Mobile Authentisierung und Signatur

J. Braun[1] · M. Horsch[1] · A. Wiesmaier[1] · D. Hühnlein[2]

Technische Universität Darmstadt[1]
{jbraun | horsch | wiesmaie}@cdc.informatik.tu-darmstadt.de

ecsec GmbH[2]
detlef.huehnlein@ecsec.de

Zusammenfassung

Mobile Geräte spielen eine immer wichtigere Rolle im täglichen Leben und sind geradezu dazu prädestiniert, das Fundament eines allgegenwärtigen und kostengünstigen Sicherheitselements zu bilden. Darüber hinaus kann ein NFC-fähiges Mobiltelefon als Kartenterminal für kontaktlose Chipkarten, wie z.B. dem neuen Personalausweis, genutzt werden oder selbst eine solche Karte emulieren. Wir stellen innovative Authentisierungsfunktionen für mobile Geräte vor. Zu den Szenarien gehören unter Anderem eine mobile AusweisApp und ein verteilter Komfort-Kartenleser zur Signaturerzeugung.

1 Einleitung

Das Mobiltelefon entwickelt sich immer stärker zum fundamentalen Kommunikationsmedium unserer Gesellschaft und ist heute schon unser alltäglicher Begleiter. Deshalb liegt es nahe, ein Mobiltelefon als Grundlage eines ubiquitär und ergonomisch nutzbaren Sicherheitselements zu begreifen.

Außerdem stellt sich die *Near Field Communication* (NFC) Technologie [ISO04, ISO05a] hier als besonders vielversprechend dar, da ein NFC-fähiges Mobiltelefon als Kartenleser für eine kontaktlose Chipkarte gemäß ISO14443 [ISO01] fungieren oder selbst eine solche Karte emulieren kann. Die ersten NFC-Geräte sind bereits auf dem Markt und das Thema genießt steigende Aufmerksamkeit. Projekte wie *Touch&Travel* und *Google Wallet* zeigen das Potential, das breite Anwendungsspektrum und die Benutzerfreundlichkeit der NFC-Technologie.

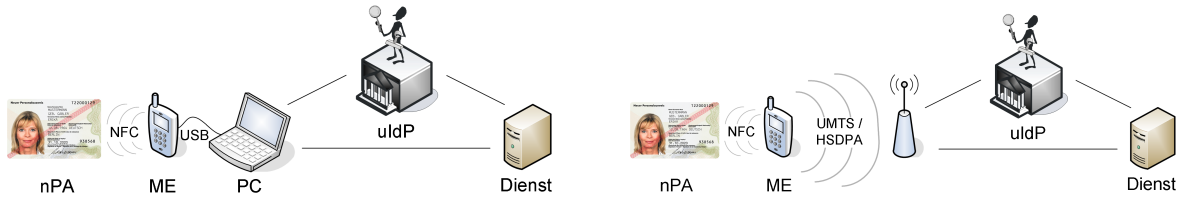
Im Folgenden zeigen wir am Beispiel des neuen Personalausweises (nPA), welche Anwendungen wir für mobile Authentisierung und Signatur gerade entwickeln und wie diese funktionieren. Ein universeller Authentisierungsdienst (uIdP) dient uns als eID-Server gemäß [BSI10b] und unterstützt als Server-basiertes Modul des verteilten Komfort-Chipkartenlesers (vCat-K) die verteilte Verschlüsselung der Signatur-PIN.

2 Mobiler Kartenleser und mobile eID-Applikation

Eine wesentliche Voraussetzung für die breite Nutzung des nPA ist die Verfügbarkeit entsprechender Chipkartenterminals. Trotz aufwändiger Subventionen [BBI10] ist bis auf Weiteres nicht davon auszugehen, dass überall entsprechende Kartenleser verfügbar sein werden. Auf der anderen Seite gehen Analysten davon aus, dass zukünftig viele Mobiltelefone mit der NFC-Technologie ausgestattet werden [JuR11], so dass sich die Nutzung dieser Telefone als Chipkartenterminal anbietet. NFC-fähige Mobiltelefone können daher ein entscheidender Kata-

