NFC-Telefon als PACE-fähiges Lesegerät für elektronische Ausweisdokumente

Diplomarbeit

Humboldt-Universität zu Berlin
Mathematisch-Naturwissenschaftliche Fakultät II
Institut für Informatik

eingereicht von: Ingo Kampe

1. Gutachter: Prof. Dr. rer. nat. Jens-Peter Redlich
2. Gutachter: Prof. Dr. rer. nat. Ernst-Günter Giessmann
Betreuer: Dr. rer. nat. Wolf Müller

Berlin, den 1. Juli 2010

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Zusammenfassung

Existierende Mobiltelefone mit einer standardisierten Near-Field-Communication Schnittstelle bieten die Möglichkeit, mit dem kontaktlos ansprechbaren neuen deutschen elektronischen Personalausweis zu kommunizieren. In der vorliegenden Arbeit wird betrachtet, welche Anforderungen an die Sicherheitsinfrastruktur und insbesondere an die Lesegeräte des neuen Personalausweises gestellt werden und inwieweit diese auf ein Mobiltelefon als Systemkomponente angewandt werden können. Der Einsatz des Telefons ermöglicht neue mobile Nutzungsszenarien und wirkt dabei direkt auf die Sicherheitseigenschaften. Die auf elliptischen Kurven basierende kryptografischen Protokolle stellen besondere Herausforderungen für die Software und auch die eingeschränkte Telefonhardware dar. Verschiedene Beispielanwendungen zeigen, was aktuell möglich ist und können dazu beitragen, das Sicherheitsniveau noch weiter anzuheben.

1 Einleitung

Schon heute ist die Anzahl der Mobilfunkverträge in Deutschland höher als die Einwohnerzahl. Neben dieser bereits sehr hohen Marktdurchdringung mit Mobiltelefonen steigt aktuell der Anteil an den noch leistungsfähigeren Smartphones rapide an. Bereits im Jahr 2010 wird fast jedes dritte neu gekaufte Telefon in Deutschland ein Smartphone sein [28]. Gleichzeitig steigt die mobile Internetnutzung und zeigt, dass die Telefone längst viel mehr Funktionen erfüllen, als nur ein Ferngespräch zu ermöglichen. Es sind tragbare Computer mit GPS-Funktion, Kamera und Browser, die außerdem als Buchersatz und Spieleplattform dienen. Die in einigen Geräten bereits vorhandene und für viele weitere Geräte angekündigte[1] integrierte Unterstützung der Near-Field-Communication (NFC) eröffnet völlig neue Möglichkeiten, das Telefon auch zum Träger unserer elektronischen Identität werden zu lassen.

Immer mehr Geschäfts- und Verwaltungsprozesse werden ins Internet verlagert. Eine elementare Vorraussetzung dafür ist der Nachweis der eigenen Identität. Bei sicherheitskritischen Anwendungen ist zur Erlangung einer authentisierten digitalen Identität immer noch das Verlassen des Mediums und die Nutzung von herkömmlichen Verfahren wie zum Beispiel *Postident* der Deutschen Post notwendig. Am 1. November 2010 wird in Deutschland ein neuer Personalausweis im klassischen Chipkartenformat eingeführt, der erstmals einen kontaktlos per NFC ansprechbaren Chip enthält. Auf Basis dieser neuen Architektur hat jeder deutsche Bürger die Möglichkeit eine Funktion zum elektronischen Identitätsnachweis (eID) freischalten zu lassen [12] und damit erstmals ein hoheitlich beglaubigtes Dokument, um seine digitalen Identität mit der realen auf sichere Art und Weise zu verbinden.

Um die eID-Funktion nutzen zu können, ist auf der Anwenderseite ein kompatibles Lesegerät notwendig. Diese Arbeit beleuchtet, welche Anwendungen bereits heute mit einem frei verfügbaren NFC-fähigen Mobiltelefon als Lesegerät und dem neuen Personalausweis möglich sind und wie sich der Einsatz auf die Infrastruktur und Sicherheit auswirkt.

1.1 Motivation

Erstmals wird es in Deutschland ein hoheitliches Dokument mit integriertem Chip geben, das anwendungsunabhängig für verschiedene Dienste und Anwendungen eine elektronische Identifikation ermöglicht. Chipkartenbasierte Identifikationsmechanismen haben bisher noch keine flächendeckende Verbreitung gefunden, weil der initiale Aufwand der neuen Infrastruktur für einzelne Unternehmen und eingeschränkte Nutzungsszenarien schlicht zu hoch war. Die Bundesrepublik ermöglicht nun, mit den Zusatzfunktionen im Ausweis und anfangs subventionierten

[1] http://www.nearfieldcommunicationsworld.com/2010/06/17/33966/all-new-nokia-smartphones-to-come-with-nfc-from-2011/

Lesegeräten das Henne-Ei-Problem aufzulösen und sowohl im eGovernment als auch in der privaten Wirtschaft Investitionen in entsprechende Anwendungen aufgrund der dann vorhandenen Anwender lukrativ zu machen. Dabei ist die Infrastruktur sowohl rechtlich als auch technisch hervorragend abgesichert, so dass die elektronische Identität ein mindestens gleichwertiger Ersatz zu existierenden Verfahren werden kann.

Wie in [3] ersichtlich, begrüßen vor allem die jüngeren und internet-affinen Nutzer den neuen Ausweis, und fast jeder zweite würde ihn auch tatsächlich zur Onlineidentifikation einsetzen. Jedoch nur 17% der Anwender würden einen Preis von über 20,- € für ein Lesegerät akzeptieren. Das einzige bisher zertifizierte Lesegerät von SCM, der Basisleser SCL011, kostet aktuell 34,90 € [2]. Da hier schon die einfachste Leserkategorie über dem Erwartungspreis liegt, lohnt sich die Suche nach einer preiswerteren Alternative.

Egal wie leicht und klein zukünftige Lesegeräte konstruiert werden - sie bleiben zusätzliche externe Erweiterungen, die besonders in einem mobilen Szenario am eigenen Laptop unangenehm sind. Zusätzlicher Transport, die Verlustgefahr und der Kabelsalat lassen also auch den Komfort zu einem Argument für eine Alternative werden.

In einem Anwendungsszenario auf Reisen kann ein elektronischer Identitätsnachweis erhebliche Vorteile zu den jetzigen Möglichkeiten bieten. Die Sprache von Anwendungen und Formularen sei hier ein Beispiel, oder auch die Online-Erledigung von authentisierungsbedürftigen Verfahren, bei denen aktuell eine Delegation an Dritte nicht oder nur sehr umständlich möglich ist. Bei einem Einsatz an einem fremden Computer mit dort installiertem Lesegerät ist das notwendige Vertrauen, um dort seinen Ausweis und auch noch die eID-PIN preiszugeben, sehr gering.

Eine hervorragende Alternative, um diese Probleme zu lösen, stellt das fast immer vorhandene eigene Mobiltelefon dar. Es ist portabel, hat alle notwendigen Hardwarekomponenten eines Lesegerätes und man ist mit der Benutzung vertraut. Bei einer kompatiblen Gestaltung und Integration kann das Telefon sowohl zu Hause als auch unterwegs eingesetzt werden und darüber hinaus auch direkt und unabhängig mit dem Ausweis kommunizieren und Operationen durchführen.

1.2 Aufgabenstellung

Ziel der Arbeit ist zu betrachten, welche Anforderungen an die Sicherheitsinfrastruktur des neuen Personalausweises und insbesondere an die Lesegeräte gestellt werden und inwieweit diese auf ein Mobiltelefon als Systemkomponente angewandt werden können. Es wird geprüft, welche Nutzungsszenarien möglich sind und wie sich diese auf die Sicherheitseigenschaften auswirken.

[2] z.B.: http://shop.de.ts.fujitsu.com/fujitsuonlineshopdeutschland/ctl500000040/cp/si500583061/cl1/

Anhand eines aktuell verfügbaren Telefons mit integriertem NFC-Chip soll untersucht werden, welche praktischen Anwendungen zum jetzigen Zeitpunkt bereits umsetzbar sind. Für die Entwicklung und den Test steht dabei eine Vorabversion des neuen Personalausweises aus dem offenen Anwendertest zur Verfügung.

1.3 Aufbau der Arbeit

Im zweiten Kapitel werden die Grundlagen und Begriffe zu Sicherheit allgemein, der elektronischen Identität und Chipkarten eingeführt.

Im dritten Kapitel wird der neue Personalausweis und die Komponenten der dazugehörigen Sicherheitsinfrastruktur beschrieben. Hierbei werden die Verfahren und Protokolle skizziert und Sicherheitsziele des Designs dargestellt. Auch die Anforderungen an Lesegeräte werden zusammengefasst.

Im vierten Kapitel werden mögliche Einsatzszenarien des Mobiltelefons als Systemkomponente in der Anwendung des Personalausweises beschrieben und die Auswirkungen auf die Sicherheitsziele analysiert und bewertet. Die Anforderungen an Lesegeräte werden auf das Mobiltelefon übertragen.

Im fünften Kapitel werden drei Beispielanwendungen entworfen und die Umsetzung sowie die praktischen Erfahrungen damit ausgewertet.

1.4 Abkürzungen

Im Abkürzungsverzeichnis auf Seite 58 werden die in dieser Arbeit verwendeten Abkürzungen erläutert. Der größte Teil geht auf entsprechende Definitionen in den verschiedenen Technischen Richtlinien des Bundesamtes für Sicherheit in der Informationstechnik (BSI) zurück [5, 11, 9, 8, 6, 7, 10].

2 Grundlagen

2.1 Authentizität und Identität

Die *Authentizität* wird in [13] als erstes Schutzziel in einem sicheren System definiert. Dabei geht es um „die Echtheit und Glaubwürdigkeit des Objekts bzw. Subjekts, die anhand einer eindeutigen Identität und charakteristischen Eigenschaften überprüfbar ist“ [13, S. 6f]. Ein Subjekt ist dabei nicht immer direkt der aktive Benutzer, sondern umfasst ebenso Prozesse in dessen Auftrag. Der Vorgang des Authentizitätsbeweises ist die *Authentifikation*. Dieser ist wesentlicher Bestandteil und Grundvoraussetzung für viele digitalisierte und netzwerkbasierte Prozesse. Die *Identität* ist hierbei die Menge aller Eigenschaften und Attribute einer Person, mit dem Ziel, diese möglichst eineindeutig zu wählen, so dass ein Subjekt anhand der Menge der Merkmale zuordbar und wiedererkennbar oder anders ausgedrückt *identifizierbar* wird. Üblicherweise reichen für eine Eindeutigkeit in Deutschland Vorname und Nachname, Meldeadresse und Geburtsdatum aus. Neben diesen allgemeinen Merkmalen gehören auch domänenspezifische Attribute (z.B. Rollen, Kontonummer, Steuernummer) und biometrische Eigenschaften (z.B. Aussehen, Fingerabdruck, Unterschrift) zur Identität eines Subjekts.

Zur Authentifikation dienen heute verschiedene Mittel, die man in drei Kategorien einordnen kann.

- Besitz (z.B. klassisch Schlüssel zu einem Schloss),
- Wissen (z.B. Passwort),
- biometrischer Merkmale (z.B. Unterschrift).

In der Praxis wird heutzutage meist eine Kombination aus diesen Möglichkeiten angewandt und man bezeichnet dies dann als n-Faktor-Authentifikation, abhängig von der Anzahl n der Faktoren. Sehr stark verbreitet ist aktuell die Nutzung der 2-Faktor-Methode aus Chipkarte (Besitz) und PIN (Wissen), wie zum Beispiel bei den Bankkarten. Im deutschen Sprachraum gibt es noch den zusätzlichen Begriff der Authentisierung, mit dem die Durchführung des Vorgangs der Authentifikation aus Sicht des sich authentifizierenden Subjekts bezeichnet wird. Dieser Blickwinkel auf den Vorgang führt dazu, dass die Begriffe oft synonym verwendet werden. Der Benutzer authentisiert sich an einem Dienst, und der Dienst authentifiziert dabei den Nutzer.

„Besitzt ein Subjekt die Berechtigung zum Zugriff auf eine Information bzw. auf ein Datenobjekt, so sagen wir, dass das Subjekt zu diesem Zugriff *autorisiert* ist“ [13, S. 4]. Bei der *Autorisierung* geht es um die Rechteverwaltung und Zugriffskontrolle eines IT Systems. Wichtige Voraussetzung dafür stellt die vorausgehende erfolgreiche Authentifizierung aller Subjekte und Objekte dar. Erst danach können die entsprechenden Zugriffe kontrolliert werden. Es geht

darum zu prüfen, ob ein Subjekt S auf einem Objekt O eine Aktion A ausführen darf. Die Entscheidung über die Zulässigkeit eines derartigen Zugriffs (S, O, A) kann in verschiedenen Sicherheitsmodellen auf unterschiedliche Art und Weise getroffen werden.

2.2 Chipkarten

Als Grundlage für die neuen Anwendungsmöglichkeiten der elektronischen Ausweise dient die fortgeschrittene Entwicklung der Chipkarten. Die neuen Ausweise sind mit ihren integrierten Chips sogar als Chipkarte zu klassifizieren. Das „Handbuch der Chipkarten“ [31] stellt hierbei ein hervorragendes detailliertes Nachschlagewerk zur Technik und Historie dar.

Als Kartenkörper dienen Polycarbonatkarten mit unterschiedlichen Formaten und Größen. Das klassische Scheckkarten-Format ID-1 ist standardisiert in ISO-7816 [24] und identisch zur Größenspezifikation td-1 für elektronische Reisedokumente in [20, Kapitel V]. Die Polycarbonatkarte stellt einen robusten und wasserbeständigen Ersatz für papierbasierte Karten dar. Zunächst nur als Träger optischer ablesbarer Informationen genutzt, wurden diese dann erweitert um den von Bankkarten bekannten Magnetstreifen. Diese haben bereits eine Speicherkapazität von ca. 1024 Bits. Größter Nachteil der Magnetstreifen ist der fehlende Schutz vor Manipulation der gespeicherten Daten. Die nächste Evolutionsstufe folgte mit der Integration eines Mikrochips in die Karte. Je nach Ausprägung unterscheidet man hierbei die Speicherkarten und Prozessorkarten, welche zusätzlich zu einem Speicher einen kompletten Mikrocontroller inklusive CPU und Betriebssystem beinhalten. Diese Prozessorkarten werden auch *Smartcards* genannt. Erst durch die Möglichkeit der Ausführung von Programmen auf dem Chip ist es möglich, kryptografische Operationen und eine Zugriffskontrolle auf die gespeicherten Daten umzusetzen.

Eine weitere Klassifikation der Chipkarten wird über die Art der Kommunikation spezifiziert. Eine SIM-Karte im Mobiltelefon ist ein Vertreter der *kontaktbehafteten* Chipkarten, bei denen das sichtbare Metallkontaktelement über eine galvanische Verbindung mit Strom versorgt und darüber auch Daten ausgetauscht werden. Kontaktbehaftete Karten sind aktuell noch leistungsfähiger und kostengünstiger als kontaktlose.

Kontaktlose Karten kommunizieren über Funk mit einem Terminal. Insbesondere Karten gemäß ISO-14443 [23] mit einer spezifizierten Funkreichweite von bis zu 10 cm eignen sich auf Grund der geringen und damit kontrollierbaren Reichweite gut für Ausweisdokumente. Der Standard ISO/IEC 14443, ist an ISO/IEC 7816 angelehnt und sieht dieselben Schnittstellen vor. Dadurch kann eine Anwendung mit entsprechenden Treibern auf einer identischen Sicht arbeiten, unerheblich, ob die verbundene Chipkarte kontaktlos oder kontaktbehaftet ist. Im Falle einer passiven RFID-basierten Karte (Radio Frequency Identification) wird die Stromversorgung ebenfalls kontaktlos über elektrische Induktion durchgeführt. Da Karte und Terminal in diesem Fall keinen direkten Kontakt benötigen, kann die Karte komplett wasser- und schmutzisoliert

werden und verschleißarm auch sehr häufig eingesetzt werden. Für elektronische Ausweisdokumente entwickelt sich dies zum Standard.

2.3 Near-Field-Communication

Near-Field-Communication (NFC) ist ein zu RFID kompatibler Funkstandard für den Nahbereich, der zusätzlich auch eine direkte Kommunikation von zwei aktiven Geräten ermöglicht (in RFID nur Leser↔Tag). Während RFID mit sehr unterschiedlichen Reichweiten arbeiten kann, ist NFC auf einen Bereich von unter 10cm ausgelegt und soll hierbei vor allem die optische und die direkte (durch galvanische Kopplung) Datenübertragung ersetzen. Bezeichnend für diesen Nahbereich wird auch von „Proximity-Coupling“ Karten und Geräten gesprochen. Es wird immer nur eine exklusive Punkt-zu-Punkt Verbindung aufgebaut und NFC arbeitet dabei im Frequenzbereich bei 13,56 MHz und erreicht eine maximale Datenübertragungsrate von 424 kBit/s.

Vor allem für die Anwendung in mobilen Geräten gedacht, ist NFC aber auch kompatibel zu ISO-14443 kompatiblen kontaktlosen Smartcards. Es bieten sich daher sehr vielseitige Anwendungsmöglichkeiten dieser Kommunikationstechnologie an, und die Marktprognosen und die Vielzahl an Pilotversuchen auf der ganzen Welt lassen die Vermutung zu, dass es sich hier um ein zukünftig sehr weit verbreitetes Standardverfahren handeln wird. Aktuelle Einsatzgebiete sind

e-Ticketing zum Beispiel von der Deutschen Bahn im Touch&Travel Pilotversuch[3],

Mobile Payment in Japan bereits als etablierter Zahlungskanal mit der Technologie "Osaifu-Keitai" von NTT domoco [4],

Elektronischen Ausweisdokumenten zum Beispiel im deutschen Reisepass.

Es ist zu erwarten, dass die verschiedenen Prognosen zu einer höheren Verbreitung von NFC [5] tatsächlich in naher Zukunft eintreffen und im Wechselspiel mit den nun aufkommenden elektronischen Ausweisen weitere Anwendungen zur Nutzung entstehen. Im Juni 2010 hat Nokia angekündigt, dass alle von ihnen neu aufgelegten Smartphones ab 2011 mit NFC ausgerüstet sein werden [6].

[3] http://www.touchandtravel.de/

[4] http://www.nttdocomo.co.jp/english/service/convenience/osaifu/index.html

[5] http://www.nearfieldcommunicationsworld.com/2010/06/04/33823/one-in-five-mobile-phones-to-feature-nfc-by-2012/

[6] http://www.nearfieldcommunicationsworld.com/2010/06/17/33966/all-new-nokia-smartphones-to-come-with-nfc-from-2011/

2.4 Lesegeräte

Chipkarten benötigen zur Kommunikation mit einer Anwendung ein passendes Lesegerät. Hierbei gibt es Leser mit kontaktbehafteten und mit kontaktlosen Schnittstellen. Die Aufgaben der Lesegeräte reichen von einfachen Chipkartenlesern zum reinen Abwickeln der Kommunikation und Stromversorgung des Chips bis zu komplexen Kartenterminals mit eigener Logik und zusätzlichen aktiven Sicherheitsfunktionen.

Abbildung 1: Lesegerät SCL011

Dadurch bedingt, dass Chipkarten aktuell meist nur begrenzt auf einer Anwendungsdomäne (z.B. Geldkarten) eingesetzt werden, sind auch die Lesegeräte meist anwendungsspezifisch und zueinander inkompatibel. Mit der Technischen Richtlinie 3119 [6] versucht das BSI nun im Rahmen der Einführung des neuen Personalausweises einen allgemeingültigen Standard zu beschreiben, mit dem kompatible Chipkartenleser produziert werden können, die auch über die eCard-Strategie der Bundesregierung hinaus in der privaten Wirtschaft mit unterschiedlichen Chipkarten genutzt werden sollen. Der Standard basiert auf den internationalen Richtlinien der International Civil Aviation Organization (ICAO) und garantiert damit auch über das deutsche Einsatzgebiet hinaus kompatible Lesegeräte, die mit ausländischen maschinenlesbaren Ausweisen umgehen können [20, 21].

Das SCL011 von SCM Microsystems ist das erste nach TR-03119 zertifizierte Lesegerät und arbeitet nach ISO/IEC-14443 mit kontaktlosen Smartcards und auch dem neuen Personalausweis zusammen. Es ist ein sogenannter *Basisleser* (Cat-B). Darüberhinaus definiert das BSI noch zwei weitere Arten, den *Standardleser* (Cat-S) und den *Komfortleser* (Cat-K). Für jede dieser Lesegerätfamilien werden bestimmte Module als Mindestanforderungen spezifiziert. Eine Übersicht dieser Anforderungen findet sich in Tabelle 2.4 auf Seite 8.

