogs.nytimes.com/2013/04/24/samsung-knox-delayed-july/. (Zitiert auf Seite 71)

[Deh13] S. Dehmel. Bring Your Own Device - Bitkom Leitfaden, 2013. URL http://www.bitkom.org/files/documents/20130404_LF_BYOD_2013_v2.pdf. (Zitiert auf Seite 21)

[Dix13] D. A. Dix. Datenschutz und Informationsfreiheit - Tätigkeitsbericht fuer das Jahr 2012, 2013. URL http://www.datenschutz-berlin.de/attachments/942/2012-JB-Datenschutz.pdf. (Zitiert auf Seite 13)

[Eng13] C. Engelken. Samsung Knox: Sicherheits-Software auf Juli verschoben, 2013. URL http://www.mobiflip.de/samsung-knox-sicherheits-software-auf-juli-verschoben/. (Zitiert auf Seite 71)

[Goo12a] Google. Android Storage Options, 2012. URL http://developer.android.com/guide/topics/data/data-storage.html. (Zitiert auf den Seiten 23 und 26)

[Goo12b] Google. Notes on the implementation of encryption in Android, 2012. URL http://source.android.com/tech/encryption/android_crypto_implementation.html. (Zitiert auf Seite 33)

[Goo13a] Google. Android Device Administration API Overview, 2013. URL http://developer.android.com/guide/topics/admin/device-admin.html. (Zitiert auf Seite 35)

[Goo13b] Google. Android Security Overview, 2013. URL http://source.android.com/tech/security/. (Zitiert auf Seite 31)

[Goo13c] Google. Manifest Permissions, 2013. URL http://developer.android.com/reference/android/Manifest.permission.html. (Zitiert auf Seite 36)

[Hae12] A. Haegi. AVANADE STUDIE: Arbeitsmodelle und Prozesse verändern sich durch Consumertechnologien signifikant, deutsche Unternehmen gehören zu den Vorreitern, 2012. URL http://www.avanade.com/de-de/about/avanade-news/press-releases/Documents/Avanade_release_WorkRedesigned_GER.pdf. (Zitiert auf Seite 12)

[Hal13] S. Halsbomer. BYOD - rechtlich äusserst riskant, 2013. URL http://www.computerwoche.de/a/byod-rechtlich-aeusserst-riskant,2516244. (Zitiert auf Seite 20)

[Hen] V. Hendru. Samsung Galaxy S4 becomes first Android phone to get DoD security approval, beats iPhone to that. URL http://www.phonearena.com/news/Samsung-Galaxy-S4-becomes-first-Android-phone-to-get-DoD-security-approval-beats-iPhone-to-that_id42600? (Zitiert auf Seite 70)

[Her13] S. Herget. Samsung Knox: Arbeit und Privates auf einem Gerät, 2013. URL http://www.teltarif.de/samsung-knox-cebit-android-smartphone-business/news/50261.html. (Zitiert auf Seite 69)

[Hul13a] R. Hulsenbusch. Beim Verbot privater mobiler Geräte im Job droht Widerstand, 2013. URL http://www.heise.de/newsticker/meldung/Beim-Verbot-privater-mobiler-Geraete-im-Job-droht-Widerstand-1872801.html. (Zitiert auf Seite 13)

[Hul13b] R. Hulsenbusch. BYOD: Verwaltungstool für mobile Geräte und Apps, 2013. URL http://www.heise.de/ix/meldung/BYOD-Verwaltungstool-fuer-mobile-Geraete-und-Apps-1793447.html. (Zitiert auf Seite 72)

[Kle13] E. Klein. Unternehmensdaten schützen bei '"Bring your own device'", 2013. URL http://www.personal-erfolg.de/unternehmensdaten-schuetzen-bei-bring-your-own-device/. (Zitiert auf Seite 13)

[Klo13] K. Klooss. BRING YOUR OWN DEVICE - Für Blackberry wird der Trend zum Feind, 2013. URL http://www.manager-magazin.de/unternehmen/it/0,2828,891364,00.html. (Zitiert auf Seite 73)

[Lix13] C. Lixenfeld. Mobile Device Management - Den BYOD-Wahnsinn im Griff behalten, 2013. URL http://www.cio.de/bring-your-own-device/2915752/index.html. (Zitiert auf Seite 75)

[Los13] J. Losche. BYOD: Chancen und Tücken einer Mobile-Strategie, 2013. URL http://www.tecchannel.de/kommunikation/handy_pda/2040464/byod_chancen_und_tuecken_einer_mobile_strategie/.