P. Schartner · J. Taeger (Hrsg.) · D•A•CH Security 2011 · syssec (2011) 1-12.

lysator für die breite Nutzung der eID-Funktion des Ausweises sein. Der große Nutzen liegt insbesondere in dem flexiblen Einsatz des Handys. Bei stationären Rechnern kann ein NFC-fähiges Gerät als Basis-Kartenleser gemäß [BSI11, Abschnitt 4] für den nPA genutzt werden. Das Gerät fungiert dabei als Schnittstelle zwischen Rechner und Ausweis (siehe Abb. 1). Die Verbindung zum Rechner kann bspw. via WLAN, Bluetooth oder USB geschehen. Mit der in [Hors09] entwickelten mobilen *PACE* (Password Authenticated Connection Establishment) Implementierung und der *NFCBTPCSC* (Nokia NFC Bluetooth PCSC Reader) [Beil09] Bibliothek lässt sich auf Basis des Nokia 6212 bspw. ein mobiler Kartenleser realisieren. Die PIN-Eingabe erfolgt dabei direkt auf dem Handy.



(a) Mobiler Basis-Chipkartenleser (b) Mobile eID-Applikation

Abb. 1: Mobiles Endgerät (ME) als Kartenleser

Um jedoch die eID-Funktion vom nPA bei mobiler Internetnutzung verwenden zu können, bedarf es einer eID-Applikation für mobile Geräte. Ein solches Pendant zur AusweisApp haben wir bereits in Zusammenarbeit mit den *T-Labs*, der *T-Systems* und der *media transfer AG* entwickelt [Hors11]. Unsere Java ME Anwendung erlaubt eine Authentisierung mit dem nPA gegenüber einer Web-Anwendung. Die Anwendung wurde erfolgreich auf dem Nokia 6212 umgesetzt und bemisst 424 KB. Die Laufzeit beträgt ohne Benutzerinteraktion auf dem 2008 erschienenen und ressourcen-beschränkten Handy 26,3 Sekunden für einen vollständigen Durchlauf. Dabei entfallen 5,4 Sek. auf den nPA und 6,2 Sek. auf die Internetkommunikation, was allein 44 % der Laufzeit entspricht. Der TLS-Verbindungsausbau ist mit 2,5 Sek., PACE 7,8 Sek., EAC 4,3 Sek. und das Auslesen der Daten mit 1,6 Sek. zu bemessen. Dabei lassen sich wie in [WHBK+11] gezeigt zusätzliche Optimierungen umsetzen. Bei ersten Messungen mit modernen Android-Geräten und schnellen Mobilfunknetzen konnten weitere Reduzierungen festgestellt werden, so dass auf aktuellen Geräten von einer Gesamtlaufzeit von 15 Sek. auszugehen ist.

3 Aktives Authentisierungstoken

Für Anwendungen denen bei jeder Authentisierung die, wie bei dem nPA, obligatorische PIN-Eingabe eher störend wäre, kann das Mobiltelefon selbst als aktives Authentisierungs-Token fungieren. Während heute ein Benutzername und Passwort das gängige Verfahren zur Anmeldung bei Rechnern ist, kann ein Handy dies universeller und benutzerfreundlicher lösen. Bei Rechnern, die über einen kontaktlosen Kartenleser verfügen, emuliert ein NFC-Gerät eine Smartcard. Diese kann auch in Szenarien der Zutrittskontrolle eingesetzt werden. Insbesondere die Kosten für den Aufbau einer solchen Infrastruktur können erheblich reduziert werden, da keine Smartcards angeschafft und bestehende Infrastrukturen, wie die Lesegeräte der Türen, weiterhin genutzt werden können. Hierbei wird ein auf dem Mobiltelefon abgelegter Schlüssel für eine symmetrische Authentisierung gemäß [ISO09, Teil 2 oder 4] gegenüber dem uIdP genutzt. In Kombination mit Verfahren zur passiven aber kontinuierlichen Authentisierung des Nutzers an seinem Mobiltelefon, etwa mittels Gangerkennung [Nick09], kann eine sichere und benutzerfreundliche Zugangskontrolle realisiert werden.