	Cat-B	Cat-S	Cat-K
Umweltanforderungen	x	x	x
Funktionale Prüfung	x	x	x
kontaktlose Schnittstelle ISO/IEC 14443	x	x	x
kontaktbehaftete Schnittstellle ISO/IEC 7816	o	o	x
Pinpad (sichere PIN Eingabe)	o	x	x
PACE	o	x	x
nPA-QES	o	o	x
Display (2x16 alphanum. Zeichen)	o	o	x
Firmwareupdate	o	x	x
Applikation im Leser	o	o	x

Tabelle 1: Chipkartenleserkategorien nach [6, Tabelle 1]

2.5 Elektronische Ausweisdokumente

Von *elektronischen Ausweisen* spricht man im Fall eines in den Ausweis eingebrachten Mikrochips, vergleichbar zu den Prozessorchipkarten (s. Abschnitt 2.2 auf Seite 5). Wie in [33] beschrieben, gibt es derzeit einen Boom der elektronischen Ausweise mit allein 10 deutschen Großprojekten. Dies ist vor allem zurückzuführen auf die Vorteile der Kombination der Mikrochipfunktionalität mit den althergebrachten Aufgaben, und darüber hinaus die völlig neue Anwendungsmöglichkeit dieser Ausweise in Onlineverfahren. Der Mikrochip kann die Zugangskontrolle zu den Identitätsdaten durchführen und durch eine redundante Speicherung der äußerlich optisch aufgebrachten Informationen auch zur Fälschungssicherheit beitragen. Elektronische Ausweise können verschiedene Funktionen in einem Chip vereinen und bieten bei entsprechend hoher Verbreitung sowohl dem Anwender als auch der Behörde oder dem Unternehmen erhebliche Vorteile. Es lassen sich in den verschiedensten Nutzungsszenarien der neuen digitalisierten Welt in Zukunft Medienbrüche durch fehlende elektronische Authentisierung oder optional einer Signatur vermeiden. Als ein Beispiel sei hier die Authentifikation eines Kreditnehmers gegenüber einem Bankinstitut angeführt. Bei fast allen Banken kann der Prozess der Kreditvergabe komplett digital im Internet abgewickelt werden, von der Kundeninformation, über die Angebotsanfrage, -zusage und den Vertragsaustausch. Für zwei Schritte wird der Prozess unterbrochen:

- die Authentisierung des Kunden, die heute meistens mit dem sehr geläufigen Postidentverfahren durchgeführt wird, um mit dem biometrischen Merkmal Unterschrift vor einem vertrauenswürdigen Dritten (dem/der Postangestellen) und der optischen Prüfung des herkömmlichen Personalausweises die Authentizität und die Korrektheit der ladungsfähigen

Meldeadresse nachzuweisen

- die persönliche Unterschrift des Kreditantrages, der dann meist gemeinsam mit den Postidentunterlagen in Papierform an die Bank verschickt wird.

Beide dieser skizzierten Anforderungen werden sich in Zukunft direkt Online mit Hilfe der neuen elektronischen Ausweise erledigen lassen, wobei die elektronische Signatur nur eine Zusatzfunktion ist, die auch heute schon mit entsprechenden Signaturkarten nutzbar ist. Weitere Vorteile sind die höheren speicherbaren Datenmengen und - bei biometrischen Funktionen - ein Schutz vor Fremdnutzung an Automaten.

Bei den Nachteilen stehen der höhere Preis und vor allem die möglichen datenschutztechnischen Auswirkungen. Digitale Daten sind immer der Gefahr des unbemerkten Kopierens, Speicherns und der missbräuchlichen Weiterverarbeitung ausgesetzt, was in den Händen des Staates viel Misstrauen hervorruft. Um so wichtiger ist es, für die verschiedenen Projekte sorgfältige Schutzmaßnahmen und starke kryptografische Absicherungen der kompletten Infrastruktur zu gewährleisten, denn ein technischer Fehler in den Sicherungsmaßnahmen würde zum Vertrauensverlust in die hoheitlichen elektronischen Ausweise führen.

Es gibt verschiedene Standardisierungsgrundlagen für elektronische Ausweisdokumente, die ausführlich in [33, Teil III] dargestellt werden. Der Älteste und mittlerweile international anerkannt und angewandte ist der ICAO Standard „Machine Readable Travel Documents“ (MRTD) mit Band 3 für Ausweise, bestehend aus zwei Teilen [20, 21]. In der Luftfahrtindustrie war der Bedarf nach weltweit kompatiblen Ausweisdokumenten als erstes vorhanden und führte zu einer Vereinheitlichung der vorhandenen Systeme und einer Festlegung der Kartenkörper sowie der Chipfunktionalität. Basierend auf den Grundlagen dieses Standard sind alle weiteren Standards in der EU und auch in Deutschland entstanden, die vor allem neue Sicherheitsfunktionen und Anwendungsbereiche ergänzen. In der EU gibt es das Standardisierungsvorhaben namens European Citizen Card (EuCC). Parallel zu diesem Versuch, die vorhandenen und aktuell in Planung befindlichen europäischen Vorhaben zu vereinen, hat das BSI unter Berücksichtigung von EuCC und ICAO die Technischen Richtlinien für den neuen Personalausweis veröffentlicht (vor allem [9]). Einige andere Länder (z.B. Finnland, Belgien) haben sogar schon elektronische Ausweise in der Praxis im Einsatz, so dass die sehr unterschiedlichen Entwicklungsstände einen einheitlichen interoperablen Standard der EuCC sehr schwierig machen.

2.6 Kryptografische Verfahren

Kryptografie und elektronische Ausweise stehen im starken wechselseitigen Einfluss. Zum einen können kryptografische Verfahren durch den Ausweis sicherer gemacht werden, und zum anderen bedürfen sichere Ausweise der Kryptografie. Ausweise können als sicheres „Token“ Attri-

bute und Rechte unverfälschbar und zugriffsbeschränkt transportieren oder auch als Ganzes als Besitzfaktor innerhalb eines Authentisierungsprozesses (s. Abschnitt 2.1 auf Seite 4) angewendet werden. Um die Zugriffsbeschränkung auf die gespeicherten Daten des Ausweises zu ermöglichen, sind bereits verschiedene kryptografische Grundfunktionaliäten notwendig, die hier kurz zusammengefasst werden.

Mit der Technischen Richtlinie 2102 [5] bietet das BSI einen Leitfaden zu grundlegenden kryptografischen Verfahren für Entwickler von neuen sicheren Systemen. Als allgemeine Hinweise wird zur Datensparsamkeit geraten, zur Aktualisierbarkeit der beteiligten Systeme auf größere Schlüssellängen und neue Algorithmen und zum Einsatz möglichst starker Verfahren, da die Vorhersage der Kryptanalyseeffizienz über einen Zeitraum von 6 Jahren hinaus sehr schwierig ist. Dieser Zeitraum steht im Kontrast zur 10 Jahre dauernden Gültigkeit des Personalausweises, der daher nur abgesichert sein kann, wenn bereits zum Zeitpunkt der Umsetzung Schlüssellängen auf einem zukünftigen kalkulierten Sicherheitsniveau verwendet werden. Für alle Verfahren wird angestrebt, dass die Sicherheit allein in der Kenntnis des geheimen Schlüsselmaterials begründet wird und nicht in dem Wissen über den Algorithmus. Zumindest außerhalb der militärischen Anwendungsbereiche hat es sich etabliert, neue Verfahren und Algorithmen zu veröffentlichen oder direkt in offenen Wettbewerben zu gewinnen, um Fehler durch eine globale Prüfung von vornherein zu minimieren.

Schutzziele

Kryptografie wird in [13, S. 279] als „Lehre von den Methoden zur Ver- und Entschlüsselung von Nachrichten zum Zweck der Geheimhaltung von Informationen gegenüber Dritten“ definiert. Tatsächlich gehen die Schutzziele von kryptografischen Verfahren aber über die hier beschriebene *Vertraulichkeit*, als eine Verhinderung der unautorisierten Informationsgewinnung, hinaus. Als einfaches Beispiel zur Veranschaulichung soll der Versand eines Briefes dienen. Die Vertraulichkeit eines geheimen Briefes stellt man her über einen Briefumschlag, der das Lesen des Textes verhindert.

Weitere klassische Schutzziele sind die *Integrität* und *Verfügbarkeit*. „Die Datenintegrität eines Systems ist gewährleistet, wenn es Subjekten nicht möglich ist, die zu schützenden Daten unautorisiert und unbemerkt zu manipulieren.“ Man kann die Integrität also auch mit Nichtveränderbarkeit beschreiben. Dieses Ziel kann man erreichen (zugegeben etwas naiv) mit dem Zukleben und Versiegeln des Briefumschlags.

Unter der *Verfügbarkeit* definiert C. Eckert die Gewährleistung eines Systems, dass niemand unautorisiert die Wahrnehmung autorisierter Aktionen beeinträchtigen kann. Wenn man berechtigt ist, einen Brief zu schreiben und zu verschicken, darf niemand dafür sorgen, dass der Brief beim Transport vernichtet wird. Ein wirklich schwer zu erreichendes Ziel im klassischen Brief-

versand. Man überbringe den Brief also persönlich.

Über diese drei klassischen Schutzziele hinaus haben sich in den letzten Jahren die Systeme und Anforderungen weiter verfeinert und die Schutzziele wurden dementsprechend auch erweitert. Eine Systematisierung dazu findet sind in [15].

Für diese Arbeit relevant sind weiterhin die bereits in Abschnitt 2.1 auf Seite 4 beschriebene *Authentizität*. Zum Beweis der Authentizität des Beispielschreibens kann man einen Fingerabdruck auf das Briefpapier setzen

Zu guter Letzt geht es mit dem Erreichen der *Verbindlichkeit* von Aktionen darum, dass ein Subjekt im Nachhinein die Durchführung einer Aktion nicht abstreiten kann. Daher setzt man Ort, Datum und Unterschrift handschriftlich unter einen Brief, so dass, so lange das Dokument existiert, ein Handschriftensachverständiger immer den Urheber mittels Schriftvergleich identifizieren bzw. einen falschen Verdächtigen ausschließen kann. Damit wäre nachvollziehbar, wer der Verfasser des Briefes war, aber ob das verzeichnete Datum der Realität entspricht, ist so nicht gesichert. Dies könnte man wiederum mit kryptografisch gesicherten Zeitstempeldiensten erreichen, spielt aber hier keine Rolle.

Besonders mit Bezug zu hoheitlichen Ausweisdokumenten spielen noch weitere Schutzziele eine Rolle. Hierzu gehört die *Originalität*, mit der sichergestellt wird, dass ein vorliegender Ausweis nicht kopiert oder geklont wurde.

Aus dem Bereich des Datenschutzes im Sinne von Privatsphäre („Privacy“) sind das die Ziele der *Anonymisierung* oder *Pseudonymisierung*. Hierbei geht es darum, dass personenbezogene Daten einer natürlichen Person entweder gar nicht mehr (anonym) oder mit Hilfe einer Zuordnungsvorschrift nur noch einem Pseudonym zugeordnet werden können. Die Anonymisierung und Verbindlichkeit sind zueinander diametral, da bei einer nicht mehr ermittelbaren natürlichen Person schwer die Durchführung einer bestimmten Aktion durch diese Person nicht abstreitbar nachgewiesen werden kann.

Für jedes System müssen entsprechende Ziele neu festgelegt, kombiniert und mittels kryptografischer Verfahren abgesichert werden.

Symmetrische Verschlüsselung

Die symmetrische Verschlüsselung ist das älteste Verschlüsselungsverfahren und bezeichnet eine Funktion, bei der sowohl zur Verschlüsselung als auch Entschlüsselung der gleiche Schlüssel genutzt wird. Mit dem Schlüssel s als Parameter bildet diese Funktion einen Klartext k auf den Chiffriertext c ab, und die inverse Funktion ergibt wieder den Klartext.

Verschlüsselung $F_s(k) = c$

Entschlüsselung $F_s^{-1}(c) = k$

Bedingung $\forall s \in S, k \in K : F_s^{-1}(F_s(k)) = k$

Beispiele für F sind AES-128, AES-256, SERPENT-128 oder Twofish-128. Als Funktion werden nur noch Blockchiffren (Alternative zu Stromchiffren) empfohlen mit der zusätzlichen Eigenschaft, dass die Länge des Klartextes einem Vielfachen der Blocklänge und der Länge des Chiffriertextes entspricht. Um diese Länge zu erreichen, gibt es verschiedene *Padding*-Verfahren, die den letzten Klartextblock auffüllen. Im geläufigen Fall, dass es mehrere Blöcke gibt, muss noch über entsprechende Betriebsarten festgelegt werden, wie diese Blöcke verarbeitet werden. Hierfür legt ISO 10116 vier Verfahren fest, von denen *Electronic Code Book* (ECB) und *Cipher Block Chaining* (CBC) in der Welt der elektronischen Ausweise zum Einsatz kommen. Um eine symmetrische Verschlüsselung also vollständig zu beschreiben, benötigt man ein 3-Tupel aus $(cipher, mode, padding)$ und ist so auch im Anwendungsquelltext aus Abschnitt 5 zu finden, unter anderem mit *„AES/CBC/NoPadding“*.

Die symmetrische Verschlüsselung gewährleistet die Vertraulichkeit.

Asymmetrische Verschlüsselung

Die symmetrische Verschlüsselung angewandt in Kommunikationsbeziehungen hat das Problem des vorher notwendigen Schlüsselaustausches. Um dieses Problem zu umgehen, wurden die asymmetrischen Verfahren entwickelt. Hierbei besitzt jeder Kommunikationsteilnehmer ein Schlüsselpaar aus zusammengehörigen privatem Schlüssel SK und öffentlichem PK. Mit Hilfe sogenannter mathematischer Einwegfunktionen mit Falltür muss gewährleistet werden, dass der Klartext aus dem Chiffriertext und der SK aus dem PK nicht oder nur mit sehr hohem Aufwand berechnet werden kann. Erst mit einer geheimen Zusatzinformation ist die Umkehrung der Funktion zu berechnen. Zu diesen Einwegfunktionen gehören die Primfaktorzerlegung und das diskrete Logarithmusproblem.

Mit dem Schlüssel PK als Parameter bildet die Verschlüsselungsfunktion einen Klartext k auf den Chiffriertext c ab und die Entschlüsselungsfunktion mit dem Schlüssel SK den Chiffriertext c auf den Klartext k.

Verschlüsselung $F_{PK}(k) = c$

Entschlüsselung $F_{SK}(c) = k$

Der große Vorteil ist, dass jeder Teilnehmer nur seinen eigenen SK geheim halten muss und nicht wie im symmetrischen Fall für jeden Kommunikationspartner einen. Auch wird bei einer eindeutigen Zuordnung von Schlüsseln zu Subjekten über Zertifikate und deren Veröffentlichung (z.B. im Global Trustpoint [7]) eine verschlüsselte Adhoc-Kommunikation ermöglicht ohne

[7] https://www.globaltrustpoint.com/

direkten vorhergehenden Schlüsselaustausch. Ein bekanntes asymmetrische Verfahren ist RSA, basierend auf dem Faktorisierungsproblem ganzer Zahlen. Weitere Details dazu finden sich in [5]. Nachteilig ist die vergleichsweise geringe Verarbeitungsgeschwindigkeit und die wesentlich größeren Schlüssellängen. Zur Erreichung eines vergleichbaren Sicherheitsniveaus von 100 Bit symmetrischer Schlüssel bedarf es laut BSI im RSA-Verfahren 2048 Bit.

Die symmetrische Verschlüsselung gewährleistet die Vertraulichkeit, aber keine Integrität oder Verbindlichkeit.

Hashfunktionen

Hashfunktionen werden verwendet, um eine lange Information auf eine kürzere mit vordefinierter fester Länge abzubilden. Vergleichbar mit einem Fingerabdruck als Verkürzung und Zusammenfassung der kompletten biometrischen Eigenschaften einer Person. Für kryptografische Anwendungen werden drei Eigenschaften von der verwendeten mathematischen Einwegfunktion erwartet.

Hashfunktion $H : \{0,1\}^* \rightarrow \{0,1\}^n$ mit fester Länge $n \in \mathbb{N}$

Einweg-Eigenschaft für $h \in \{0,1\}^n$ ist es praktisch unmöglich, einen Wert $m \in \{0,1\}^*$ mit $H(m) = h$ zu finden

2.-Urbild-Eigenschaft für $m \in \{0,1\}^*$ ist es praktisch unmöglich, einen Wert $m' \in \{0,1\}^* \setminus \{m\}$ mit $H(m) = H(m')$ zu finden

Kollisionsresistenz Es ist praktisch unmöglich, zwei Werte $m, m' \in \{0,1\}^*$so zu finden, dass $m \neq m'$ und $H(m) = H(m')$ gilt

Das BSI empfiehlt Hashlängen ab 200 Bit und schränkt die empfohlenen Verfahren auf SHA-224, SHA-256. SHA-384, SHA-512 ein.

Hashfunktionen allein erfüllen keines der definierten Schutzziele, sind aber wichtiger Bestandteil bei Schlüsselableitungen und der digitalen Signatur.

Nachrichtenauthentifizierung

Der *Message Authentication Code* (MAC) ist eine Prüfsumme zu einem Nachrichtentext, in die ein geheimer Schlüssel als Parameter mit einfließt. Die Prüfung auf der Gegenseite erfolgt durch die Wiederholung des MAC-Algorithmus und den Vergleich der Ergebnisse. Bei einer Manipulation der Daten sind die Prüfsummen unterschiedlich.

MAC-Funktion $F_s(k) = m$

MAC-Funktionen basieren auf symmetrischen Blockchiffren (CMAC) oder Hashfunktionen (HMAC) zur Berechnung der Prüfsumme.

Die MAC-Funktion gewährleistet die Integrität und Authentizität einer Nachricht, aber keine Vertraulichkeit oder Verbindlichkeit.

Zertifikate

Digitale Zertifikate verbinden zuordbar und verifizierbar einen Schlüssel und Attribute mit einer Identität. Mit Hilfe einer Vertrauenshierarchie zu einer zertifikatsaustellenden Instanz kann diese Zuordnung, welche durch eine digitale Signatur innerhalb des Zertifikats vom Aussteller verbindlich festgelegt wird, überprüft werden. Für Zertifikate gibt es verschiedene, auch standardisierte Formate (z.B. *X.509* oder *Card Verifiable Certificates*), auf die hier nicht näher eingegangen werden soll.

Digitale Signatur

Die digitale Signatur ist ein kryptografisches Verfahren zur Erstellung eines Signaturwerts aus den zu signierenden Daten, der eindeutig und verifizierbar verknüpft ist mit einer Identität. Hierfür gibt es verschiedene Verfahren wie zum Beispiel RSA oder DSA (*Digital Signature Algorithm*).

Beim RSA-Verfahren basiert die digitale Signatur auf der asymmetrischen Verschlüsselung kombiniert mit Hashfunktionen. Auf einem beliebigen Eingabetext wird ein Hashwert berechnet, der mit dem geheimen Schlüssel eines signierenden Subjekts verschlüsselt wird. Eine signierte Nachricht besteht also aus dem originalen Text, dem Signaturwert und optional dem öffentlichen Zertifikat der unterzeichnenden Person. Bei der Prüfung der digitalen Unterschrift wird mit der gleichen Hashfunktion auf dem Text eine Prüfsumme gebildet, und diese mit dem Ergebnis aus der mit dem passenden öffentlichen Schlüssel entschlüsselten originären Prüfsumme verglichen. Wenn beide übereinstimmen, ist die Signatur gültig und das Dokument wurde nicht verändert.

Die digitale Signatur gewährleistet die Verbindlichkeit und Authentizität über die ausschließliche Bindung des privaten Schlüssels an den Absender und die Integrität der signierten Daten.