[Mer12] N. Meru. BYOD-Leitfaden - Anforderungen und Lösungen, 2012. URL http://documents.sysob.com/meraki_BYOD_WhitePaper.pdf. (Zitiert auf Seite 18)

[Rat11] M. Rath. ByoD - Private Hardware in der Firma nutzen, 2011. URL http://www.tecchannel.de/netzwerk/management/2033617/byopc_private_hardware_in_der_firma_nutzen/. (Zitiert auf Seite 11)

[RM13] J. Rivera, R. van der Meulen. Gartner Predicts by 2017, Half of Employers will Require Employees to Supply Their Own Device for Work Purposes, 2013. URL http://www.gartner.com/newsroom/id/2466615. (Zitiert auf Seite 83)

[Sam13] Samsung. Samsung Knox - Comprehensive mobile solution for work and pay, 2013. URL http://www.samsung.com/global/business/business-images/resource/brochure/2013/05/knox_0510_leaflet_web-0.pdf. (Zitiert auf den Seiten 8, 68 und 69)

[Saw12] A. Sawall. Jedes dritte Unternehmen setzt Tablets ein, 2012. URL http://www.golem.de/news/tablets-jedes-dritte-unternehmen-setzt-tablets-ein-1304-98835.html. (Zitiert auf Seite 13)

[Saw13] A. Sawall. "Bring your own device" in der Verwaltung verboten, 2013. URL http://www.golem.de/news/berliner-landesdatenschuetzer-bring-your-own-device-in-der-verwaltung-verboten-1303-98421.html. (Zitiert auf Seite 13)

[Sch13] H.-P. Schuler. Mobile Device Management auch für Kleinbetriebe, 2013. URL http://www.heise.de/newsticker/meldung/Mobile-Device-Management-auch-fuer-Kleinbetriebe-1858144.html. (Zitiert auf Seite 74)

[Ser13] S. Serowy. Qualitätssiegel vom Pentagon: Wird Android endlich sicher?, 2013. URL http://www.androidpit.de/qualitaetssiegel-vom-pentagon-wird-android-endlich-sicher. (Zitiert auf Seite 70)

[Sha13] A. Shaffry. ByoD - Prozessqualität rauf, Security-Risiken auch, 2013. URL http://www.computerwoche.de/a/byod-prozessqualitaet-rauf-security-risiken-auch,2532386. (Zitiert auf Seite 18)

[Sin12] N. Singh. B.Y.O.D. Genie Is Out Of the Bottle - 'Devil Or Angel'. In *Journal of Business Management and Social Sciences Research* [Sin12], S. 12. (Zitiert auf den Seiten 17 und 87)

[SM13] C. Stach, B. Mitschang. Privacy Management for Mobile Platforms - A Review of Concepts and Approaches, 2013. 1. (Zitiert auf den Seiten 8 und 43)

[Spi13] Spiegel. Jeder Dritte ist rund um die Uhr für den Chef erreichbar, 2013. URL http://www.spiegel.de/karriere/berufsleben/jeder-dritte-ist-rund-um-die-uhr-fuer-den-chef-erreichbar-a-894700.html. (Zitiert auf Seite 12)

[Sta13a] C. Stach. How to Assure Privacy on Android Phones and Devices?, 2013. 1. (Zitiert auf Seite 43)

[Sta13b] C. Stach. Wie funktioniert Datenschutz auf Mobilplattformen?, 2013. 1. (Zitiert auf den Seiten 8, 41 und 42)

[Tec12] Techconsult. Consumerization Study CIO Challenges 2012 - Herausforderungen im Umgang mit "Bring your own". Technischer Bericht, techconsult, 2012. URL http://www.techconsult.de/images/pi/ms_consumerization/Market_Paper-Herausforderungen_im_Umgang_mit_bring_your_own.pdf. (Zitiert auf den Seiten 8, 13 und 14)

[Thy13] M. Thylmann. Tablet Computer drängen in die Berufswelt, 2013. URL http://www.bitkom.org/de/markt_statistik/64054_75913.aspx. (Zitiert auf Seite 12)

[Vie12] J. Vielmeier. BRING YOUR OWN DEVICE: Consumerization motiviert Mitarbeiter und macht IT-Chefs Angst, 2012. URL http://neuerdings.com/2012/11/28/bring-your-own-device/. (Zitiert auf Seite 13)

[Was13] C. Waschkau. BYOD: Aktuelle Studien, Zahlen, Daten, Fakten, 2013. URL http://blog.net2net.de/byod/byod-aktuelle-studien-zahlen-daten-fakten. (Zitiert auf Seite 13)

[Wen13] J. Wendt. BRING YOUR OWN DEVICE - Container auf dem Smartphone, 2013. URL http://www.zeit.de/digital/mobil/2013-03/computer-arbeitsplatz-byod-container. (Zitiert auf Seite 12)

[ZO12] Z. Zhao, F. C. C. Osorio. TrustDroid: Preventing the use of SmartPhones for information leaking in corporate networks through the used of static analysis taint tracking, 2012. URL http://www.computer.org/csdl/proceedings/malware/2012/4880/00/06461017-abs.html. (Zitiert auf Seite 65)

Alle URLs wurden zuletzt am 18. 07. 2013 geprüft.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

Ort, Datum, Unterschrift