4 Verteilter Komfort-Chipkartenleser

Während bei bisherigen Signaturkarten – im Einklang mit § 17 Abs. 2 SigG und § 15 Abs. 2 SigV – einfache Kartenleser zur Erzeugung von qualifizierten elektronischen Signaturen (QES) genutzt werden können, ist diese Möglichkeit beim nPA technisch ausgeschlossen. Die Signatur-Funktion (eSign-Funktion) des nPA kann erst dann genutzt werden, wenn sich ein Komfort-Chipkartenleser (Cat-K) gemäß [BSI11, Abschnitt 6] gegenüber dem Ausweis mit dem EAC-Protokoll gemäß [BSI10a] authentisiert hat. Dabei werden Sitzungsschlüssel für das *Secure Messaging* (SM) zwischen dem Terminal und dem nPA vereinbart, so dass die Übertragung der Signatur-PIN (eSign-PIN) vom Kartenleser zum Ausweis über diesen kryptographisch geschützten Kanal erfolgen kann. Da solche Leser zwingend über eine Tastatur, ein Display und einen sicheren Schlüsselspeicher verfügen müssen, sind sie vergleichsweise teuer. Es ist daher nicht zu erwarten, dass diese eine weite Verbreitung in der Bevölkerung finden werden.

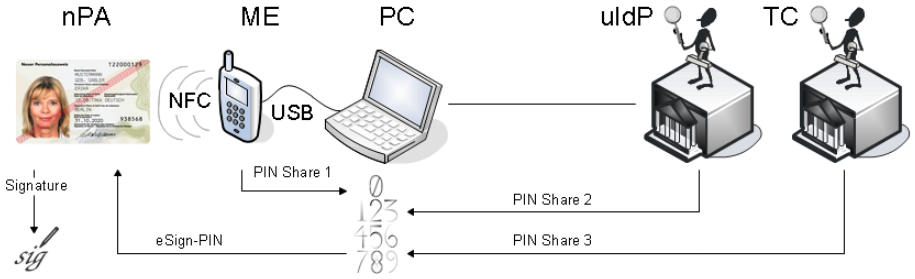


Abb. 2: Verteilter Komfort-Chipkartenleser (vCat-K)

Als kosteneffiziente Alternative schlagen wir einen *verteilten Komfort-Kartenleser* (vCat-K) vor, bei dem der technisch notwendige EAC-Kanal mit dem nPA vom universellen Authentisierungsdienst (uIdP) aufgebaut wird. Die mit Secure Messaging geschützten *Application Protocol Data Units* (APDU) für die Signaturerstellung werden dann *gemeinsam* von einer entsprechend erweiterten mobilen oder stationären eID-Applikation, mit dem uIdP sowie einer dritten vertrauenswürdigen Instanz, bspw. einem Trust Center (TC), erstellt. Der vCat-K ermöglicht es, unter Benutzung eines einfachen lokalen RFID-Lesers oder NFC-Mobiltelefons, einen Komfort-Chipkartenleser zu emulieren, so dass die eSign-Funktion des nPA künftig überall kostengünstig genutzt werden kann.

Die sichere Eingabe und kryptographische Aufbereitung der eSign-PIN erfolgt durch verteilt ablaufende kryptographische Protokolle (Multiparty Computation (MPC) [DaKe10]), die letztlich mit klassischen *Secret Sharing Verfahren* wie [Sham79] realisiert werden können. Damit ist sichergestellt, dass die eSign-PIN bei keinem der Teilnehmer vollständig vorliegt oder eingegeben werden muss und keine Signatur ohne Zutun des Ausweisinhabers erstellt werden kann.

4.1 PIN Eingabe und Signaturerzeugung

Im Standardszenario wird das EAC-Protokoll zwischen dem nPA und dem Komfort-Kartenleser durchgeführt. Die Auslösung der Signaturerzeugung erfolgt durch die Eingabe der eSign-PIN über das PIN-Pad. Der Hashwert wird von der eID-Anwendung berechnet und zur Bestätigung auf dem Display des Kartenlesers angezeigt. Im Szenario des vCat-K kommt für die Übergabe der PIN an den nPA ein Secret Sharing Verfahren und sichere MPC zum Einsatz.

4.1.1 Grundlagen

Perfektes Secret Sharing

Beim Secret Sharing wird ein Geheimnis in n sogenannte Shares so aufgeteilt, dass die Kenntnis einer nicht qualifizierten Teilmenge von $t < k$ der Shares absolut nichts über das Geheimnis verrät. Damit wird die perfekte Sicherheit garantiert, solange ein Angreifer nicht in Besitz von mindestens $k \leq n$ Shares gelangen kann.