Kryptografische Protokolle

Um reale Anforderungen an die Sicherheit von Systemen zu erfüllen, müssen oft verschiedene Schritte aus den dargestellten Grundfunktionen zu einem Ablauf, dem kryptografischen Protokoll, zusammengefasst werden. Kryptografische Protokolle gibt es in einer großen Vielfalt, und sie unterliegen oft sehr dynamischen Entwicklungsprozessen. Es gibt einige etablierte Standardverfahren, wie z.B. den Diffie-Hellmann-Schlüsselaustausch, die dann immer wieder angepasst

und ergänzt werden, um gefundene Schwachstellen zu vermeiden oder neue Anforderungen zu erfüllen. Das Design derartiger Protokolle ist sehr anspruchsvoll, da bei falscher Kombination auch die eigentlich sicheren Grundfunktionalitäten zu einem unsicheren Protokoll führen können. In der theoretischen Betrachtung und Analyse von Protokollen versucht man daher, bestimmte Klassen zu definieren, für die bereits Eigenschaften bewiesen sind, und neue Protokolle in diese Klassen einzuordnen.

Eine dieser Klassen sind *Zero-Knowledge-Protokolle* , in denen zwei Kommunikationsteilnehmer den Besitz eines Geheimnisses beweisen, ohne dieses Geheimnis preiszugeben und ohne einem Beobachter Rückschlüsse auf das Geheimnis oder andere Informationen preiszugeben. Darüberhinaus gibt es weitere Schutzziele, die speziell auf Kommunikationsprotokolle zutreffen.

Folgenlosigkeit (engl. Forward-secrecy) beschreibt, dass das Aufdecken eines Schlüssels innerhalb einer Kommunikation keine weiteren Rückschlüsse auf vorhergehende oder nachfolgende Schlüssel zulässt

Nicht-Nachverfolgbarkeit (engl. Untraceability) meint, dass die Teilnehmer durch einen Protokolllauf nicht wiedererkannt und identifiziert werden können und somit auch keine Kommunikationsprofile entstehen

Nicht-Verknüpfbarkeit (engl. Unlinkability) meint, dass Protokollläufe nicht mit einander verknüpft werden können, also keine charakteristischen Informationen über einen Lauf gewonnen werden können

Nicht-Übertragbarkeit (engl. Non-Transferability) meint, dass ein Teilnehmer nach Durchführung des Protokolls nicht in der Lage ist, gegenüber Dritten zu beweisen, dass der Protokolllauf stattgefunden hat

Alle dieser Schutzziele werden zum Beispiel vom PACE-Protokoll (Abschnitt 3.3 auf Seite 21) des neuen Personalausweises erfüllt.

Elliptische-Kurven-Kryptographie

Die im neuen Personalausweis verwendete Elliptische-Kurven-Kryptographie (ECC) wird in [8] eingeführt. Jedes asymmetrische kryptografische Verfahren kann auf Berechnungen mit elliptischen Kurven übertragen werden, wenn es auf dem diskreten Logarithmusproblem auf endlichen Körpern beruht. Dieses neue elliptische Kurvenlogarithmusproblem ist komplexitätstheoretisch härter als das bisher meist verwendete auf der primen Restklassengruppe und es reichen daher kürzere Schlüssellängen aus, um ein vergleichbares Sicherheitsniveau zu erreichen. Darum sind

diese prädestiniert für die Einsatzumgebung Chipkarte. Für die Anwendung in dieser Arbeit wird vor allem der auf elliptischen Kurven basierende Diffie-Hellmann-Schlüsselaustausch zum Einsatz kommen, wie er in [8, Kapitel 4.3.2] beschrieben ist.

3 Der neue Personalausweis und die dazugehörige Infrastruktur

3.1 Einleitung

Auf Grundlage des Personalausweisgesetzes[12] wird erstmals ein Personalausweis mit integriertem Chip in Deutschland eingeführt. Aus Marketinggründen wurde die offizielle Bezeichnung von „elektronischer Personalausweis“ auf „neuer Personalausweis“ (nPA) geändert, was leider weder charakterisierend noch im Geschichtsverlauf eindeutig ist. Neben der weiterhin bestehenden Funktion als Dokument zur hoheitlichen Identitätsfeststellung bringt vor allem der neue elektronische Identitätsnachweis (eID) eine Basisfunktion der Authentifikation auf ein sichere digitale Infrastruktur, die gesamtwirtschaftlich genutzt werden kann. Hiermit wird erstmals eine allgemein verfügbare Chipkarte angeboten, die es ermöglicht, eine starke Mehr-Faktorauthorisierung auch im Onlineumfeld durchzuführen. Die Ausgestaltung der Protokolle und Reglementierungen soll es erlauben, bisherige Medienbrüche, wie zum Beispiel das Postidentverfahren, zu ersetzen.

Mit dem neuen Verfahren wird eine wechselseitige Authentisierung von Ausweis und Dienst durchgeführt und mit dem zugrundeliegenden Protokoll, dem *Password Authenticated Connection Establishement* (PACE) und der *Extended Access Control* in Version 2 (EACv2), werden vielfältige Sicherheitsziele erreicht. Diese werden inAbschnitt 3.5 auf Seite 24 detailliert bewertet.

Das Bundesamt für Sicherheit in der Informationstechnik (BSI) hat ca. 15 Dokumente veröffentlicht, die die technischen Spezifikationen der einzelnen Infrastrukturbestandteile und die Prozesse festlegen. Maßgeblich für die Gestaltung des nPA ist die Technische Richtlinie 3110 in der Version 2.03[9], die auf dem ICAO Standard 9303[20, 21] basiert, um eine internationale Interoperabilität der ePassport-Anwendung sicherzustellen. Um die Verbindung und Kompatibilität zur restlichen Chipkartenwelt herzustellen, entsprechen Paketformate, Kodierungen und Datenstrukturen auf dem Chip beinahe vollständig dem ISO Standard 7816-4[25].

Der Begriff *Terminal* wird in den Richtlinien des BSI relativ vielfältig verwendet mit unterschiedlichen Bedeutungen. In der TR 3128[10] findet sich eine gute Strukturierung, die in dieser Arbeit verwendet wird. Zunächst spielen für die eID-Anwendung die *Authentisierungsterminals* eine Rolle. Diese werden in *hoheitliche* (auch Inspektionssysteme genannt) und *nichthoheitliche Terminals* unterschieden. Innerhalb der nichthoheitlichen Terminals gibt es die *integrierten Terminals* und die *verteilten Terminals*, und letztlich werden bei *Remote Terminals* noch der lokale und remote Anteil unterschieden (Übersicht in Abbildung 2 auf Seite 18). Als *Remote Terminal* wird die entfernte Komponente des Dienstes oder eines eID-Providers bei der Onlineau-

thentisierung genannt. In der TR 3110 werden dann noch authentisierte und nichtauthentisierte Terminals unterschieden. Diese Zustände kann jedes der bereits beschriebenen Terminals einnehmen und meint, dass erst nach dem Durchlaufen der Terminalauthentiserung innerhalb der EAC (Abschnitt 3.2 auf Seite 19) von einem *authentisierte Terminal* gesprochen werden kann. Zum Beispiel beim PIN-Management haben aber auch *nichtauthentisierte Terminals* Lese- und Schreibrechte. Es ist also damit keineswegs ein unberechtigtes Terminal oder gar ein Angreifer gemeint.

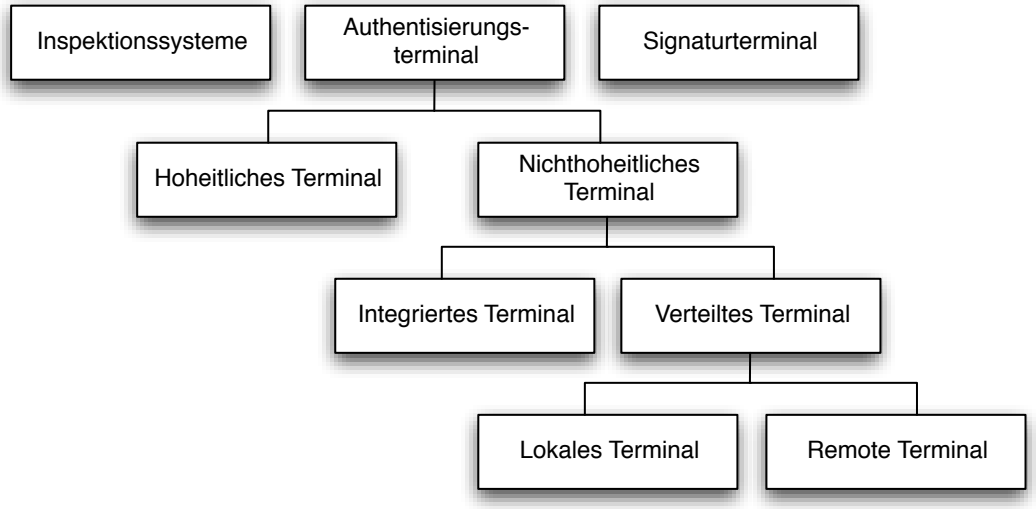

Abbildung 2: Strukturierung der Terminaltypen

Diese Arbeit ist fokussiert auf den nichthoheitlichen Einsatz der eID-Anwendung und beschreibt daher im folgenden auch nur dafür relevante Teile des nPA. Mit der hoheitlich beglaubigten eID kann sich der Ausweisinhaber analog zur analogen Umwelt im Internet authentifizieren. Je nach Zertifizierung und Berechtigung des Dienstes darf dieser nach erfolgreichem Durchlaufen der EACv2 die für ihn notwendigen Identitätsdaten aus dem Ausweis auslesen.

Zusätzlich zum Auslesen von Daten bietet die eID-Anwendung folgende Spezialfunktionen, die in einer sehr datenschutzfreundlichen Art aufgebaut sind.

Dienste- und kartenspezifische Kennung (Pseudonym) Ein Terminal mit dem Recht *Restricted Identification* (RI) kann die Pseudonymfunktion aufrufen. Hierbei erzeugt der Ausweis mit dem Zertifikat des Dienstanbieters und einem auf dem Chip gespeicherten Geheimnis ein Pseudonym, das der Dienstanbieter zur zukünftigen Identifizierung der Karte nutzen kann. Die Pseudonyme sind jeweils nur für den Dienst eindeutig und können dienstübergreifend nicht verknüpft werden. Die RI kann zum Beispiel sehr gut als starkes Token zum Zurücksetzen eines vergessenen Passwortes genutzt werden.

Altersverifikation Eine weitere Funktion ist die sichere Altersverifikation eines Ausweisinhabers. Bei dieser interessiert meistens die Frage, ob schon ein bestimmtes Alter erreicht ist, wobei das konkrete Geburtsdatum keine Rolle spielt. Für diesen Fall kann ein Testdatum als Teil der Terminalauthentisierung übergeben werden. Dieses wird vom Chip verifiziert und beantwortet. Pro Protokolldurchlauf ist damit nur eine Anfrage möglich, und die Anwendung braucht keinen Lesezugriff auf das Geburtsdatum.

Wohnortabfrage Ähnlich zur Altersverifikation können lokalisierte Dienste die eID-Anwendung zur Verifikation eines bestimmten Wohnortes verwenden. Genutzt wird hierfür der amtliche Gemeindeschlüssel mit der kleinstmöglichen Einschränkung auf einen bestimmten Wohnort (Gemeinde). Auch eine Anfrage nach höheren Gliederungsebenen (z.B. Bundesland) ist möglich. Analog zur Altersverifikation wird die Wohnortabfrage als Teil der Terminalauthentisierung übergeben, so dass ein Dienstanbieter den Wohnort nicht durch wiederholtes Abfragen eingrenzen kann.

3.2 Authentisierung nach EACv2

Die Zugriffskontrolle auf die Daten des nPA wird über die allgemeine elektronische Authentisierungsprozedur (*General Authentication Procedure*) hergestellt. Als Sicherheitsobjekte dienen hierbei verschiedene Passwörter (Abschnitt 3.4 auf Seite 22), ein geheimer Schlüssel im Terminal und ein geheimer Schlüssel im Chip.

Die Zugriffsrechte sind abhängig vom verwendeten Passwort, und bei authentisierten Terminals von den in den jeweiligen Berechtigungszertifikaten verbrieften Rechtebeschreibungen. Hierbei werden die Rechte in dem eID *Certificate Authorization Holder Template* (CHAT) in fünf Bytes kodiert und mit dem Prozess der effektiven Authorisierung über die bitweise Verundung der CHAT's der kompletten Zertifikatskette bestimmt. Damit wird sichergestellt, dass mit einem Terminalzertifikat nicht mehr Rechte vergeben und angewandt werden können als die darüberliegenden Authoritäten zugesprochen bekommen haben.

Im besonderen Fall einer Onlineauthentisierung wird das Terminal in zwei Komponenten geteilt, wobei das lokale Terminal (Bürgerclientsoftware) als Schnittstelle und Transporthilfe dient, um den sicheren Kanal vom Ausweischip bis zu einem entfernt im Internet laufenden *Remote Terminal* herzustellen. An den einzelnen Schritten der Authentisierungsprozedur ändert sich dadurch nichts. Diese werden im folgenden beschrieben.

PACE

Das PACE-Protokoll ermöglicht den Aufbau eines verschlüsselten und integritätsgesicherten Kanals zwischen zwei Kommunikationspartnern, hier zwischen dem Chip und dem Terminal,

und dem gleichzeitigen Nachweis, dass sich beide im Besitz des gleichen Passwortes befinden.

Das Terminal muss im Rahmen des PACE-Protokolls die angestrebten Rechte als CHAT übergeben (für den Fall, dass keine Rechte eines authentisierten Terminals benötigt werden, kann der CHAT auch weggelassen werden). Die Berechtigungen aus dem CHAT werden dem Benutzer vorher angezeigt und dieser kann eine Untermenge davon auswählen. Dann bestätigt er mit dem Passwort das Einverständnis zur EAC-Durchführung und authentifiziert sich. Hierbei entsteht ein Angriffsszenario, dass inAbschnitt 5.5 auf Seite 49 beschrieben und eine mögliche Lösung dazu vorgestellt wird.

Als Ergebnis des PACE-Protokolls werden zwei starke Schlüssel zur Verschlüsselung und MAC-Funktion eines sicheren Kanals geliefert, sowie - bei entsprechend vorausgegangenen CHAT - Referenzen auf die im Chip aktuell vorhandenen öffentlichen CVCA-Zertifikate. Alle nachfolgenden Schritte der EAC finden innerhalb des sicheren Kanals statt.

Terminal-Authentisierung

Zum Nachweis der Zugriffsrechte des Terminals bzw. implizit des dazugehörigen Diensteanbieters dient die Terminalauthentisierung (TA). Hierbei wird eine aktuelle Zertifikatskette an den Chip übertragen, die dort validiert wird. Anschließend wird mit einem Challenge-Response-Verfahren nachgewiesen, dass das Terminal den zum öffentlichen Schlüssel des Terminalzertifikates gehörigen privaten Schlüssel besitzt.

Passive Authentisierung

Jedes per TA authentisierte Terminal kann mit der *passiven Authentisierung* die Integrität der gespeicherten Datengruppen prüfen. Hierfür wird das *Document Signer Certificate*, das mit dem EF.CardSecurity vom Chip ausgelesen wird, validiert und die digitale Signatur des EF.CardSecurity überprüft. Anschließend kann das Terminal Hashwerte der Datengruppen berechnen und diese mit den signierten Werten im EF.CardSecurity vergleichen.

Chip-Authentisierung

Bei der Chip-Authentisierung (CAuth) werden mit einem ECDH-Schlüsselaustausch neue Sitzungsschlüssel generiert und der private CAuth-Schlüssel auf dem Chip wird überprüft. Dass der dazugehörige öffentliche Schlüssel nicht verändert wurde, wird über die passive Authentisierung sichergestellt. Mit der Gültigkeit des CAuth-Schlüsselpaares ist sichergestellt, dass der Ausweis echt ist.

3.3 Password Authenticated Connection Establishement

Das PACE-Protokoll ist ein neu entwickelter Vertreter der authentisierenden Schlüsselaustauschprotokolle und wurde speziell für die Anwendung im nPA entwickelt, kann aber auch unabhängig davon in anderen Umgebungen eingesetzt werden. PACE ist als patentfreie Alternative zu bestehenden Verfahren entworfen worden und befindet sich auch im Prozess einer internationalen Standardisierung. Dank dieser Eigenschaften gibt es auch bereits eine freie Implementierung[8]. PACE ist offen gestaltet und kann mit verschiedenen Chiffrieralgorithmen genutzt werden. Für den nPA kommt dabei elliptische Kurvenkryptografie zum Einsatz und es werden zwei ECDH durchgeführt.

Mit PACE sind, basierend auf einem gemeinsamen Geheimnis, zwei Entitäten in der Lage, einen starken Sitzungssschlüssel auszuhandeln. Das Geheimnis kann dabei eine geringe Entropie besitzen und beeinflusst nicht die Qualität der Schlüssel. In[1] wird bewiesen, dass PACE im real-or-random Sicherheitsmodell sicher ist, d. h. ein Angreifer, der mehrere parallel durchgeführte Protokollabläufe beobachtet, kann nicht unterscheiden, ob es sich bei den Nachrichten um Schlüssel oder einfach zufällige Daten handelt. Im Protokollablauf wird niemals das gemeinsame Geheimnis übertragen und alle Nachrichten basieren auf Zufallszahlen, die nach deren symmetrischer Verschlüsselung wieder zufällig sind.

PACE gehört zur Klasse der Zero-Knowledge-Protokolle. Selbst wenn ein Angreifer direkt als eine Entität das Protokoll durchführt, ist er nicht in der Lage, das Geheimnis zu erfahren oder gültige Schlüssel abzuleiten. Er kann lediglich genau ein Geheimnis auf Korrektheit testen und muss dann einen neuen Protokolllauf durchführen. Der Aufwand entspricht also dem einer Brute-Force-Attacke (*online dictionary attack*), dem einfachen Durchprobieren aller möglichen Passwörter. Dies wird mit Hilfe eines Fehlbedienungszählers verhindert (sieheAbschnitt 3.4 auf Seite 22).

Auch bei einer Aufzeichnung des kompletten Durchlaufs können daraus später keine Informationen über das Geheimnis gewonnen werden (*offline dictionary attack*).

Das Protokoll erzeugt Sitzungsschlüssel aus denen weder vorhergehende noch nachfolgende Schlüssel abgeleitet werden können. Die Folgenlosigkeit (*forward-secrecy*) ist also ebenfalls gewährleistet.

Zur Vereinfachung und Abbildung auf den Einsatz im nPA werden für die beiden Entitäten im Folgenden der Ausweischip (PICC) und das Terminal (PCD) angenommen. Das Protokoll kann in vier Phasen unterteilt werden:

Übertragung der Nonce Die EC-DomainparameterD_{PICC} und die mit einem aus dem Passwort abgeleiteten Schlüssel verschlüsselte Nonce s werden an PCD übertragen, wo

[8] http://sourceforge.net/projects/openpace/

die Nonce entschlüsselt wird.

Map2Point Auf beiden Seiten erfolgt die Ableitung der ephemeralen EC-Domainparameter $\widetilde{D}$ mit dem neuen Generator $\widehat{G}$ mittels generischem Diffie-Hellmann-Mapping $\widehat{G} = s \cdot G + H$. H ist ein DH-Schlüssel der Kurve G aus dem Produkt zweier Punkte mit $H = x_{PICC} \cdot x_{PCD} \cdot G$.

ECDH PICC und PCD durchlaufen einen anonymes Diffie-Hellmann-Schlüsselaustausch-Protokoll und erhalten damit beide den Schlüssel K. Aus diesem werden mit der Ableitungsfunktion KDF die zwei Schlüssel $K_{MAC} = KDF_{MAC}(K)$ und $K_{Enc} = KDF_{Enc}(K)$ abgeleitet.

Authentifizierungstoken Zur wechselseitgen Authentifizierung berechnen beide Seiten mit $T_{PCD} = MAC(K_{MAC}, \widetilde{PK_{PICC}})$ und $T_{PICC} = MAC(K_{MAC}, \widetilde{PK_{PCD}})$ ein Token und vergleichen es. Wenn es übereinstimmt, ist der Durchlauf erfolgreich und die beiden Sitzungsschlüssel K_{MAC} und K_{Enc} können als gemeinsames starkes Geheimnis verwendet werden.

3.4 Passwörter

Im PACE-Protokoll kommen für die nichthoheitliche eID-Anwendung und zum PIN-Management folgende Passwörter zum Einsatz:

- die auf dem Kartenkörper aufgedruckte Nummer (CAN)
- das persönliche Passwort des Karteninhabers (PIN)
- das persönliche Super-Passwort des Karteninhabers (PUK)
- das Sperrkennwort

Die CAN ist eine auf dem Kartenkörper aufgedruckte sechsstellige zufällige Nummer und dient zum Aufbau eines sicheren Kanals zwischen Ausweis und Terminal. Sie ist nicht persönlich gebunden und stellt im übertragenen Sinne das Äquivalent einer kontaktbehafteten Schnittstelle dar. Es wird damit ausgesagt, dass der Ausweis beabsichtigt mit dem jeweiligen Terminal in Kontakt treten soll. Beim mehrstufigen Fehlbedienungsprozess (FBP) wird sie außerdem dazu verwendet, um den letzten Versuch einer gültigen PIN-Eingabe zu legitimieren. Die CAN selber ist ein nicht blockierendes Passwort und kann beliebig oft verwendet werden.