Sichere Multiparty Computation

Sichere MPC bezeichnet die verteilte Berechnung einer beliebigen Funktion durch n Teilnehmer. Dabei kann eine sogenannte nicht qualifizierte Menge von $t < n$ Teilnehmern nichts weiter über das Ergebnis lernen, als ihre eigenen Ein- und Ausgaben. Sichere MPC kann basierend auf *Shamir's Secret Sharing* [Sham79] umgesetzt werden. In diesem Fall muss $t < n/2$ gelten [BeGW88, DaKe10, CrDM00], um perfekte Sicherheit gegen passive, adaptive Angreifer zu garantieren. Damit ist $n = 3$ und $t = 1$ die minimal mögliche Parameterwahl. MPC kann auch für $n = 2$ Teilnehmer bspw. basierend auf dem *Paillier Kryptosystem* [Pail99] umgesetzt werden [DaKe10]. Die Sicherheit des MPC-Protokolls hängt dann jedoch von der Sicherheit des Paillier Cryptosystems ab und die perfekte Sicherheit geht verloren.

Sicheres Multiparty AES

Für die verteilte AES Verschlüsselung müssen sowohl der Klartext als auch der AES Schlüssel in byteweise nach dem verwendeten Secret Sharing Verfahren aufgeteilter Form bei den Protokollteilnehmern vorliegen. In jeder Runde der AES Verschlüsselung werden die Teilschritte *SubBytes*, *ShiftRows*, *MixColumns* und *AddRoundKey* durchgeführt. Alle Teilschritte bis auf SubBytes können lokal durchgeführt werden. SubBytes erfordert zusätzliche Kommunikation zwischen den Teilnehmern. Die Verschlüsselung eines 128 Bit Blockes dauert im Durchschnitt ca. 2 Sekunden [DaKe10]. Danach liegt bei jedem Teilnehmer jeweils ein Share des Schlüsseltextes vor. Diese können zu einem klassischen AES Schlüsseltext kombiniert werden. Die Verwendung von Initialisierungsvektoren (IV) und Chaining Modes für Klartexte die länger als 128 Bit sind, kann auch für die verteilte AES Verschlüsselung umgesetzt werden.

4.1.2 Sichere verteilte PIN-Eingabe und Verschlüsselung

Wie erläutert, werden für die verteilte PIN-Eingabe und Verschlüsselung drei Teilnehmer benötigt: Der uIdP, der Client und als dritte Instanz das TC bei dem das Signaturzertifikat erworben wurde. Das TC gilt a priori als vertrauenswürdig und ist von vornherein in das Signatur-Szenario involviert, es kann jedoch auch ein unabhängiger vertrauenswürdiger Dienstanbieter verwendet werden. Das Aufteilen der PIN in drei Shares wird durch den Nutzer bei Aktivierung der eSign-Funktion in einer sicheren Umgebung durchgeführt. Jeweils ein Share wird dann an den uIdP und das TC über einen sicheren Kanal übermittelt. Der Ausweisinhaber behält zusätzlich die vollständige eSign-PIN zur herkömmlichen Verwendung. Für die Nutzung der eSign-Funktion wird zuerst das PACE-Protokoll mit der auf dem Ausweis aufgedruckten *Card Access Number* (CAN) ausgeführt. Es folgt das EAC-Protokoll zwischen nPA und uIdP. Nach Abschluss der gegenseitigen Authentisierung liegen beim uIdP die Sitzungsschlüssel K_{MAC} und K_{ENC} vor. K_{MAC} wird für die Berechnung des *Message Authentication Codes* (MAC) und K_{ENC} für das Verschlüsseln einer APDU im Secure Messaging Kanal verwendet. K_{ENC} wird vom uIdP ebenso nach Shamir in Shares aufgeteilt. Danach wird jeweils ein Share an das TC und eines an den Client über einen TLS-Kanal übermittelt. K_{MAC} muss vollständig an den Client übertragen werden, da dieser für jede APDU abschließend den MAC berechnen muss.

Zum Auslösen der Signaturerzeugung wird eine *Verify* APDU mit der eSign-PIN im Datenteil an den Ausweis gesendet. Beim Secure Messaging [BSI10a, Anhang F] wird AES im CBC-Modus (AES CBC) für die Verschlüsselung und AES im CMAC-Modus für die Berechnung des 8 Byte langen MAC verwendet. Das Padding der Daten erfolgt gemäß ISO 7816-4 [ISO05b]. Die Padding-Bytes und der IV werden vom uIdP erzeugt, aufgeteilt und an die Teilnehmer verteilt. Da das Secret Sharing byteweise durchgeführt wird, können die Teilnehmer ihr PIN- und Padding-Share lokal verknüpfen, um das notwendige 16 Byte Share zu erhalten. Damit sind die Voraussetzungen für sicheres Multiparty-AES erfüllt und die AES Verschlüsselung der eSign-PIN kann verteilt ablaufen. Als Ergebnis liegt jeweils ein Schlüsseltext-Share der eSign-PIN beim Client, uIdP und TC vor. Mit *Lagrange Interpolation* lässt sich aus jeweils zwei der Shares das Kryptogramm rekonstruieren, welches einer klassischen AES Verschlüsselung der auf 16 Byte gepaddeten PIN unter dem Sitzungsschlüssel K_{ENC} entspricht. Der uIdP sendet hierfür sein verschlüsseltes Share an den Client. Dieser führt die Rekonstruktion und das abschließende Erstellen der APDU durch.

iPIN zur Vermeidung von Replay Angriffen

Probleme entstehen, wenn von einem Angreifer ausgegangen wird, der Zugriff auf den nPA hat und das PIN-Share des Ausweisinhabers mithören kann. Daher ist eine starke Authentisierung des Ausweisinhabers beim uIdP im Vorfeld erforderlich. Diese dürfte jedoch nicht über den nPA durchgeführt werden, da der Angreifer diesen mit entsprechendem Zugriff ebenso zur Authentisierung nutzen könnte. Um dieses Problem zu lösen schlagen wir indizierte PIN-Listen (iPIN-Listen) vor. Aufgrund des Secret Sharings der PIN gibt es 2^{48} verschiedene Share-Kombinationen[1] aus drei Shares, welche die selbe PIN rekonstruieren. Aus diesen Share-Kombinationen können vom Ausweisinhaber indizierte Listen erstellt werden, sodass die drei Shares mit dem selben Index die eSign-PIN rekonstruieren.[2] Verbleibt eine Liste beim Ausweisinhaber und verteilt er jeweils eine an den uIdP und das TC, dann erhält man ein sicheres Einmal-Passwort Verfahren zur Auslösen der Signaturerzeugung. Bei jedem Signaturvorgang gibt der uIdP, analog zum iTAN-Verfahren, den Index des PIN-Shares bekannt. Damit kann auf eine Authentisierung im Vorfeld verzichtet werden, da diese durch Kenntnis und Verwendung der korrekten iPIN erfolgt.

4.1.3 Signaturerzeugung

Zum Erstellen der Signatur muss mit einer *PSO:Compute Digital Signature* der zu signierende Hashwert an den Ausweis übermittelt werden. Das Dokument bzw. der Hashwert wird vom Client an den uIdP gesendet, der im Besitz des Schlüssels K_{ENC} ist, und den Hashwert verschlüsseln kann. Zunächst sendet der uIdP den Hashwert, signiert mit seinem speziell dafür vorgesehenen privaten Schlüssel, über einen zweiten unabhängigen Kanal (bspw. per SMS) zur Bestätigung an den Ausweisinhaber. Diese Schritte erfolgen vor der in Abschnitt 4.1.2 beschriebenen iPIN-Eingabe. Dadurch kann der uIdP gleichzeitig auch den Index der zu verwendenden iPIN bekannt geben und der Ausweisinhaber bestätigt den Hashwert über die Eingabe der iPIN. Der Versand des verschlüsselten Hashwertes an den Client erfolgt dann nach der PIN-Verarbeitung. Nachdem der nPA die eSign-PIN sowie den Hashwert erhalten und verifiziert bzw. signiert hat, erhält der Client die mit K_{ENC} verschlüsselte Signatur vom Ausweis. Um Fälschungen durch den uIdP auszuschließen, leitet der Client die verschlüsselte Signatur zusam-

[1] Da 6 Bytes (48 Bit) für die Darstellung der eSign-PIN innerhalb der APDU verwendet werden. Für sechsstellige iPINs ist die Anzahl jedoch auf 10^6 Kombinationen beschränkt. Anzumerken ist auch, dass sich bei Änderung eines Shares zwangsläufig die anderen beiden Shares ändern.