Die PIN wird in dem nach ihr benannten Brief als fünfstellige Transportvariante an den Ausweisinhaber übermittelt. Die Transport-PIN muss dann vor dem ersten Einsatz der eID-Funktion

in eine sechsstellige Zahlenfolge geändert werden. Mit der PIN und dem Ausweis können dann 2-Faktorauthentisierungen durchgeführt werden.

Die PUK ist eine zehnstellige nicht änderbare Ziffernfolge, die gemeinsam mit dem alphanumerischen Sperrkenntwort ebenfalls im PIN-Brief mitgeteilt wird. Mittels der PUK ist es möglich, im FBP eine gesperrte PIN wieder freizuschalten. Die Benutzung der PUK ist auf 10-mal beschränkt. Sollte anschließend die Verwendung notwendig werden, muss persönlich eine Ausweisbehörde aufgesucht werden, die das Recht über die TA zum Zurücksetzen der Bedienungs- und Fehlbedienungszähler und zum Neusetzen der PIN nachweist. Da laut [7, Kapitel 3.3] festgelegt ist, dass eine PIN-Änderung nur dort oder mit dem Wissen der alten PIN durchgeführt werden kann (*PIN-change*), wird der zusätzlich als optional beschriebene Mechanismus des Neusetzens der PIN mittels der PUK aus [9, Kapitel 3.5] ausgeschlossen.

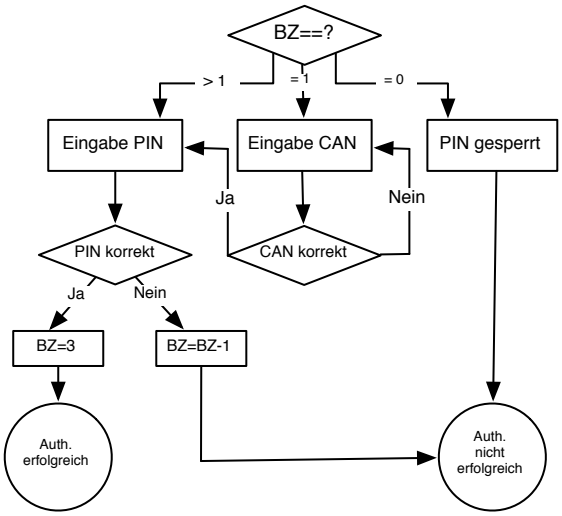

Abbildung 3: Fehlbedienungsprozess nach TR 3127[7, Abb. auf Seite 17]

Um einen Brute-Force-Angriff auf die PIN des Ausweises zu verhindern, wurde diese, wie in Abbildung 3 abgebildet, mittels eines Bedienungszählers (BZ) blockierend gestaltet. Die PIN soll nach drei Versuchen gesperrt werden. Um einen Sabotageangriff zu vermeiden, bei dem beliebige PINs probiert werden bis die PIN-Eingabe gesperrt ist, wird der Bedienungszähler von initial 3 auf 1 heruntergezählt, und die PIN dann zunächst vorübergehend außer Kraft gesetzt. Weitere PIN-Eingabeversuche schlagen nun fehl ohne Auswirkungen auf den Zustand des Ausweises. Erst nach der Eingabe der CAN, womit wieder die physische Verfügbarkeit des Ausweises und die Absicht der Terminalverbindung bestätigt wird, kann im AnSchluss noch einmal die PIN eingegeben werden (*PIN-resume*). Wenn dies wieder fehlschlägt, wird diese geblockt und kann nur noch mit der PUK wieder freigegeben werden (*PIN-unblock*). Die Operationen

zum PIN-Management sind inAbschnitt 5.3 auf Seite 42 implementiert.

3.5 Bewertung der Sicherheitseigenschaften

Dem Design der Systemarchitektur des nPA liegen verschiedene Schutzziele zu Grunde, die hier bewertet werden. Das Schadenspotential mit Einschränkung auf die nichthoheitliche eID-Anwendung ist vergleichbar gering. Die Biometriedaten sind hier nicht zugreifbar und die restlichen Datengruppen jeweils nur mit den entsprechenden Leserechten im Berechtigungszertifikat. Inhaltlich sind die Informationen eines einzelnen nPA's auch bei physischem Zugriff auf den Ausweis durch Ablesen des Aufdrucks direkt zu beschaffen, so dass ein elektronischer Angriff auf die Chipdaten sich nur bei Erreichen großer Ausweismengen lohnt, um zum Beispiel die Meldeadressen zu verkaufen. Ein direkter Zugriff auf eine große Anzahl von Ausweisen ist auf Grund der Funkreichweite von unter 10 cm ebenfalls schwierig, so dass sich vor allem Manipulationen an der Infrastruktur und an den Diensten lohnen.

Vertraulichkeit

Die Vertraulichkeit der Identitätsdaten im nPA wird an allen Schnittstellen und Übertragungswegen sichergestellt. Zunächst sind die Daten auf dem Chip des Ausweises vor Zugriffen geschützt und nur authentisierte Terminals sind in der Lage, eine erfolgreiche TA zu durchlaufen und Leserechte zu erlangen. Welche Teilrechte das sind, wird von der zertifizierenden DV-Stelle für den jeweiligen Anwendungsfall einzeln geprüft und auf ein Minimum festgelegt. Ob der Zugriff auf die Daten auch durch nichthoheitliche Terminals erfolgen darf, ist geschützt durch die Eingabe der Benutzer-PIN. Durch das PACE-Protokoll wird ein vertraulicher Kanal zwischen Chip und lesendem Terminal hergestellt. Dieser Kanal wird im Fall der Onlineauthorisierung bis zum Remote-Terminal beim eID-Provider aufgebaut und durch das Internet getunnelt.

Integrität

Die Integrität während der Kommunikation wird durch den sicheren Kanal im PACE-Protokoll sichergestellt (anhand der MAC in den APDUs). Auf dem Chip werden im *CardSecurity*-Objekt Hashwerte aller Datengruppen gespeichert und anschließend darauf eine digitale Signatur erstellt. Damit ist sichergestellt, dass die Daten auf dem Chip nicht unbemerkt verändert werden können. Dieses Verfahren wird auch als *Passive Authentication* bezeichnet.

Der mögliche Datenabgleich der auf dem Chip gespeicherten Daten mit den auf dem Ausweiskörper aufgedruckten Daten ist ebenfalls eine Integritätseigenschaft des nPA.

Verfügbarkeit

Die Verfügbarkeit auf Protokoll-Ebene wird wieder durch den sicheren Kanal im PACE-Protokoll sichergestellt. Darin ist mit dem kryptografisch abgesicherten Sequenzzähler ausgeschlossen, dass Angreifer Nachrichten einfügen oder unterschlagen können. Eine physikalische Sabotage einer eigentlich berechtigten Verbindung, zum Beispiel durch Störung des Funkfeldes, kann nicht ausgeschlossen werden.

Authentizität

Im nPA und dessen Prozessen wird die Authentizität aller Komponenten untereinander gewährleistet. Der Benutzer authentisiert sich mittels PIN gegenüber dem Ausweis, der Ausweis wiederum mittels CAuth gegenüber dem Terminal, und das Terminal mittels Berechtigungszertifikat gegenüber dem Chip.

Die PIN ist als Bindung an den Ausweis schwächer als eine unter Beobachtung persönlich geleistete Unterschrift. Bei der Onlineauthentisierung ist eine Delegation (freiwillig oder unfreiwillig) der Authentisierung zum Beispiel innerhalb der Familie erstmals möglich. Man kann einfach PIN und Ausweis einer vertrauten Person übergeben. Dies kann dazu führen, dass manche Anwendungen die neuen Onlineverfahren nicht akzeptieren werden. Zusätzlich wird die geleistete Unterschrift in herkömmlichen Authentifikationsverfahren oft in Doppelfunktion, einmal als biometrischer Authentisierungsnachweis und zum anderen auch zur Erlangung der Verbindlichkeit, eingesetzt. Eine PIN-Eingabe kann und soll das nicht leisten, und die Alltagsanforderungen der Dienste muss zeigen, inwieweit hier auf den Nachweis einer Unterschrift verzichtet werden kann.

Bei einer Nutzung in den Szenarien, die nicht am heimischen PC stattfinden, bietet sich dem Anwender eine neue potenziell unbekannte Umgebung, in der sowohl die verwendeten Geräte als auch der angesprochene Dienst kein Vertrauen des Anwenders und zusätzlich eventuell keine bekannte Reputation besitzen. Dem Anwender stellt sich also die Frage, ob er das Risiko einer Nutzung eingehen soll. Während das PACE-Protokoll und die EAC 2.0 sicherstellen, dass eine manipulierte Terminalsoftware weder das Geheimnis noch wieder verwendbaren Daten erhalten kann, besteht die Möglichkeit, mittels manipulierter Hardware an die PIN zu gelangen, welche über das eingebaute Pinpad an einem integrierten Terminal oder das manipulierte Lesegerät eingegeben wird. Unter der Voraussetzung, dass ein manipuliertes Terminal niemals eine Zertifizierung absolviert und damit auch kein Berechtigungszertifikat erhält, wird eine TA nicht möglich sein und somit auch keine Leserechte für die Chipdaten erteilt. Die Vertraulichkeit ist hier also gewährleistet. Der Verlust der PIN kann aber bei einem Diebstahl des Ausweises zum Mißbrauch der elektronischen Identität führen. Es empfiehlt sich also, regelmäßig oder sogar nach

jeder Nutzung die PIN zu ändern (weitere Ausführungen dazu in5.3.1). Darüber hinaus kann das Terminal mit der gestohlenen PIN eine Änderung auf eine beliebige unbekannte PIN durchführen und damit die weitere Nutzung des nPA sabotieren. Die Möglichkeit der PIN-Änderung nach Authentisierung im PACE-Durchlauf mit einer PUK soll nicht umgesetzt werden. Damit bleibt nach einem derartigen Sabotageangriff nur die eine Möglichkeit, persönlich eine Ausweisbehörde aufzusuchen.

Verbindlichkeit

Eine Verbindlichkeit soll im Betrachtungsfeld der eID-Anwendung des nPA intendiert nicht erfolgen. Aus Gründen des Datenschutzes sind die Protokolle und Prozesse so gestaltet, dass nicht verbindlich festgestellt werden kann, wer den Ausweis benutzt hat oder welcher Ausweis für eine eID-Anwendung benutzt wurde. Lediglich die RI erzeugt ein verbindliches Pseudonym, das mit einem kartenspezifischen Schlüssel eindeutig im Raum des jeweiligen Berechtigungszertifikats erzeugt wird.

Originalität

Der geschützte Speicher im Chip enthält die aufgedruckten Daten des Ausweises und stellt mittels dieser Redundanz eine starke Möglichkeit zur Überprüfung der Daten auf dem Ausweiskörper zur Verfügung. Die Fälschungsssicherheit eines neuen Ausweises ist damit höher als beim Vorgänger. Der Aufwand erhöht sich mindestens um die Komplexität des Chipklonens oder Chipfälschens. Zum Klonen wäre es erforderlich, den geschützten Chipspeicher auszulesen und einen Chip inklusive der in Hardware ausgeführten Algorithmen zu reproduzieren. Das Ziel eines solchen Klonversuchs, zu einem gültigen Dokument ein zweites Exemplar zu bekommen mit identischen Daten, ist hierbei viel zu gering als das sich der Aufwand lohnen würde.

Lukrativer ist der Versuch, den privaten Schlüssel der Chip Authentication (dem Ausweisgenerationsschlüssel) aus dem geschützten Speicher auszulesen, um mit dessen Unterstützung eigene „gültige“ Ausweischipkarten herzustellen. Dieser Vorgang ist zwar nur mit extrem hohem Aufwand und entsprechenden Kosten möglich, liefert aber als Ergebnis keinen chipindividuellen Schlüssel, sondern einen Generalschlüssel. Als Gegenmaßnahme zu diesem möglicherweise eines Tages erfolgreichen Angriff wurde in der neuesten TR3110 noch ein optionales *ChipSecurity*-Objekt hinzugefügt [9, A.1.2]. Dieses enthält chip-individuelle Schlüssel, welche wahrscheinlich als Ausweichlösung aktiviert werden können. Besonders für die hoheitlichen Inspektionssysteme ist diese Option durchaus sinnvoll.

Datensparsamkeit, Anonymität und Pseudonymität

Verschiedene Datenschutzmassnahmen des nPA sollen die Daten, die bei einer Nutzung anfallen, minimieren und in der anschließenden Verwendung maximal einschränken:

- ein zufällig erzeugter ATR (lediglich Landeskennung) verhindert die Ausweiswiedererkennung
- zur Vermeidung von signierten Eingabedaten wurde keine *Aktive Authentisierung* umgesetzt (s. [9, Anhang I])
- Vermeidung von signierten auslesbaren Datengruppen, um einen Adresshandel mit hoheitlich beglaubigten Meldeadressen zu verhindern
- spezielle Funktionen zur Alters- und Wohnortabfrage verhindern die Preisgabe der exakten Daten
- die authentisierten Terminals sind über die Berechtigungszertifikate in verschiedene Sektoren unterteilt, so dass keine deutschlandweiten Profile über Sperrmerkmale oder Restricted IDs angelegt werden können. Jeder Dienst hat seine isolierten und nicht mit anderen Diensten kombinierbaren Datensätze.
- PACE als grundlegenes Protokoll erfüllt die Eigenschaften Nicht-Nachverfolgbarkeit, Nicht-Verknüpfbarkeit und Nicht-Übertragbarkeit

Es ist laut den nichttechnischen Regelungen verboten, das Sperrmerkmal des Ausweises zu speichern und als Wiedererkennungsmerkmal zu nutzen. Es bleibt aber die technische Möglichkeit bestehen, und somit steht diese Funktion unabhängig von den Berechtigungen des jeweiligen Terminals bezüglich der RI allen Diensten offen. Das Sperrmerkmal ist ebenso sektorspezifisch und von der Gestaltung her vergleichbar sicher wie die RI, aber eine separate Freischaltung im Berechtigungszertifikat suggeriert hier dem Anwender eine Abwahlmöglichkeit des RI-Einsatzes, die es jedoch aus rein technischer Betrachtung des Sperrmerkmals als alternative ID nicht gibt.

4 Mobiltelefon als Systemkomponente

Ein Mobiltelefon gehört ähnlich wie das Portmonee mittlerweile zur Grundausstattung des zivilisierten Menschen und hat in unserer Gesellschaft den Status der permanenten Verfügbarkeit erreicht. Mit der Verbreitung der leistungsfähigen Klasse der sogenannten Smartphones und der ersten Geräte mit integriertem NFC-Chip entsteht das Interesse, diese Komponente auch

in der Architektur des neuen Personalausweises zur Anwendung zu bringen. Welche Anwendungsmöglichkeiten es gibt und wie sich diese auf die Sicherheitseigenschaften der bisherigen Infrastruktur auswirken, wird in diesem Kapitel analysiert.

4.1 Merkmale und Abgrenzung

Bei der Betrachtung des Mobiltelefons als neue Komponente einer Architektur stellt sich die Frage, warum hiermit überhaupt eine eigene Geräteklasse vorliegt und ob die spezifischen Eigenschaften Auswirkungen auf den Einsatz und in diesem Fall auf die Sicherheit haben. Die Hardwarebasis eines Smartphones unterscheidet sich nur noch in einer Leistungsbegrenzung von tragbaren Computern der Kategorie Laptop oder auch von Subnotebooks. Die Kernkomponenten wie Prozessor, Arbeitsspeicher, permanenter Speicher und Ein- und Ausgabeschnittstellen sind in beiden Geräteklassen vorhanden, so dass man ein Smartphone durchaus als Computer und spezieller noch als tragbaren Computer klassifizieren kann.

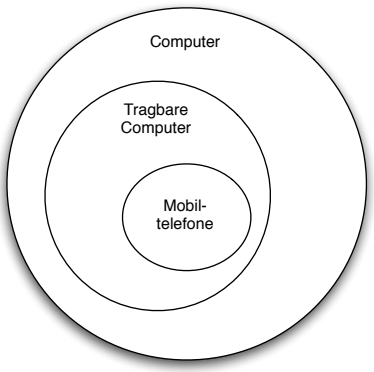

Abbildung 4: Geräteklassen

Wesentliche charakteristische Unterschiede bestehen durch die feste Integration der Tastatur und des Bildschirms, sowie der hardwaregestützten Personalisierung mittels der SIM-Karte. Die Integration von Tastatur und Bildschirm sind sicherheitsrelevant, da Angriffe, die durch das Austauschen oder Einbringen von zusätzlicher Hardware zur Belauschung der Tastatureingaben (z.B. mittels Keylogger am Verbindungskabel oder durch den Kompletttausch der Tastatur) oder zur Manipulation der Bildschirmausgaben, erschwert werden.

Die SIM-Karte ist eine personalisierte Chipkarte und dient dazu, den Benutzer zu authentifzieren. Sie stellt innerhalb eines Telefons einen Sicherheitskern dar, dessen Bedeutung mit den neuen Anwendungen wie der Onlineauthentisierung, e-Ticket oder auch m-Payment (mobiles Bezahlen) stark wachsen wird. Die SIM-Karte ist in der Lage, Daten sicher und nur für den authorisierten Nutzer beschränkt zugreifbar zu machen. Diese Daten können auch Schlüssel für kryptografische Funktionen sein. Als authorisierter Vertrauensanker ist es möglich, in Zukunft personalisierte Anwendungen zur Verfügung zu stellen, die nur bei Besitz der entsprechenden SIM-Rechte oder Schlüssel funktionieren. Es wäre zum Beispiel auch denkbar, dass der Mobilfunkprovider als Verwalter der Identitäten Signaturschlüssel auf den Chip aufbringt, mit dessen Hilfe signierte Applikationen zur Laufzeit verifiziert werden können. Hierdurch könnten Manipulationen an der Software erschwert werden.

Eine weitere Diskussion verfolgt aktuell die direkte Anbindung zukünftiger SIM-Karten an

den NFC-Chip. Mit einem *Single Wire Protocol* (SWP) soll es möglich werden, dass Anwendungen auf der SIM-Karte als „Sicheres Element“ direkt mit per NFC verbundenen Gegenstellen kommunizieren können, ohne dass dazu zusätzliche Software des Telefons eingreift.

4.2 Nutzungsszenarien

Für die Nutzung auf dem Telefon wird nur die eID-Anwendung betrachtet. Die ePassport-Anwendung wird von hoheitlichen Terminals, sogenannten Inspektionssystemen ausgeführt, bei denen eine Verlagerung des Terminals auf das Handy sinnfrei ist. Auch wären die Anforderungen mit dem sicheren Einbringen und Speichern eines hoheitlichen Berechtigungszertifikats aktuell nicht zu erreichen.

Bei der eSign-Anwendung ist die Nutzung auf den heutigen Bildschirmformaten und -auflösungen von Telefonen sehr schwierig und liegt nicht im Fokus dieser Arbeit. Für eine qualifizierte elektronische Signatur ist die Sichtprüfung des zu unterschreibenden Dokuments notwendig. Wie sich diese gestalten lässt und wie man ein Mobiltelefon zu einem vertrauenswürdigen Anzeigegerät (*Trusted Viewer*) machen kann, sei Gegenstand zukünftiger Arbeiten.

Für die eID-Anwendung werden im folgenden die betrachteten konkreten Nutzungsszenarien beschrieben.

4.2.1 Szenario-1: Authentisierung am Automaten

Das erste Nutzungsszenario beinhaltet die nPA-Anwendung an Automaten. Verkaufsautomaten sind integrierte Terminals im Offlinebetrieb mit eigenem Terminalzertifikat. Diese beinhalten sowohl Kartenlesegerät, Tastatur als auch Anwendungslogik. Typisches Beispiel ist die Altersverifikation am Zigarettenautomaten.

4.2.2 Szenario-2: Onlineauthentisierung stationär an Heimrechner

Das zweite Szenario stellt die Nutzung des nPA am heimischen PC mit externem Lesegerät dar. Die eID-Funktion kann hier z.B. zur Anmeldung an eGovernment-Portalen genutzt werden.

4.2.3 Szenario-3: Onlineauthentisierung stationär an Fremdrechner

Das dritte Szenario beinhaltet die Nutzung eines Fremdrechners als lokales Terminal der Onlineauthentisierung in einer unsicheren Umgebung (z.B. Internetcafe). Eine Beispielanwendung ist die Authentifikation gegenüber einer Bank zur Sperrung der Kreditkarte.

4.2.4 Szenario-4: Onlineauthentisierung mobil

Das vierte Nutzungsszenario stellt die vollständig mobile Nutzung in einer potenziell unsicheren Umgebung dar. Als mobiles Gerät dient im Szenario zunächst ein tragbarer Computer mit externem Chipkartenleser.

4.2.5 Szenario-5: PIN-Management

Im fünften Nutzungsszenario möchte der Anwender die eID-PIN ändern, den Fehlbedienungszähler zurücksetzen oder eine Sperrung aufheben.

4.3 Einsatzmöglichkeiten

Für den Einsatz eines Mobiltelefons mit integriertem NFC werden im folgenden die allgemeinen Vor- und Nachteile und im Anschluss die konkreten Einsatzmöglichkeiten innerhalb der Ausweisinfrastruktur beschrieben.

Vorteile Wichtigster Vorteil für den Einsatz des Mobiltelefons ist die sehr hohe *Verbreitung*. Wie schon in der Einleitung beschrieben, sind selbst die CAT-B Leser teurer als allgemein akzeptiert wird. Damit besteht die Gefahr, dass sich die Nutzung des nPA als weit verbreitete Onlineauthentifizierungskarte nicht in der Masse durchsetzen wird. Eine Übertragung der Funktionalität aufs Handy könnte ein entscheidendes Momentum zur Etablierung als Standardauthentisierung sein.

Die physische Manipulation der Telefonhardware (Skimming) ist wesentlich erschwert durch die permanenten Mitführung und Nutzung und ein weiterer Vorteil liegt in dem *personalisierten Sicherheitskern* mit der SIM-Karte des Telefons. Damit ist es möglich, Daten (Schlüssel, Passwörter) sicher und zugriffsbeschränkt zu speichern.

Nachteile Die *Sicherheit* kann sich abhängig von der Ausgestaltung und der Sicherheit des mobilen Betriebssystems gegenüber einer geschlossen Lesegeräteanwendung verschlechtern. Zusätzlich steigt der Entwicklungsaufwand zur Unterstützung von mobilen Geräte durch die Vielzahl der etablierten Betriebssysteme stark an. Die Kosten einer Zertifizierung der Hardware, der sehr volatilen Gerätegenerationen, sowie für die Sicherheitsprüfung der jeweiligen Betriebssystemsoftware können die Kosten eines dedizierten Lesegerätes bei weitem übersteigen und müssten dem Anwender direkt oder indirekt ebenfalls in Rechnung gestellt werden. Trotz anderer Skalierungseffekte bei den wesentlich höheren Telefonstückzahlen könnte das den ursprünglich angestrebten Preisvorteil beim Anwender zunichte machen.

Die Tabelle4.3 bietet eine Übersicht der möglichen Komponentenersetzungen durch ein Mobiltelefon. Das Telefon als integriertes Terminal spielt keine Rolle, da für diese Konstruktion ein Berechtigungszertifikat in das Telefon eingebracht und regelmäßig erneuert werden müsste. Die beschriebenen Nutzungsszenarien haben für einen mobilen „Automaten“ in Form eines Telefons keine Anwendung.

Szenario / Einsatz als	1	2	3	4	5
Lesegerät	+	++	-	++	-
lokales Terminal exkl. Lesegerät	-	+	-	+	-
lokales Terminal inkl. Lesegerät	-	+	+	++	++

\+ technisch möglich
++ möglich und sinnvoll

Tabelle 2: Einsatzmatrix für ein Mobiltelefon in den Nutzungsszenarien

Für das Nutzungsszenario Szenario-1 ist lediglich ein Einsatz des Mobiltelefons denkbar als externes Lesegerät, aber in der Umsetzung aufwändig und unwahrscheinlich. Alle Komponenten sind bereits im Automaten enthalten, aber man könnte eine Schnittstelle entwerfen, die es erlaubt, ein mitgebrachtes Lesegerät zu nutzen. Der Vorteil wäre ein Schutz vor Skimming des Automaten im Falle einer Ausprägung als CAT-S Leser.

In allen anderen Szenarien gibt es die Möglichkeit, das Telefon als lokales Terminal mit integriertem Lesegerät zu nutzen. Dieser Einsatz der neuen Komponente Telefon entspricht dem Wechsel der gesamten Umgebung auf ein voll mobiles Szenario. Für Szenario-2 und Szenario-3 hat das den Nachteil, dass man von einer vollwertigen PC-Umgebung mit großer Bildfläche auf ein Mobiltelefon wechseln muss, was im Fall des Wechsels nur für eine nPA-Nutzung zeitaufwändig, und für einen permanenten Wechsel mit starken Komforteinbußen verbunden ist. Lediglich im mobilen Szenario-4 gleicht sich der Verlust des großen Bildschirms mit der gewonnenen besseren Transportfähigkeit des kleinen, leichten Einzelgerätes wieder aus.

Für Szenario-3 wird eine Veränderung der vorgefundenen Systemkomponenten ausgeschlossen. Der Einsatz des Telefons als Lesegerät in Szenario-3 wäre denkbar, wenn es möglich wäre, das Telefon als Standardlesegerät am nicht veränderten Betriebssystem des PCs anzuschließen, zum Beispiel per USB.

Die Nutzung des Telefons lediglich als Rechnerkomponente, auf welcher der Bürgerclient ausgeführt wird, ist theoretisch möglich, birgt aber die Schwierigkeit eines Anschlusses des externen Lesegerätes an das Mobiltelefon. Selbst falls hardwareseitig ein USB-Anschluss verfügbar ist, fehlt es auf handelsüblichen Geräten an der passenden Treibersoftware. Dieses Szenario ist für die Zukunft nicht ausgeschlossen, wird hier aber nicht weiter verfolgt.

Im folgenden werden die beiden als sinnvoll erarbeiteten Einsatzszenarien als Lesegerät und als lokales Terminal mit integriertem Lesegerät genauer untersucht.

4.3.1 Lesegerät

Lesegeräte gibt es wie beschrieben in Abschnitt 2.4 auf Seite 7 in drei Klassen. Um die Anforderungen bei einer Übertragung auf das Mobiltelefon bewerten zu können, bietet sich eine Orientierung an den Merkmalen der jeweiligen Klasse, wie sie in [6] beschrieben sind, an.

Bei *CAT-K* Lesern wird eine kontaktbehaftete Schnittstelle vorausgesetzt, was für die Anwendung auf das NFC-Telefon ein Ausschlusskriterium ist.

Ziel dieser Arbeit ist die Untersuchung eines PACE-fähigen Lesegerätes, womit die *CAT-B* Leserklasse ebenfalls ausscheidet.

Damit bleiben die Anforderungen an einen Standardleser [6, Kapitel 5.4] auf ein Mobiltelefon (Mt) zu übertragen. Die Nummerierung entspricht aus Referenzgründen der Quelle.

1.1 Sichere PIN-Eingabe: Anforderung erfüllbar. Es gibt ein Pinpad, Möglichkeit der Anzeige des Betriebszustandes und PACE lässt sich implementieren .

1.2 Sichere PIN-Übertragung über die kontaktlose Schnittstelle: Anforderung erfüllbar. Die Software auf dem Mt kann Secure Messaging kontrollieren.

1.3 Autarke Bildung des Schlüsselmaterials: Anforderung erfüllbar. Das Schlüsselmaterial kann innerhalb des PACE-Protokolls im Mt erzeugt werden.

2.1 Unterstützung verschiedener Betriebssysteme: Anforderung schwer erfüllbar. Die hier geforderte Treiberunterstützung für gängige Betriebssysteme und die Verwendung der eCard-API ist adhoc nicht umsetzbar. Eine Treiberunterstützung benötigt ein standardisiertes Protokoll, für dessen Implementierung auf der Mt-Seite entsprechende Lowlevel-APIs notwendig sind, die auf den heute verfügbaren Zielgeräten nicht existieren. Dafür müsste es gelingen, eine USB-Verbindung herzustellen und dann über PC/SC eine Integration in die eCard-API umzusetzen. Eine eCard-API Implementierung auf dem Mt ist mit heutigen Stand ebenfalls als sehr schwierig einzustufen. Die API ist relativ komplex und umfangreich. Inwieweit es möglich ist, nur Teile davon herausgelöst zu realisieren oder eine mobile eCard-API davon abzuleiten, ist Gegenstand weiterer Arbeiten.

3.1 Optimierung der Kosten des Standard-Chipkartenlesers Anforderung erfüllbar. Dieser Punkt ist mit einem Mt, als von Natur aus schon bestehendem Multifunktionsgerät mit unterschiedlichen Anwendungen, hervorragend gelöst. Die Kostenoptimierung

entsteht hier nicht nur durch die Skalierung auf verschiedene Anwendungen, sondern ebenfalls über die Vermeidung der Hardwarekosten.

4.1 Installation des Standard- Chipkartenlesers Anforderung erfüllbar. Die Anwenderfreundlichkeit eines Mt als Lesegerät lässt sich mit äquivalenten Mitteln herstellen wie bei einem Standardleser. Als schon von anderen Funktionen her vertrautes Gerät ist die Bedienerintuition am Mt wahrscheinlich sogar höher. Problematisch vergleichbar zu Punkt 2.1 ist die hier aufgeführte Verbindung zum PC mittels einer Standardschnittstelle. Diese muss entworfen und umgesetzt werden.

Neben diesen Anforderungen werden in Tabelle 4.3.1 die obligatorischen Module für ein Mt als Standardleser analysiert.

Nr.	Eigenschaft	auf Mobiltelefon
2S-1	Fehlertoleranz	umsetzbar
2S-2	Elektrische Eigenschaften kontaktlos	vorhanden
2S-3	Hostinterface	umsetzbar*
2S-4	Transport von Zeichen	vorhanden
2S-5	Chipkartenprotokoll	umsetzbar
2S-6	Programmierschnittstelle	umsetzbar*
2S-7	Pinpad	vorhanden
2S-8	Sichere PIN-Eingabe	vorhanden
2S-9	PACE Verfahren	umsetzbar
2S-10	Firmwareupdate	vorhanden
2S-11	Sicherheitsmodus Anzeige	umsetzbar**
2S-12	Umweltanforderungen	umsetzbar
2S-13	Sicherheitsgutachten	umsetzbar
2S-14	Verfügbarkeit	vorhanden
2S-15	PACE Schlüsselerzeugung	umsetzbar

Tabelle 3: Obligatorische Module eines Standardlesers

(*) Auch hier tauchen die oben bereits beschriebenen Herausforderungen eines Standardhostinterface (z.B. USB oder Bluetooth) und einer Zugriffsschnittstelle (z.B. PC/SC) für die Integration des Lesegerätes in den Bürgerclient auf.

(**) Im Punkt 2S-11 wird eine Sicherheitsmodus-Anzeige gefordert, die dem Benutzer eindeutig signalisieren soll, dass sich das Gerät in einem speziellen gesicherten Modus befindet zur PIN-Eingabe. Neben der Fragestellung eines entsprechenden GUI/Haptik-Designs und der Untersuchung der Akzeptanz durch den Anwender bleibt hier die Frage offen, wie denn ein gesonderter abgesicherter Modus innerhalb des Mts umgesetzt werden kann. Viele Mts unterstützen

parallele Nutzerprozesse und müssten für die PIN-Eingabe entsprechend gesperrt werden.

4.3.2 Lokales Terminal mit integriertem Lesegerät

In diesem Einsatzszenario wird zusätzlich zu der Betrachtung in 4.3.1 weitere Logik auf das Mobiltelefon gebracht, um die volle Funktionalität eines Lokalen Terminals der Onlineauthentisierung zu erreichen. Diese Zusatzlogik entspricht dem Bürgerclient als Vermittler zwischen nPA und dem Remote-Terminal des Onlinedienstes oder dessen eID-Providers. Der Bürgerclient in der aktuellen Testversion für Mac OSX ist 179 MB groß. Die Programmgröße dient hier als Indiz für die Herausforderungen bei der Portierung eines Bürgerclienten auf eine mobile Plattform. Vorraussetzung wird auch an dieser Stelle sein, die implementierte eCard-API auf eine mobile Version zu verschlanken. Trotz dieser anstehenden Umsetzungsschwierigkeiten bleibt der Einsatz auf einem Telefon sinnvoll, insbesondere für das Szenario-4.

Für Szenario-5 ist nur eine Untermenge der Anwendungsfunktionalität des Bürgerclienten notwendig. Dieses Nutzungsszenario und weitere Beispielanwendungen eines nicht authentisierten Terminals werden inAbschnitt 5 vorgestellt.

4.4 Bewertung der Sicherheit

Die in Abschnitt 3.5 auf Seite 24 bewerteten Sicherheitseigenschaften des nPA und der dazugehörigen Protokolle werden beeinflusst von dem Einsatz des Mobiltelefons und dem Szenario einer mobilen Nutzung. Hierbei sind die Authentizität, die Vertraulichkeit und die Datenschutzaskepte betroffen.

Das beschriebene Problem der Authentizität von fremden Geräten lässt sich durch den Einsatz eines Mobiltelefons vermeiden. Die Möglichkeiten zur Manipulation eines täglich genutzten und immer mitgeführten Gerätes sind als wesentlich aufwändiger zu bewerten als bei selbst aufgestellten Automaten oder Lesegeräten in Internetcafe's. Ein weiterer Vorteil entsteht durch die Vertrautheit des Anwenders mit seinem Telefon im Vergleich zu wechselnden Automatenschnittstellen oder Bürgerclientversionen. Das Risiko einer Fehlbedienung wird reduziert. Dies kann zum Beispiel auf Reisen mit fremdsprachigen Schnittstellen während der PIN-Eingabe zu einer höheren Bedienungssicherheit verhelfen. Das Mobiltelefon als personalisiertes Gerät kann Metainformationen zur nPA-Nutzung speichern und zur Verbesserung der Sicherheit nutzen. Es wäre zum Beispiel möglich, die CAN oder auch die *EF.CardAccess* auf der SIM-Karte abzulegen, und bei einer Abweichung vom aktuell eingesetzten Ausweis den Nutzer um die Bestätigung eines Ausweiswechsels zu fragen.

Andererseits besteht beim Einsatz des Telefons die Möglichkeit der Manipulation der darauf laufenden Lesegerät-/Terminalsoftware durch Trojaner oder Malware. Dieser könnte über

Betriebssystemmanipulation die PIN mitschneiden und diese als Passwort in anderen Authentisierungen des Benutzers verwenden. Da Anwender oft ähnliche Passwörter in unterschiedlichen Systemen verwenden, stellt dies einen möglicherweise erfolgreichen Angriff dar. Auch im Bereich des Datenschutzes kann manipulierte Software ein Risiko darstellen. Diese könnte die nPA-Nutzungsdaten (Zeitpunkt und Ort) protokollieren. Aus diesem Bewegungsprofil, kombiniert mit dem Wissen des Einsatzes einer starken Authentifizierung, lassen sich potenziell relevante Informationen ableiten.

Demzufolge ist beim Mobiltelefoneinsatz ein wichtiges zusätzliches Schutzziel, die Integrität der Softwarebestandteile der Lesegerätanwendung zu gewährleisten, und darüber hinaus, als notwendige Basis, die des kompletten Betriebssystems. Auf Grund der Ähnlichkeit von Smartphonearchitekturen mit Standard-PCs ist diese Aufgabe komplex und wird bereits mit den Standards der Trusted Computing Group[9] vorbereitet. Die SIM-Karte könnte auf dem Telefon als Vertrauensanker dienen. Die Gestaltung der Betriebssystemarchitektur hat maßgeblichen Einfluss auf die Zertifizierbarkeit eines Mobiltelefons als Lesegerät.

[9] http://www.trustedcomputinggroup.org/trusted_computing

5 Anwendungen des Telefons als nicht authentisiertes Terminal

5.1 Einleitung

Unter der Prämisse einer zum jetzigen Zeitpunkt praktisch möglichen Umsetzung von Funktionalitäten im Zusammenspiel zwischen einem frei verfügbaren Telefon und dem neuen Personalausweis schränken sich die Anwendungsmöglichkeiten durch verschiedene Aspekte stark ein.

Zunächst besteht zum aktuellen Zeitpunkt keine Möglichkeit aus einer EAC-PKI ein gültiges Berechtigungszertifikat für eine Terminalauthentifizierung auf einem Mobiltelefon zu bekommen. Um diesen Teil der Infrastruktur zu simulieren und die entsprechenden Protokolle zu implementieren, bedürfe es also eines komplett simulierten Ausweises mit dazugehöriger PKI und serverseitiger Infrastruktur. Auch wenn bereits gute Ansätze zur Umsetzung einer virtuellen Smartcard[10] existieren, würde das den Rahmen dieser Arbeit sprengen. Daher wird der Anwendungskreis auf ein nichtauthentisiertes Terminal beschränkt. Das Telefon wird durch eine zusätzliche Software zum mobilen Lesegerät. Hierbei sind auch praktisch einsetzbare Funktionalitäten entstanden, die gleichzeitig dazu dienen können die Sicherheit im Umgang mit dem Ausweis zu erhöhen.

Weiterhin ist das Ziel dieser Arbeit die Möglichkeiten bereits vorhandener Mobiltelefone zu erforschen, um vor allem eine mögliche Verbesserung der Bedienbarkeit und durch die sehr hohe Verbreitung von Mobiltelefonen eine hohe Durchdringung mit ausweiskompatiblen Lesegeräten, je nach Ausprägung möglicherweise aller drei Leserkategorien, zu erreichen. Eine kurze Marktanalyse soll also ein Gerät heraus filtern mit internem NFC Chip und möglichst frei oder mit Unterstützung des Herstellers programmierbarem Betriebssystem mit NFC Kommunikationsunterstützung. Wie in 5.1 dargestellt, schrumpft die Anzahl verfügbarer Kandidaten von 11 Geräten mit NFC Unterstützung (aus Wikipedia [11] und NFC Research [12]) bei der Suche nach frei verkäuflichen Exemplaren auf lediglich 3 Geräte das Motorola L7 SLVR in der NFC Spezialversion sowie die beiden Nokia Geräte 6212 und 6131. Eine gut dokumentierte Untersützung der NFC Schnittstelle in Form des umgesetzten JSR-257 (Contactless Communication API) findet sich dann in dem Java SDK von Nokia, so dass die Auswahl auf das 6212[13] fällt und wie zu sehen beinahe konkurrenzlos ist.

Angesichts der Herstelleranzahl und millionenfach verkaufter Mobilgeräte ist diese einge-

[10] http://sourceforge.net/projects/vsmartcard/
[11] http://en.wikipedia.org/wiki/Near_Field_Communication#NFC-enabled_handsets
[12] http://www.nfc-research.at/index.php?id=45
[13] http://developers.nokia.com/Devices/Device_specifications/6212_classic/

Gerät	verfügbar?	Bemerkung
Nokia 6216 Classic	nein	abgekündigt (Feb. 2010)
Nokia 6212 Classic	ja	
Nokia 6131 NFC	ja	
Samsung S5230	nein	NFC Version nicht im Handel
Samsung SGH-X700 NFC	nein	Prototyp
Samsung D500E	nein	ältestes Gerät (seit 2005)
SAGEM my700X Contactless	nein	
LG 600V contactless	nein	
Motorola L7 (SLVR) NFC	ja	verwendet von der Deutschen Bahn im Touch&Travel Pilot
Benq T80	nein	ev. nicht mehr verfügbar
Sagem Cosyphone	nein	bisher nur angekündigt

Tabelle 4: Verfügbare NFC-Telefone (Stand: März 2010)

schränkte Auswahl an NFC Geräten in Deutschland durchaus eine Überraschung und macht die Annahme mit der Nutzung von Mobiltelefonen eine höhere Verbreitung an Lesegeräten zu erreichen zumindest für den aktuellen Stand hinfällig. Lediglich Nokia als Pionier im NFC Umfeld und Gründungsmitglied des NFC Forums stellt bisher mit seinen Geräten in größeren Stückzahlen Seriengeräte zur Verfügung. Das NFC Forum von Nokia, Phillips und Sony gegründet zur Standardisierung und Kommerzialisierung von NFC im März 2004 hat mittlerweile ca. 130 Mitglieder. Es ist also zu erwarten, dass die verschiedenen Prognosen zu einer höheren Verbreitung von NFC [14] tatsächlich in naher Zukunft eintreffen und im Wechselspiel mit den nun aufkommenden elektronischen Ausweisen auch entsprechende Anwendungen zur Nutzung entstehen. Im Juni 2010 hat Nokia angekündigt das alle von ihnen neu aufgelegten Smartphones ab 2011 mit NFC ausgerüstet sein werden [15].

5.2 Architektur

Die Anwendungsentwicklung für Mobiltelefone unterscheidet sich (noch) stark von der für die klassische PC und auch die mobile PC Umgebung. Auch wenn diese Geräteklassen immer mehr verschmelzen und die Telefone mehr und mehr zu Smartphones und dementsprechend leistungsfähiger werden, gibt es auch über die Hardware hinaus beschränkende Aspekte. Während vor allem CPU und Arbeitsspeicher fokussiert auf die Größe und den Stromverbrauch meist deutlich

[14] http://www.nearfieldcommunicationsworld.com/2010/06/04/33823/one-in-five-mobile-phones-to-feature-nfc-by-2012/

[15] http://www.nearfieldcommunicationsworld.com/2010/06/17/33966/all-new-nokia-smartphones-to-come-with-nfc-from-2011/

langsamer arbeiten, besteht bei den Betriebssystemen und den darauf aufsetzenden Schnittstellen für die GUI und den Zugriff auf die Systemkomponenten eine sehr stark diversifizierte Welt.

5.2.1 Software

Das Nokia 6212 bietet dank Java ME™ Unterstützung die Möglichkeit in der grafischen Darstellung und den Standardbibliotheken systemunabhänging zu programmieren. Der Zugriff auf bestimmte Funktionen wie kryptografische Bibliotheken oder die Kommunikation per NFC Standard sind dann aber abhängig von der Unterstützung und der Bereitstellung entsprechender Bibliotheken durch den Hersteller. Nokia bietet in der „Security and Trust Services API for J2ME“ (SATSA, JSR-177) Bibliothek nur grundlegende Hashalgorithmen und symmetrische Verschlüsselung an, die für die Umsetzung der TR-3110[9] nicht ausreichen. Insbesondere die Unterstützung von elliptischen Kurven muss daher durch die Bouncy Castle Bibliothek ergänzt werden. Eine Übersicht über alle verwendeten Komponenten zeigt die Tabelle 5.

Komponente	Bereich	Version
Java™SE SDK (build 1.6.0_20)	Entwicklung	1.6.0_20
JUnit [16]	Test	4.8.2
Apache Ant[17]	Paketierung	1.7.1
Antenna[18]	Paketierung	1.2.1-beta
ProGuard[19]	Paketierung	4.5
Series 40 Nokia 6212 NFC SDK[20]	Laufzeit	1.0
Bouncy Castle Crypto APIs[21]	Laufzeit	1.45
Light Weight User Interface Toolkit (LWUIT)[22]	Laufzeit	1.3

Tabelle 5: Softwarekomponenten

5.2.2 Hardware

Neben dem Zielgerät 6212 Classic wurde für die Entwicklung weitere Hardware verwendet. Zunächst als Grundlage ein Macbook Pro mit dem externen Smartcardlesegerät SCL011[23] vom Hersteller SCM Microsystems und des Weiteren der Testausweis der Bundesdruckerei aus dem

[16] http://junit.org/
[17] http://ant.apache.org/
[18] http://antenna.sourceforge.net/
[19] http://proguard.sourceforge.net/
[20] http://www.forum.nokia.com/
[21] http://bouncycastle.org/
[22] http://java.sun.com/javame/technology/lwuit/
[23] http://www.scmmicro.com/de/products-services/chipkartenleser-terminals/kontaktlos-dual-interface/scl011.html

offenen Anwendertest. Der Emulator aus dem Nokia SDK funktioniert nur auf dem Betriebssystem Windows und außerdem unterstützt der Nokia NFC Manager zur Kommunikation über die kontaktlose Schnittstelle das SCL011 Lesegerät nicht, so dass eine vollständige Emulation zum Testen nicht möglich war. Die Unit- und Integrationstests zu den funktionalen Anforderungen mit Ansprache des Ausweises wurden daher direkt aus dem Java SDK heraus über die smartcardio API durchgeführt und die Tests der LWUIT Oberfläche mit dem Anwenderworkflow wurde über eine Simulation eines generischen Mobiltelefons mit Hilfe des Oracle Java ME SDK 3.0 für Mac OS durchgeführt. Für das Nokia 6212 sind leider keine Hardwarespezifikationen zur CPU veröffentlicht und dies trifft leider auch die ebenfalls auf S40 5. Edition basierenden Geräte mit ähnlichen Serienbezeichnungen zu. Das nächste dann bereits S60 unterstützende Gerät ist das 6220 Classic mit einer ARM11 CPU mit 369 MHz. Dies stellt damit sehr wahrscheinlich die Obergrenze der CPU Ausstattung dar. Für Java ME Anwendungen steht ein maximaler Heapspeicher von 2MB zur Verfügung und die Applikation darf höchstens 1MB groß sein.

5.2.3 Schnittstellen

Die Applikation hat zur Benutzung eine graphische Benutzerschnittstelle über die die aktuelle Aufgabe ausgewählt, die jeweilig notwendigen Daten eingegeben und der Status dargestellt wird. Zur Dateneingabe wird die im Telefon integrierte Tastatur verwendet. Weiterhin wird über den integrierten NFC Chip eine Funkschnittstelle zum Chip des Ausweises etabliert. Für die Zeitaktualisierung (Details siehe 5.4) wird zusätzlich eine Internetverbindung benötigt. Der Zugriff auf das Dateisystem ist zur Ablage einer Logdatei optional. Weitere Komfortfunktionen könnten neben einem Zugriff auf das Dateisystem des Telefons auch den Zugriff auf die SIM Karte als sicheren Speicher notwendig machen. Welche Funktionen das sein könnten, wird in den folgenden Anwendungsbeschreibungen erläutert.

5.2.4 Implementierung

Hier wird die, allen folgenden Funktionalitäten zu Grunde liegende, Implementierungsstruktur beschrieben. Mit dem Namen Mobiler Personalausweis (kurz moPA) wird die Javaapplikation als zusammengefasstes Softwarepaket bezeichnet und installiert. Die Basispaketstruktur und ein Teil der Klassen zur Umsetzung der initialen Verbindung inklusive eines PACE Protokolldurchlaufs ist angelehnt an die Beschreibung von Moritz Horsch [17, Kapitel 5.3]. PACE wird als Grundlage für alle Ausweisanwendungen genutzt, so dass sich die Wiederverwendung der gelungenen Struktur anbietet. Die Klassen und Pakete sind für die jeweiligen neuen Funktionalitäten erweitert. Insbesondere wurde die Klasse `CipherSuite` (Abbildung 5 auf Seite 40) erweitert um generische Methoden zum ver- und entschlüsseln mit AES, wie es dann im

`SecureMessaging` (Abbildung 6 auf Seite 41) Protokoll benötigt wird.

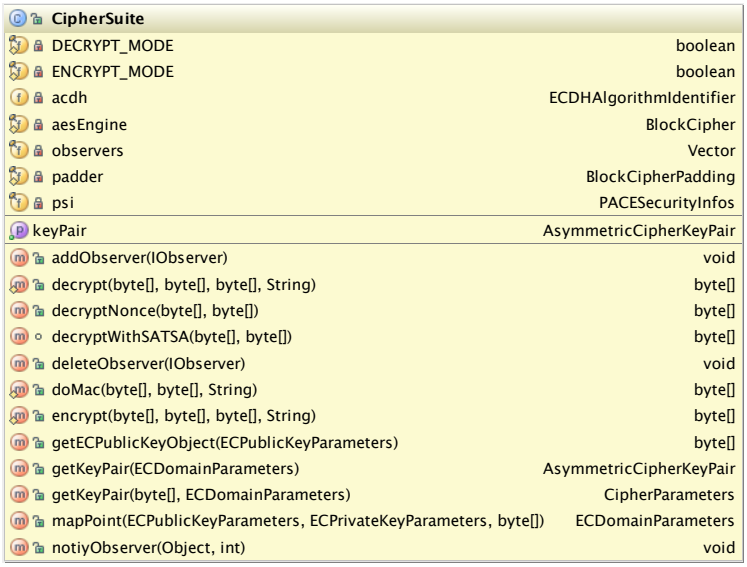

Abbildung 5: Klassendiagramm der `CipherSuite`

Die Kommunikation zwischen Chip und Lesegerät wird in strukturierten Datenpaketen sogenannten Application Protocol Data Units (APDU) nach dem standardisierten Format im ISO 7816 [25] abgehandelt. Es gibt dabei zwei Typen die Kommando- und Antwort-APDUs. Kommando-APDUs (cAPDU) bestehen aus einem *Header* und einem *Body*, Response-APDUs (rAPDU) aus *Body* und *Trailer*. Bei beiden ist der *Body* optional.

- $cAPDU = ((CLA, INS, P1, P2), [Lc, Data, Le])$
- $rAPDU = ([Data], (SW1, SW2))$

Lc ist die Länge der Kommandodaten und Le die Länge der erwarteten Antwortdaten beide in 0 bis 3 Byte kodiert. In [9, Anhang F] wird die Struktur einer *Secure Messaging* APDU und deren Mapping auf die herkömmliche ungesicherte Variante beschrieben. Hierbei werden die Daten der APDU jeweils verschlüsselt und mit einem MAC integritätsgesichert. Beide dafür verwendeten Schlüssel stammen aus dem vorhergehenden PACE Durchlauf.

- $cAPDU = ((CLA, INS, P1, P2),\ [Lc', '87'\ L\ '01'\ Enc.Data,\ '97'\ L\ Le,\ '8E\ 08'\ MAC,\ '00'])$
- $rAPDU = (['87'\ L\ '01'\ Enc.Data,\ '99\ 02'\ SW1\ SW2,\ '8E\ 08'\ MAC],\ (SW1, SW2))$

Die Klasse `SecureMessaging` stellt die grundlegende Logik zur Verschlüsselung/Entschlüsselung sowie der Integritätssicherung mittels Message Authentication Codes

(MAC) von ADPU's zur Verfügung. Mit Hilfe dieser Verfahren kann der sichere Kanal zwischen den beteiligten Kommunikationspartnern hergestellt werden. Über die Absicherung hinaus wird in der Klasse der Sequenzzähler (SSC) verwaltet. Wenn eine Nachricht manipuliert wurde, schlägt die MAC Prüfung fehl und es wird eine `AuthException` ausgelöst, die dann von den darüber liegenden Schichten entsprechend behandelt werden muss.

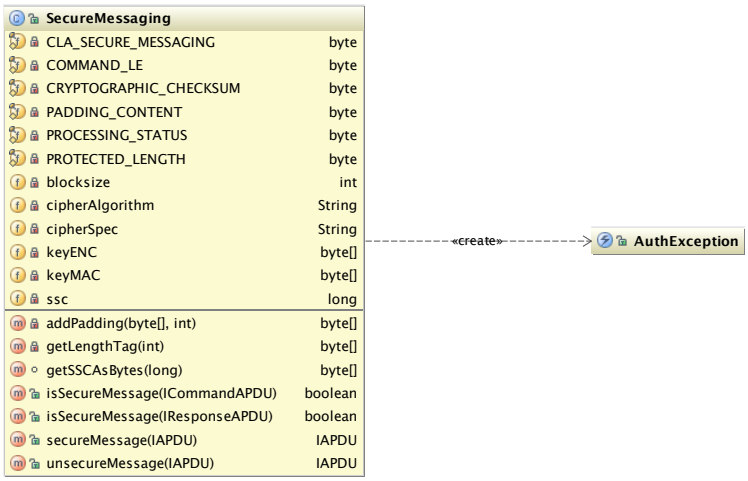

Abbildung 6: Klassendiagramm vom `SecureMessaging`

Nach Problemen mit der NFC Verbindungsqualität ist die GUI jetzt so gestaltet, dass nach Auswahl der Funktion alle erforderlichen Eingabedaten abgefragt werden und erst dann die Funkverbindung zum Ausweis hergestellt wird. So wird verhindert, dass durch die Bewegung bei der Bedienung der Tastatur bereits Verbindungsabbrüche entstehen, die jeweils zum kompletten Protokollabbruch führen. Die Bedienelemente sind überschaubar. Es gibt ein Hauptmenü, eine Eingabemaske für das jeweilige Geheimnis, eine Verlaufs- und eine Ergebnisanzeige. Beispiele dazu finden sich in Abbildung 7 auf Seite 42.

Abbildung 7: Benutzeroberfläche des Prototyps im Emulator

5.2.5 Paketierung

J2ME Anwendungen stellen im Vergleich zu herkömmlichen Javaanwendungen erhöhte Anforderungen an den Kompilier- und Bauprozess. Das Endresult und damit die fertig zu distributierende Software sind eine .jar Datei mit der maximalen Größe von 1MB (gerätespezifisch) sowie eine dazugehörige Beschreibungsdatei .jad in der die Midlets sowie die Dateigröße und die zu startende Main Klasse spezifiziert wird. Das Gerät kann die Applikation dann wie ein natives Programm bei der Auswahl im Ordner starten. Weitere Vorraussetzung ist eine Vorprüfung (Preverify) der kompilierten Klassen, um Platz zu sparen und den Startprozess der Applikation zu verkürzen. Da die Applikation inklusiver aller benutzten Drittbibliotheken in einem Archiv zusammengefasst werden, ist es außerdem notwendig zwei Operationen des ProGuard Tools `shrink` und `obfuscate` durchzuführen. Nur damit ist es möglich das aktuell ca. 400 KB große `mobilerPA.jar` herzustellen. Ohne diese Maßnahmen wäre das Ergebnis 1,6 MB groß und damit nicht installierbar auf dem Zielgerät. Zusätzlich sorgt die Verkleinerung natürlich für schnellere Ladezeiten und weniger Speicherverbrauch zur Laufzeit.

5.3 PIN-Management

5.3.1 Motivation

Durch die Realisierung der PIN Management Funktionalitäten auf dem Telefon kann der Nutzer mit einer ihm vertrauten Benutzerführung und auf einem bekannten Gerät sicherstellen, seine eventuell von einem fremden Gerät gesperrte PIN wieder zu aktivieren. Hierbei könnte die Sperrung durch ein schlechtes Bedienungskonzept, eine fremde Sprache, schlechte oder manipulierte Hardware oder Funkproblemen durch schlechte Antennen verursacht sein. Im mobilen Nutzungsszenario an einem Automaten ist es außerdem möglich, dass dieser die Prozedur der

Freischaltens der PIN nach Fehleingaben nicht beherrscht oder das ein eventuell notwendiges Neusetzen der PIN mit Hilfe der PUK unabhängig vom Zielgerät an einem völlig anderen Ort (z.B. zu Hause) mit vertraulichem Zugang zum PUK Brief stattfinden soll.

Über diese Wiederherstellung oder Erlangung von gewünschter Funktionalität hinaus, kann ebenso eine mögliche Erhöhung der Sicherheit (siehe Abschnitt 4.4 auf Seite 34) durch die Nutzung des Telefons als Terminal erreicht werden. Unter der Vorraussetzung einer sicheren Telefonarchitektur lassen sich verschiedene Anwendungen einer telefonbasierten PIN-Verwaltung realisieren.

Szenario PIN Speicher Menschen sind vergesslich und neigen dazu nicht zufällige PINs zu nutzen sowie diese mit persönlichen Daten zu korrelieren oder bekannte Muster oder komplette Geheimnisse für verschiedene Dienste wiederzuverwenden. Die Anzahl der relativ häufig und hoch priorisierten Passwörter nimmt hierbei immer mehr zu. Es wird daher für neue Anwärter immer schwieriger akzeptiert zu werden. Neben der EC-Karten PIN, der Geräte-PIN des Telefons, der PIN der SIM-Karte, zum Onlinebankzugang, für die Haustür und zum Öffnen des Autos stellt nun also der Zugang zur digitalen Identität eine neue und sechsstellige Kombination dar, die möglichst im Kopf behalten werden soll. Eine Möglichkeit, um dieses Problem zu entschärfen ist die Verkettung von PIN Legitimationen, wobei es sinnvoll ist mit einer möglichst starken PIN weitere weniger starke abzusichern. Eine Speicherung der eID PIN auf dem Telefon kann verschlüsselt entweder auf dem Gerät, auf der SIM Karte oder auf einem zusätzlichen sicheren Element erfolgen und wäre damit durch die meist vierstelligen numerischen Telefon-PIN's zugriffsgeschützt.

Diese Speicherung im Einsatzszenario als *Standardleser* oder *mobiler Bürgerclient* kann dazu beitragen, dass starke Passwörter genutzt werden und das die Eingabe der eID PIN an einem potentiell unsicheren Ort nicht optisch beobachtet werden kann. Im Gegenzug verschlechtert sich die Bindung von einem Authentifizierungsvorgang an das Wissen des Geheimnisses, da jemand sowohl das Telefon als auch den Personalausweis nach Diebstahl fremdnutzen kann und die den Ausweisinhaber bindende PIN nicht pro Vorgang sondern meistens nur beim Starten des Gerätes abgefragt wird. Als Schutz sollte für dieses Risiko eine Abfrage der SIM/Geräte/SE PIN bei jedem Zugriff umgesetzt werden. Bei entsprechenden Fehleingaben sollte der Speicher gesperrt werden.

Szenario Einmal-PIN Um das Abhören der PIN während der Nutzung der eID Funktion ins Leere laufen zu lassen und eine mögliche Wiedererkennung des Inhabers einer bestimmten PIN durch das Terminal zu verhindern, wäre es möglich einen Mechanismus zu entwerfen, der dafür sorgt, dass nach jeder Nutzung sofort eine PIN Änderung durchgeführt wird. Eine manuelle Än-

derung wäre einem Anwender aus Gründen des Nutzungskomforts kaum zuzumuten und daher bietet sich auch hier die spezielle Verwendung des Mobiltelefons als mobiler Bürgerclient an. Nach jeder Nutzung der eID Funktion mit einer PIN könnte automatisch eine PIN Änderung auf eine neue zufällig gewählte PIN erfolgen, die dann wiederum an einem sicheren Speicherort des Telefons wie zum Beispiel der USIM Karte abgelegt wird. Der Zugriff auf diesen Speicher könnte mit einer separaten Meta-PIN mit dem rechtmäßigen Nutzer verknüpft werden oder implizit wie im PIN-Speicher Szenario mit der SIM Karten PIN geschützt werden. Im Unterschied zum PIN-Speicher kann die Einmal-PIN so gestaltet werden, dass die tatsächlich aktuelle PIN niemals das Telefon verlassen kann und damit niemandem bekannt ist. Man könnte also mit Hilfe eines sicheren Speicherelements dafür sorgen, dass die PIN nur durch Angriffe zur Laufzeit einer Authentisierung auf dem Gerät angegriffen werden kann. Nachteilig bei dieser Lösung ist die ausschließliche Festlegung auf das Telefon als Ausweislesegerät, da die PIN nicht mehr manuell in anderen Lesegeräten eingegeben werden kann. Die 2-Faktor Authentifizierung aus Wissen (PIN) und Besitz (Ausweis) würde sich erweitern um einen zusätzlichen Besitz: das Mobiltelefon. Aus Angreifersicht wäre dies also ein erhöhter Aufwand, aber aus Nutzersicht ergibt sich auch die Abhängigkeit, dass bei Verlust der SIM Karte oder des Masterpassworts auch der Ausweis erneuert bzw. zurückgesetzt werden muss, was nur durch realen persönlichen Zugang zu einem hoheitlichen Terminal möglich ist.

5.3.2 Implementierung

Wie in Kapitel 3.5.1 der Technischen Richtlinie 3110 [9] dargestellt, gibt es für nicht authentisierte Terminals drei mögliche Operationen zur Verwaltung der PIN: `change`, `resume` und `unblock`. Diese werden in 5.3.2 inklusive der notwendige Paßwörter und Protokollschritte dargestellt.

Funktion	Paßwort	Protokollschritte
`change`	PIN alt, PIN neu	PACE mit PIN alt, RRC mit Daten
`resume`	CAN, PIN	PACE mit CAN, PACE mit PIN
`unblock`	PUK	PACE mit PUK, RRC ohne Daten

Tabelle 6: PIN Management Funktionen

Die drei Grundfunktionen der PIN Verwaltung sind separat im Prototypen umgesetzt, um die Vorraussetzungen für die Anwendung in einem der beschriebenen Szenarien zu testen und zu verifizieren. Erst mit einer vollständigen eID Funktionalität auf dem Telefon lassen sich die Szenarien komplett umsetzen und bleiben daher Objekte zukünftiger Arbeit. Es fehlt zum aktuellen

Zeitpunkt noch die Testgegenstelle mit gültigem Berechtigungszertifikat.

Bei den PIN Operationen werden verschiedene PACE Durchläufe durchgeführt die im Detail inklusiver der ausgetauschten APDUs bereits in 5.2.4 beschrieben sind. Spezifisch für diese Anwendung wird die APDU `Reset Retry Counter (RRC)` erstellt. Die dazugehörigen APDU's werden in den Tabellen 5.3.2 und 5.3.2 dargestellt. Hierbei wird der logische Ablauf innerhalb des Programms berücksichtigt. Zunächst wird eine ungesicherte Kommando-APDU RRC erstellt, die wird das an das SecureMessaging (SM) Protokoll übergeben und entsprechend in gesicherte APDU transformiert und dann zum Ausweis geschickt. Als Ergebnis kommt über den SM Kanal wieder eine gesicherte Antwort-APDU, welche von der SM Schicht verfiziert und entschlüsselt wird. Aus welchen Komponenten die beschriebenen APDU Tupel bestehen wird erläutert in Abschnitt 5.2.4 auf Seite 39. Die Werte sind als hexadezimale Beschreibung von Bytefolgen zu interpretieren und für den Fall des Nichtvorhandenseins mit `null` gekennzeichnet.

Beschreibung	Header	Body	Trailer
RRC	(00, 2c, 02, 03)	(06, 31 32 33 34 35 36, null)	-
RRC SM	(0c, 2c, 02, 03)	(1d, 87 11 01 4e 47 04 ae 93 25 dc 79 00 93 98 c0 bc 25 a3 f8, null, 8e 08 8e a3 d9 db 4b 34 43 89, 00)	-
PCD $\Longleftrightarrow$ PICC			
Response SM	-	(null, 99 02 90 00, 8e 08 b0 4f b6 c7 26 6f 7a 16)	(90, 11)
Response	-	(null)	(90, 00)
$K_{Mac} =$ `FFA5F6E2ACEE0D08ED00876ABFEBBA88`			
$K_{Enc} =$ `8B2B15520EEFFE3BA45F304E1D88552F`			

Tabelle 7: APDU Kommunikation für PIN `change`

Beschreibung	Header	Body	Trailer
RRC	(00, 2c, 03, 03)	(null, null, null)	-
RRC SM	(0c, 2c, 03, 03)	(0a , null, null, 8e 08 0e 94 b0 83 ae 49 06 1b, 00)	-
PCD $\Longleftrightarrow$ PICC			
Response SM	-	(null, 99 02 90 00, 8e 08 b0 4f b6 c7 26 6f 7a 16)	(90, 11)
Response	-	(null)	(90, 00)
$K_{Mac} =$ FFA5F6E2ACEE0D08ED00876ABFEBBA88			
$K_{Enc} =$ 8B2B15520EEFFE3BA45F304E1D88552F			

Tabelle 8: APDU Kommunikation für PIN `unblock`

Die Klasse `de.hub.sar.mopa.apps.pin.PinManagement` stellt die drei angestrebten Grundfunktionalitäten zur PIN Verwaltung zur Verfügung und kann in einen mobilen Bürgerclienten oder andere mobile Ausweisanwendungen integriert werden.

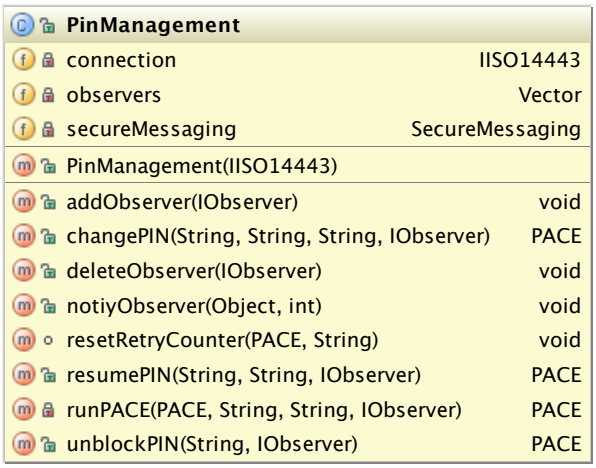

Abbildung 8: Klassendiagramm der Klasse `PinManagement`

5.4 Aktualisierung der Zeit mit einer aktuellen Zertifikatskette

5.4.1 Motivation

Der Chip des Ausweises enthält keine Batterie und auch keine eigene Uhr oder andere Mechanismen, um unabhängig von der eID Nutzung die aktuelle Zeit festzustellen. Für die Nutzung zum Beispiel des Langwellenzeitsignals DCF77 reicht die RFID induzierte Stromversorgung für einen Empfänger nicht aus. Daher arbeitet der Ausweis aktuell mit einem Näherungsverfahren

basierend auf dem spätesten Ausstellungsdatum eines Zertifikates in der Berechtigungszertifikatskette des authorisierten Terminals.

Die Uhrzeit wird abgebildet als monoton wachsende Folge von Zeitpunkten mit dem Datum der Personalisierung des Ausweisexemplares als Startwert. Dieses Verfahren schließt nicht aus, dass ein ausversehen oder absichtlich falsch ausgestelltes Zertifikat die Uhrzeit soweit in die Zukunft versetzt, dass anschließend alle Zertifikate ungültig werden. Ob der Ausweis gegen zu große Zeitsprünge eine Plausibilitätsprüfung durchführt ist der Spezifikation nicht zu entnehmen. Da die Änderung aber nur durch gültige Zertifikate durchgeführt wird, kann man davon ausgehen, dass ein derartiger Angriff bedeuten würde, dass die komplette PKI gebrochen oder fehlerhaft wäre. Der Ausweis muss dem DV Zertifikat vertrauen und bei der Übergabe der Zertifikatskette wird nicht nur die Zeit sondern insgesamt die aktuelle Zertifikatskette gespeichert.

Während in der [9, Kapitel 2.2.5] noch allgemein von gültigen Zertifikaten als Quelle aktueller Zeit gesprochen wird, ist in Kapitel 5.2.3 der [7] nur von hoheitlichen Zertifikaten die Rede, daher bleibt die Annahme, dass es sich tatsächlich um hoheitliche Zertifikate handeln muss.

Ausweise können „auf Grund der verarbeitungsspezifischen Engpässe“ [10, Kapitel 3.3.6] keine Rückruflisten verarbeiten und verwenden daher Berechtigungszertifikate mit kurzer Laufzeit (3 Tage), um eine impliziten Widerruf zu gewährleisten durch das Verweigern der Ausstellung von Verlängerungszertifikaten. Um diese laufzeitbasierte Zertifikatswiderruf zu prüfen, ist der Ausweis zwingend auf die möglichst genaue aktuelle Uhrzeit (inkl. Datum) angewiesen, die dann gegen das Ablaufdatum des Zertifikats geprüft wird. Da durch das beschriebene Zeitaktualiserungsverfahren die Uhrzeit abhängig ist von der Nutzung ergibt sich hier eine Angriffsmöglichkeit auf das Widerrufsverfahren. Bei einem in der Vergangenheit liegenden aktuellem Datum des Ausweises könne alle seit diesem Datum ausgestellten Zertifikatsketten zur Terminalauthentisierung herangezogen werden. Je seltener ein Ausweis benutzt wird, desto größer wird dieses Zeitfenster. Ein Dienst der irgendwann ein gültiges Berechtigungszertifikat für sein Terminal bekommt, hat also gute Chancen über einen relativ großen Zeitraum hinweg immer wieder Ausweise mit den einmal erteilten Rechten auslesen zu können.

Eine mögliche Lösung für dieses Problem stellt ein nutzungsunabhängiger Zeitaktualiserungsdienst auf einem möglichst mobilen und vom eigentlichen Zielterminal separiertem Gerät dar. Für diesen Dienst stellt das Mobiltelefon die ideale Umgebung dar. Auch die beschriebenen Nachteile einer Softwareterminallösung auf einem Telefon gelten für diese Anwendung nicht, da hier eine vom Vertrauensanker signierte Zertifikatskette zum Ausweis transportiert werden soll. Dafür würde das Telefon einen PACE Durchlauf mit dem CHAT zur Ankündigung einer TA durchführen und nach erfolgreicher Übergabe der Zertifikatskette im sicheren Kanal zwischen Ausweis und Telefon das Protokoll abbrechen. Bei entsprechender Gestaltung gibt es bei diesem Prozess keine Angriffsmöglichkeit. Ungültige Zertifikate würden vom Ausweis verworfen und

manipulierte PACE Protokollabläufe sind ausgeschlossen.

Um die Missbrauchsmöglichkeiten bei einem derartigen Dienst weiter zu begrenzen, sollten spezielle Zeitaktualisierungszertifikate ausgegeben werden, in denen keine Rechte zum Lesen oder Schreiben von Ausweisdaten enthalten sind. außerdem sollte der Gültigkeitszeitraum möglichst klein gewählt werden (z.B. drei Stunden), um eine möglichst präzise Zeit im Ausweis zu setzen. Da nach obiger Angabe nur hoheitliche Zertifikate in Frage kommen, muss dieser Zeitdienst von einer hoheitlichen Stelle betrieben werden und das Zertifikat möglichst frei zur Verfügung stehen. Da das Zertifikat in sich ein nicht manipulierbares Dokument darstellt, könnte auf eine Absicherung der Transportschicht sogar verzichtet werden und ein einfacher http GET request könnte bereits das Zertifikat abholen. Ein derartiges Protokoll ließe sich auch gut in einer mobilen Ausweissoftware integrieren.

Das ein derartiges Vorgehen sinnvoll ist und auch geplant ist lässt sich der [7, Kapitel 5.2.3] entnehmen:

„Um dem Ausweisinhaber die Möglichkeit zu geben, das angenäherte Datum seines Ausweises zu aktualisieren, kann er das Datum über ein online zur Verfügung gestelltes tagesaktuelles Zertifikat vom Typ hoheitliches nationales Authentisierungsterminal aktualisieren. Um das angenäherte Datum zu aktualisieren, ist die Durchführung der Terminalauthentisierung nicht notwendig, das Übermitteln der Zertifikatskette an den Chip ist ausreichend. Da die Terminalauthentisierung nicht durchgeführt wird, werden auch keine Zugriffsrechte vergeben.“

5.4.2 Implementierung

Bei der Verwendung eines hoheitlichen Zertifikats zur Zeitaktualisierung bietet sich die Möglichkeit, diese mit einem CHAT eines Inspektionsterminals durchzuführen. Für einen erfolgreichen PACE-Durchlauf und die Einleitung der TA reicht dann als Passwort die CAN aus, welche permanent im Ausweis gespeichert werden kann. Diese stellt als ablesbarer Wert ein schwächeres und nicht personengebundendes Geheimnis dar. Wenn die Ausweise irgendwann ein Display bekommen auf dem dynamische CANs angezeigt werden, verfällt der Vorteil der permanenten Speicherung. Zusätzlich zur CAN kann die Uhrzeit des letztes Zertifikats zwischengespeichert werden, so dass weitere Updates nur gemacht werden, wenn notwendig.

Die komplette Funktion zum Zeitupdate kann als separate Aufgabe im moPA Prototypen oder später auch integriert in eine vollständige mobile Bürgerclientversion eingesetzt werden. Vor jeder Aktion, die dann eine Rückrufprüfung eines Zertifikats beinhaltet, ist ein Zeitupdate möglich und wie oben beschrieben aus Sicherheitsgründen auch zu empfehlen.

Auch wenn ein spezieller Zeitaktualisierungsdienst vom BSI bereits beschrieben und geplant ist, wurde dieser leider im Rahmen der Bearbeitungszeit dieser Arbeit nicht zur Verfügung gestellt. Es ist daher kein Test möglich und damit auch keine abschließende Implementierung zu

dieser Idee. Zur Umsetzung vorbereitet ist der erste Schritt der Terminalauthentisierung in dem das Zertifikat an den Ausweis übermittelt wird. Der dazugehörige Programmteil befindet sich in der Klasse `de.hub.sar.mopa.apps.id.TimeUpdate`.

5.5 Authorisierungsprüfer

5.5.1 Motivation

Wichtiger Bestandteil der neuen Ausweisinfrastruktur und der wechselseitigen Authentifizierung der Teilnehmer ist der mittels Zertifikat verbriefte Nachweis von Zugriffsrechten durch das Terminal. Insbesondere in einem mobilen Nutzungsszenario von integrierten Terminals (Automaten) ist der Ausweisanwender zunächst auf augenscheinliche Merkmale angewiesen, die ihn dazu bewegen den Ausweis tatsächlich mit dem Lesegerät des Automaten zu verbinden und dann auch noch zusätzlichen die eID PIN in den Automaten einzugeben um z.B. eine Altersverifikation durchzuführen. Um diesen Vertrauensvorschuss technisch abzusichern, lässt sich eine neue Funktion auf dem Mobiltelefon nutzen.

Das Telefon simuliert dabei über die NFC Schnittstelle einen Personalausweis und spricht mit dem Automaten das Protokoll bis zum Beginn des PACE Durchlaufs. Ein Bestandteil des „Manage security environment: Set Authentication Template“ (MSE:SetAT) Paketes zu Beginn des PACE Protokolls ist die Bekanntgabe des Terminaltyps und der angestrebten Zugriffsrechte. Dieses Certificate Holder Authorization Template (CHAT) muss übertragen werden, falls das Terminal im weiteren Verlauf eine der Ausweisapplikationen als authorisiertes Terminal verwenden möchte. Der CHAT wird abgeleitet aus dem CHAT des Berechtigungszertifikats und die Rechte können vom Ausweisinhaber über eine GUI eingeschränkt werden. Da diese GUI nur am Automaten existiert, kann der Nutzer nicht sicher sein, das die angezeigten mit den anschließend übermittelten Daten übereinstimmen.

Ein Telefon als simulierter Ausweis ist in der Lage dazu den tatsächlich übermittelten CHAT anzuzeigen und damit vor dem Einsatz des nPA die korrekte Funktion des Terminals zu prüfen.

5.5.2 Implementierung

In der Klasse `de.hub.sar.mopa.apps.id.ChatCatcher` befindet sich der erste Prototyp zu der hier entworfenen Idee. Zur Vereinfachung beantwortet das Programm die APDU's bis zum MSE:SetAt mit aufgezeichneten Werten aus dem aktuellen Testausweis. Wenn das MSE:SetAT empfangen wurde, wird zunächst entsprechend der Struktur in 5.5.2 der CHAT extrahiert.

Pos.	Länge	Wert	Beschreibung
1	1	06	Tag für Objektidentifikator der Rolle
2	1	09	Länge des Objektidentifikators der Rolle
3	9	04 00 7F 00 07 03 01 02 RR	Objektidentifikator der Rolle: RR = 01: Hoheitliches Terminal RR = 02: Authentifikationsterminal RR = 03: Signaturterminal
4	1	53	Tag für Zugriffsrechte
5	1	XX	Länge der Zugriffsrechte (01, 05 oder 01)
6	var.	XX .. XX	Zugriffsrechte

Tabelle 9: Struktur des CHAT Datenobjektes (aus [11, Tabelle 27])

Für die eID Anwendung bestehen die Zugriffsrechte im CHAT aus fünf Bytes entsprechend der Bitmatrix in [9, Tabelle C.4]. Im Prototypen werden diese einfach als Text entsprechend angezeigt. Hier besteht aus Sicht der Bedienbarkeit viel Verbesserungspotential. Durch die hohe Anzahl der verschiedene Rechtebeschreibungen wäre eine auf kleine Displays optimierte Symbolik wünschenswert.

Die Implementierung konnte leider aus zeitlichen Gründen nicht vollständig abgeschlossen werden und auch nicht gegen ein tatsächliches integriertes Terminal getestet werden. Da für den Funktionstest hier keine „echte“ EAC PKI notwendig ist, kann eine direkte Kommunikation zwischen zwei parallel laufenden Instanzen der moPA Implementierung jeweils eine im PICC Modus und eine im PCD Modus als Test ausgeführt werden. Um dieses Verfahren noch generischer und unabhängig von Plattformen und Implementierungssprachen zu machen, ist eine APDU Kommunikation über einen TCP/IP Socket zwischen Softwarekarte und Softwareterminal möglich.

5.6 Ergebnis

Im folgenden werden die Ergebnisse aus den praktischen Erfahrungen mit der Anwendungsentwicklung für das Nokia 6212 und den Tests des Prototypen mit dem neuen Personalausweis beschrieben. Zunächst bleibt festzuhalten, dass das ursprüngliche Ziel im praktischen Teil dieser Arbeit erreicht wurde. Der Prototyp ist auf dem Mobiltelefon lauffähig und kann erfolgreich mit dem Ausweis kommunizieren und zum Beispiel die PIN des Ausweises ändern. Bei der Umsetzung des Prototyps sind viele Probleme aufgetaucht, die vor allem begründet sind in der Architektur des Ausweis als geschlossenes System, dass aus Sicherheitsgründen (Verhinderung von Laufzeitanalysen) im initialen Verbindungsaufbau während des PACE Protokolls keine detaillierten Fehlermeldungen ausgibt, sondern das Protokoll teilweise bis zum Ende durchläuft

und mit allgemeinen Fehlern wie `0x6A80` (Falsche Parameter im Datenfeld) oder `0x6300` (Authentifizierung fehlgeschlagen) beendet wird. Es gibt kein detailliertes Entwicklerhandbuch und die Spezifikationen lassen manchmal auf dem Bytelevel Interpretationsspielräume, die man ohne einen erfolgreichen Referenzlauf schwer korrigieren kann. Hilfreich bei der Entwicklung war die vorhandene und mit Quelltext verfügbare Implementierung OpenPACE [24]. Aus dieser konnten Referenz-APDUs gewonnen werden für Unittests der Javaentwicklung.

Zusätzlich erschwerend ist die Instabilität der Programmabläufe auf dem Zielgerät, die erst mit zahlreichen Tests auf die Hardware zurückgeführt werden konnte. Es kommt in unregelmäßigen Abständen zu Fehlern, die vor allem auf Verbindungsabbrüche des NFC Funkkanals zurückzuführen sind. Der NFC Chip des Nokia 6212 ist für die Nutzung als Austauschkanal von Visitenkarten und zum Auslesen von RFID Tags ausgelegt. Dementsprechend sind die Antennen umlaufend am Gehäuserand platziert und grade ausreichend um eine kurzzeitige Verbindung aufzubauen. Der Ausweis erfordert nun aber stabile Verbindungen für die eID Anwendung die im Falle eines möglichen mobilen Bürgerclienten auch mehrere Minuten erhalten bleiben müsste. Dies ist mit dem Nokiagerät nicht möglich. Die intuitive Bedienung durch Legen des Ausweises in die Hand, dann darauf das Telefon und mit der anderen Hand erfolgt die Bedienung der Tastatur zur PIN und Dateneingabe funktioniert leider nicht. Für den Aufbau der NFC Verbindung muss der Ausweischip auf dem oberen Gehäuserand platziert werden und verdeckt dadurch entweder Tastatur oder Display. Um den Prototypen trotzdem lauffähig zu bekommen, wurden der Prozess in der GUI so gestaltet, dass erst alle Eingaben erfolgen und dann der Ausweis über die Telefontastatur gelegt werden kann. Für eine praktische Massenanwendung scheidet das Gerät dadurch aus.

Ein weitere Aspekt zur Bedienerfreundlichkeit der Mobiltelefon + Ausweis Kombination ist die zu langandauernde Laufzeit der Protokollschritte, welche bei der Nutzung verunsichert. Da die Performanz ein zusätzliches k.o. Kriterium für das eingesetzte Gerät und die Implementierung darstellt, wird diese noch detaillierter analysiert.

5.6.1 Analyse der Laufzeiten

In [17] sind bereits Messwerte zur Protokolllaufzeit von PACE auf dem identischen Gerät beschrieben. Um diese mit der hier vorliegenden Implementierung zu vergleichen, werden die Messpunkte (Transaktionen) identisch strukturiert. Zusätzlich wird der Protokollschritt der PIN Änderung gemessen und es werden Transaktionen als 3-Tupel von Schritten zur Vorbereitung, Kommunikation und Nachbereitung definiert, um die Laufzeiten der reinen APDU Kommunikation von der softwareseitigen Verarbeitungsgeschwindigkeit abzugrenzen.

[24] http://sourceforge.net/projects/openpace/

Schritt $S \models V \parallel K \parallel N$

Transaktion $T \models n * (V, K, N)$

Protokoll $P \models n * T$

Applikation $A \models n * P$

Die einzelnen Transaktionen der Applikation `PIN-change` sind wie folgt definiert:

1. Lesen der PACE Parameter
   - 1.1 Selektieren von MasterFile und EF.CardAccess
   - 1.2 Auslesen der EF.CardAccess
   - 1.3 Erstellen der `SecurityInfos`
2. PACE Protokoll
   - 2.1 MSE:SetAt
   - 2.2 Get Encrypted Nonce
   - 2.3 Map Nonce
   - 2.4 Key Agreement
   - 2.5 Mutual Authentication
3. Pin Management
   - 3.1 ResetRetryCounter

Die Testläufe wurden manuell auf dem Nokia 6212 jeweils mit einem frisch gestarteten Programm und leerem Log durchgeführt. Der Mehraufwand durch die Messung und detaillierte Protokollierung der einzelnen Schritte liegt bei 15% und wurde über die Dauer der Gesamtlaufzeit eines Testlauf einmal mit und einmal ohne Protokoll näherungsweise ermittelt. In der Tabelle 10 auf Seite 55 werden die Messergebnisse in Millisekunden jeweils der drei Schritte pro Transaktion für 10 Messungen dargestellt. Das arithmetische Mittel der Gesamtlaufzeit einer PIN Änderung auf dem Telefon aus 10 Testläufen liegt bei 41,7 Sekunden mit einer Standardabweichung von 0,52. Ohne die letzte Transaktion der eigentlichen PIN Änderung liegt der Durchschnitt bei 41,1 Sekunden und ist damit nur geringfügig besser als in [17, Kapitel 5.4]. Verschiedene Versuche in der Implementierungsphase konnten hier keine signifikanten Verbesserungen erzielen. Auch ein Profiling der Applikation inklusive Speicher und CPU Hotspotanalyse brachte keine weiteren Vorteile. Die sehr häufig auftretenden Bytearray Operationen und auch die Kryptobibliothek wurden noch mal überarbeitet und auch verschiedene AesEngines aus dem bouncycastle Paket getestet. Aufgrund der dürftigen Ergebnisse aus den Verbesserungsmassnahmen im

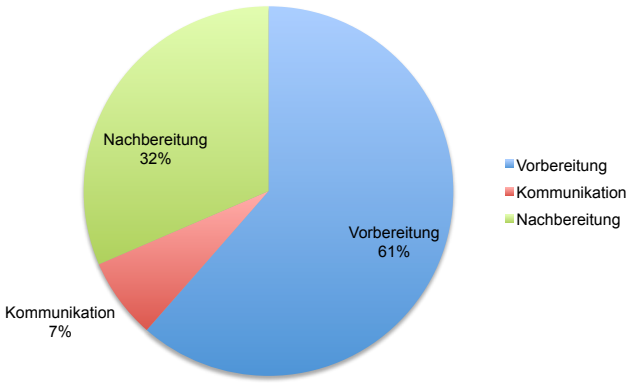

Abbildung 9: Verteilung der Laufzeiten / Schrittkategorie

Programmquelltext lässt sich der Schluss ziehen, dass man mit der Kombination von J2ME und Nokia 6212 an Hardwarelimitierungen angekommen ist.

Entgegen den ersten Vermutungen das hierbei vor allem die schlechten Antennen für eine sehr langsame NFC Verbindung sorgen und die Zeit dabei verbraucht wird, zeigt die Betrachtung der Verteilung der Laufzeit auf die Kategorien der Transaktionsschritte (9), dass lediglich 7% der Laufzeit auf die APDU Kommunikation entfallen. Dieser Wert enthält neben der reinen Kommunikation auch noch die komplette Verarbeitungszeit auf dem Ausweis. Das heißt, dass die Chipkartenoperationen im Vergleich zur Softwareimplementierung auf dem Telefon um ein vielfaches schneller ist. Um zu verifizieren, dass hier die Hardware der limitierende Faktor ist und nicht die Implementierung wurden die Tests auch noch mal auf dem Entwicklungsrechner mit externem Lesegerät durchgeführt. Hier wird eine durchschnittliche Gesamtlaufzeit für die PIN Änderung von lediglich 1,4 Sekunden erreicht. Es also deutlich herauszulesen, dass hier die Hardware des Telefons mit den Kernfaktoren CPU- und Speichergeschwindigkeit eine nutzerfreundliche Abarbeitungszeit verhindern.

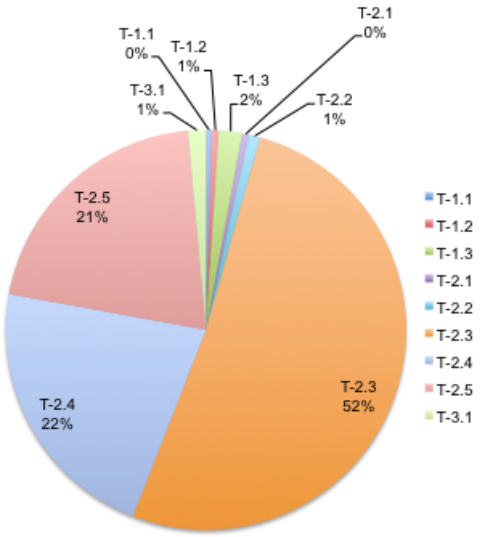

Abbildung 10: Verteilung der Laufzeiten pro Transaktion

In den Laufzeiten der einzelnen Transaktionen (Abb. 10) sieht man vor allem $T_{2.3}$ mit einem Anteil von 52% der Gesamtlaufzeit herausstechen. Hier wird in der Vorbereitungsphase das EC-Schlüsselpaar erzeugt und die ca. 8s Laufzeit werden dabei vermutlich zum größten Teil für die Nutzung des Zufallszahlengenerators verbraucht. In der Nachbereitungsphase werden dann relativ viele kryptografische Operationen mit elliptischen Kurven beim Erzeugen des ephemeralen Schlüsselpaares durchgeführt und ca. 12s dafür verbraucht. Das lässt den Schluss zu, dass vor allem die komplexen kryptografischen Operationen das Mobilgerät überfordern. Da die verwendete Nokia Platform aber schon heute aus Perspektive der aktuellen Smartphonegenerationen als veraltet angesehen werden kann, ist es dringend erforderlich bei dem nächsten „modernen“ verfügbaren Gerät mit NFC Chip diese Anwendung neu zu testen. Sehr wahrscheinlich spielen die Performanzprobleme dann nur noch eine untergeordnete Rolle.

	Test-0	Test-1	Test-2	Test-3	Test-4	Test-5	Test-6	Test-7	Test-8	Test-9	Ø	σ
1.1v	71	64	71	60	66	72	70	70	69	63	68	4
1.1k	139	118	129	118	139	143	98	140	141	139	130	15
1.1n	0	0	0	0	0	0	0	0	0	0	0	0
1.2v	50	48	52	40	39	52	38	51	50	50	47	6
1.2k	174	235	171	231	171	171	171	171	173	171	184	26
1.2n	0	0	0	0	0	0	0	0	0	0	0	0
1.3v	0	0	0	0	0	0	0	0	0	0	0	0
1.3k	0	0	0	0	0	0	0	0	0	0	0	0
1.3n	971	10	979	10	981	977	994	985	1000	978	789	410
2.1v	88	182	85	182	91	90	88	85	83	91	107	40
2.1k	122	121	121	122	127	124	120	121	120	120	122	2
2.1n	0	1	0	0	0	0	0	0	0	0	0	0
2.2v	104	29	27	33	105	102	103	100	101	101	81	35
2.2k	268	269	270	269	268	268	268	267	266	266	268	1
2.2n	41	2	40	2	41	40	40	41	40	40	33	16
2.3v	8213	8259	8400	8262	8297	8507	8423	8450	8390	8338	8354	95
2.3k	1031	1031	417	1031	1031	408	1032	1032	361	1031	841	308
2.3n	12091	12150	12314	12420	12201	12514	12456	12345	12335	12224	12305	137
2.4v	8305	8229	8504	8350	8385	8236	8182	8352	8377	8426	8335	98
2.4k	1030	1031	309	1031	1035	1030	299	1036	271	1031	810	357
2.4n	0	0	1	1	0	0	0	0	0	1	0	0
2.5v	8240	8390	8550	8401	8321	8314	8318	8434	8426	8497	8389	93
2.5k	232	231	234	232	232	233	233	232	232	232	232	1
2.5n	0	0	0	0	0	0	0	0	0	0	0	0
3.1v	234	308	230	300	210	232	236	229	239	234	245	32
3.1k	358	358	358	359	357	358	361	358	358	357	358	1
3.1n	1	1	1	1	1	1	2	2	1	1	1	0
Sum.	41763	41067	41263	41455	42098	41872	41532	42501	41033	42391	41698	520

Tabelle 10: Messwerte der Laufzeiten von Transaktionsschritten der *PIN-change* Applikation (in ms)

6 Fazit und Ausblick

Die Verfügbarkeit, der Komfort und das Vertrauen in ein täglich genutztes Gerät waren die grundlegende Motivation, um ein Mobiltelefon als Lesegerät mit dem neuen Personalausweis zu verbinden. Sowohl in der praktischen Umsetzung als auch bei der Betrachtung der Möglichkeiten des Telefons als neue Systemkomponente offenbaren sich Schwierigkeiten, die weiterführende Arbeiten erfordern.

Zunächst war die Entwicklung von Beispielanwendungen zur Nutzung des Nokia 6212 Classic als nichtauthentisiertes Terminal erfolgreich. Der Prototyp ist auf dem Mobiltelefon lauffähig, kann erfolgreich mit dem Ausweis kommunizieren und bietet zum Beispiel die Möglichkeit die eID-PIN des Ausweises zu ändern. Bei der Umsetzung und den anschließenden Testläufen konnten problematisch hohe Ausführungszeiten der kryptografischen Operationen auf dem Gerät nachgewiesen werden. Darüberhinaus kam es durch instabile Funkverbindungen immer wieder zu Protokollabbrüchen während der Ausführung. Beide Beobachtungen begründen den Schluss, dass eine praktische Nutzung des Nokia 6212 zur Anwendung mit dem neuen Personalausweis nicht zu empfehlen ist.

Für das Integrationsszenario des Telefons als Standardlesegerät stellte sich die Gestaltung, der von der TR-3119 geforderten Treiberunterstützung für gängige Betriebssysteme als Herausforderung heraus. Insbesondere in dem angestrebten mobilen Szenario einer Nutzung an unterschiedlichen und potentiell fremden, nicht veränderbaren PC's bleibt die Frage offen, welche Möglichkeiten einer Bluetooth- oder USB-Verbindung bestehen, die auf im Betriebssystem vorhandene Standardtreiber aufsetzen.

Weiteres Problem bleibt die Umsetzung der eCard-API auf dem Telefon mit seinen beschränkten Ressourcen an CPU, Speicher und Netzwerkbandbreite. Mit der Entwicklung einer mobilen Variante der eCard-API könnte in weiteren Arbeitsschritten die Umsetzung eines mobilen Bürgerclienten erfolgen, die eine autonome Verwendung des Telefons als lokales Terminal in der Onlineauthentisierung ermöglicht.

Eine Grundvorraussetzung für die vollwertige Integration des Telefons, zur Erlangung der notwendigen Zertifizierungen und - im Falle eines Terminals - der Berechtigungszertifikate, stellt die Sicherheit des mobilen Betriebsystems dar. Da diese mittlerweile offene Systeme mit zunehmender Annäherung an klassische PC Betriebsysteme sind, muss weitere Arbeit investiert werden in die Umsetzung einer sicheren Plattform, die die Integrität der Anwendungssoftware garantieren kann.

Die Nutzung von PACE und der beschriebenen Infrastruktur auf einem Telefon scheint insgesamt machbar, aber aktuell vor allem sinnvoll zu sein für eine Anwendungsklasse mit weniger hohen Sicherheitsbedarf. Es wäre möglich, das Telefon zu einer mobilen Smartcard zu

entwickeln. Dabei könnte sich das Telefon auf verschiedenen Wegen in die bereits existierende Infrastruktur integrieren. Es wäre möglich, per NFC einen Ausweis zu simulieren und mit einem Lesegerät zu kommunizieren oder auch ein Lesegerät mit eingelegter Smartcard darzustellen, welches über USB mit dem Bürgerclient eines PC's spricht. Diese mobile Smartcard könnte für einfache Zugangssysteme oder Ausweise mit geringem Schadenspotential eingesetzt werden und damit der neuen Technologie zu einer höheren Verbreitung helfen, ohne für den weniger sicherheitskritischen Anwendungsbereich die Aufwände für zertifizierte Lesegeräte und eine Berechtigungs-PKI zu verursachen.

Abkürzungsverzeichnis

ATR	Answer-to-Reset (ISO 7816 Chipkartencharakteristikanzeiger)
BSI	Bundesamtes für Sicherheit in der Informationstechnik
CAN	Card Access Number
CAuth	Chip-Authentisierung
CHAT	Certificate Authorization Holder Template
EACv2	Extended Access Control in Version 2
ECC	Elliptische-Kurven-Kryptographie
eID	Elektronischer Identitätsnachweis (Anwendung im nPA)
EuCC	European Citizen Card
ICAO	International Civil Aviation Organization
MAC	Message Authentication Code
MRTD	Machine Readable Travel Document
nPA	Neuer elektronischer Personalausweis
PACE	Password Authenticated Connection Establishement
PCD	Proximity Coupling Device
PICC	Proximity Integrated Circuit Card
PIN	Personal Identification Number
PK	Public Key
PKI	Public Key Infrastructure
PUK	Personal Unblocking Key
RFID	Radio Frequency Identification
RI	Restricted Identification
SIM	Subscriber Identity Module
SK	Secret Key
TA	Terminalauthentisierung

Literatur

[1] BENDER, JENS, MARC FISCHLIN und DENNIS KÜGLER: *Security Analysis of the PACE Key-Agreement Protocol*. In: *ISC '09: Proceedings of the 12th International Conference on Information Security*, Seiten 33–48, Berlin, Heidelberg, 2009. Springer-Verlag.

[2] BENDER, JENS, DENNIS KÜGLER, MARIAN MARGRAF und INGO NAUMANN: *Sicherheitsmechanismen für kontaktlose Chips im deutschen elektronischen Personalausweis*. Datenschutz und Datensicherheit, 3:173–177, 2008.

[3] BITKOM und ARIS: *Studie: Neuer Personalausweis*. `http://www.bitkom.org/de/publikationen/38338_63570.aspx`, Feb 2010. [Online; letzter Zugriff 20.06.2010].

[4] BSI: *Risiken und Chancen des Einsatzes von RFID-Systemen*. Technischer Bericht, Bundesamt für Sicherheit in der Informationstechnik, 2004.

[5] BSI: *Kryptographische Verfahren: Empfehlungen und Schlüssellängen*. Bundesamt für Sicherheit in der Informationstechnik, 1.0 Auflage, Jun 2008.

[6] BSI: *Anforderungen an Chipkartenleser mit ePA Unterstützung*. Bundesamt für Sicherheit in der Informationstechnik, 1.1 Auflage, 2009.

[7] BSI: *Architektur Elektronischer Personalausweis*. Bundesamt für Sicherheit in der Informationstechnik, 1.10 Auflage, Okt 2009.

[8] BSI: *Elliptic Curve Cryptography*, Band Technical Guideline BSI TR-03111. Bundesamt für Sicherheit in der Informationstechnik, 1.11 Auflage, Apr 2009.

[9] BSI: *Advanced Security Mechanisms for Machine Readable Travel Documents*. Bundesamt für Sicherheit in der Informationstechnik, 2.03 Auflage, März 2010.

[10] BSI: *EAC-PKI'n für den elektronischen Personalausweis*. Bundesamt für Sicherheit in der Informationstechnik, 1.01 Auflage, März 2010.

[11] BSI: *Test plan for eID and eSign compliant eCard reader systems with EAC 2*, Band 2. Bundesamt für Sicherheit in der Informationstechnik, 1.0 Auflage, Mai 2010.

[12] BUNDESREPUBLIK DEUTSCHLAND: *Gesetz über Personalausweise und den elektronischen Identitätsnachweis sowie zur Änderung weiterer Vorschriften*. Bundesgesetzblatt, (33):1346–1359, Jun 2009.

[13] ECKERT, CLAUDIA: *IT-Sicherheit: Konzepte, Verfahren, Protokolle*. Oldenbourg Verlag, Muenchen, 6. Auflage, Jan 2009.

[14] EUROPÄISCHE KOMMISSION: *Progress Report on the Single European Electronic Communications Market*. Technischer Bericht 15, Europäische Kommission für das Europaparlament, Mai 2009.

[15] FEDERRATH, HANNES und ANDREAS PFITZMANN: *Gliederung und Systematisierung von Schutzzielen in IT-Systemen*. Datenschutz und Datensicherheit: DuD, 24(12):704–710, 2000.

[16] FINKE, THOMAS und HARALD KELTE: *Radio Frequency Identification – Abhörmöglichkeiten der Kommunikation zwischen Lesegerät und Transponder am Beispiel eines ISO14443-Systems*. Technischer Bericht, Bundesamt für Sicherheit in der Informationstechnik, 2004.

[17] HORSCH, MORITZ: *MobilePACE - Password Authenticated Connection Establishment implementation on mobile devices*. bachelorthesis, TU Darmstadt, Sep 2009.

[18] HÜHNLEIN, DETLEF und MANUEL BACH: *Die Standards des eCard-API-Frameworks*. Datenschutz und Datensicherheit - DuD, 32(6):379–382, Jun 2008.

[19] HYPPÖNEN, KONSTANTIN: *Open Mobile Identity*. Doktorarbeit, University of Kuopio, März 2009.

[20] ICAO: *ICAO Doc 9303 - Machine Readable Travel Documents - MRtds with Machine Readable Data Stored in Optical Character Recognition Format*, Band 1. International Civil Aviation Organisation, 2008.

[21] ICAO: *ICAO Doc 9303, Machine Readable Travel Documents, Part 3, Specifications for Electronically Enabled MRtds with Biometric Identification Capability*, Band 2. International Civil Aviation Organisation, Mai 2008.

[22] ICAO: *Supplemental Access Control for Machine Readable Travel Documents*. Technischer Bericht 0.94, International Civil Aviation Organisation, Feb 2010.

[23] ISO 14443-4: *Identification cards — Contactless integrated circuit(s) cards — Proximity cards — Part 4: Transmission protocol*. ISO/IEC, 2000.

[24] ISO 7816-1: *Identification cards – Integrated circuit(s) cards with contacts – Part 1: Physical characteristics*. ISO/IEC, 1998.

[25] ISO 7816-4: *Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange*. ISO/IEC, 2005.

[26] KIM, MIJIN, BYUNGHEE LEE, SEUNGJOO KIM und DONGHO WON: *Weaknesses and Improvements of a One-time Password Authentication Scheme*, 2009.

[27] LAMPORT, LESLIE: *Password authentication with insecure communication*. Commun. ACM, 24(11):770–772, 1981.

[28] MERZ, HERBERT: *Bitkom Pressekonferenz: Mobile Kommunikation, Marktanalyse 2010*, Feb 2010.

[29] MITCHELL, CHRIS: *Security for Mobility*. IEEE Press, Piscataway, NJ, USA, 2003.

[30] NAUMANN, INGO und GILES HOGBEN: *Security Issues in the Context of Authentication Using Mobile Devices (Mobile eID)*. position paper, European Network and Information Security Agency, Nov 2008.

[31] RANKL, WOLFGANG und WOLFGANG EFFING: *Handbuch der Chipkarten : Aufbau - Funktionsweise - Einsatz von Smart Cards*. Hanser, München, 5. Auflage, 2008.

[32] ROSSNAGEL, ALEXANDER, GERRIT HORNUNG und CRISTOPH SCHNABEL: *Die Authentisierungsfunktion des elektronischen Personalausweises aus datenschutzrechtlicher Sicht*. Datenschutz und Datensicherheit, 3:168–172, 2008.

[33] SCHMEH, KLAUS: *Elektronische Ausweisdokumente - Grundlagen und Praxisbeispiele*. Hanser, 2009.

[34] SCHNEIER, BRUCE: *Secrets and Lies: Digital Security in a Networked World*. John Wiley & Sons, 2004.