[2] Das Erstellen der iPIN-Listen könnte auch als Service des TC oder uIdP realisiert werden. Der Ausweisinhaber müsste dann die eSign-PIN im Ausweis entsprechend setzen.

men mit dem Hashwert und dem eigenen Schlüssel-Share an das TC weiter. Aus dem eigenen und dem Schlüssel-Share des Clients rekonstruiert das TC den Schlüssel K_{ENC}, entschlüsselt die Signatur und vergleicht diese mit dem erhaltenen Hashwert. Nach positiver Prüfung wird die entschlüsselte Signatur an den Client übermittelt.

4.2 Sicherheit des Verfahrens

Im Szenario des vCat-K wird davon ausgegangen, dass sowohl das Trust Center als auch der Authentisierungsdienst zertifizierte und vertrauenswürdige Parteien sind.

Sicherheit der PIN

Aufgrund des Secret Sharings haben sowohl der uIdP, der Client[3] als auch das TC keine Kenntnisse über die tatsächliche eSign-PIN. Durch Einsatz des vorgestellten iPIN-Verfahrens kann ein Angreifer durch Kompromittierung des Clients lediglich in Besitz einer iPIN gelangen. Diese ist jedoch nur für eine Sitzung gültig und erlaubt keine Rückschlüsse auf die tatsächliche PIN. Die vollständige verschlüsselte PIN liegt nur dem Client vor. Der uIdP und das TC besitzen nur ihre jeweiligen Shares. Der Schlüssel K_{ENC} liegt nur dem uIdP und dem TC vor, sodass die PIN von keiner Partei entschlüsselt werden kann. Damit ist die Sicherheit der PIN garantiert, solange nicht mindestens zwei Teilnehmer zusammenarbeiten, bzw. kompromittiert sind.

Sicherheit gegen Signaturfälschung

Bei der Signaturerstellung sind Client und uIdP involviert. Unter der Annahme eines kompromittierten Clients, könnte dieser vom Ausweisinhaber unbemerkt, einen anderen Hashwert an den uIdP schicken und damit beliebige Dokumente signieren. Mittels der vom uIdP signierten Bestätigung über einen zweiten unabhängigen Kanal (sowie Endgerät), kann der zu signierende Hashwert jedoch verifiziert werden. Dies realisiert die Eigenschaft des vertrauenswürdigen Displays des Komfort-Kartenlesers. Selbst wenn nur ein Smartphone als Client zur Verfügung steht und an dieses auch die Bestätigungs-SMS gesendet wird, kann ein Angreifer der die vollständige Kontrolle über das Smartphone hat den signierten Hashwert nicht manipulieren. Jedoch besteht in diesem Fall das selbe Problem wie bei einem Rechner mit angeschlossenem Cat-K Leser. Gelingt es dem Angreifer dem Nutzer auf dem Rechner einen falschen Hashwert anzuzeigen sowie diesen ebenso an den Cat-K Leser zu übermitteln würde auch hier der Nutzer den angezeigten Hashwert fälschlicherweise als korrekt verifizieren. Verwendet man dagegen einen vCat-K bestehend aus Rechner mit Kartenleser und einem separaten Smartphone zur Verifikation, so kann der Hashwert des Dokuments auf dem Smartphone unabhängig berechnet werden, was eine verlässliche Verifikation mit dem vom uIdP signierten Hashwert ermöglicht.

Es besteht die Gefahr, dass ein kompromittierter uIdP einen anderen als dem Ausweisinhaber angezeigten Hashwert an den Ausweis sendet. Da der Hashwert im Secure Messaging Kanal verschlüsselt ist und der Client den Schlüssel nicht besitzt, kann er den Hashwert vor der Signaturerstellung nicht überprüfen. Daher wird die erstellte Signatur nicht zurück zum uIdP gesendet, sondern an das TC. Mit den zusätzlichen Informationen kann das TC die Signatur entschlüsseln und verifizieren. Aufgrund der Signatur wird ein geänderter Hashwert bei der Überprüfung durch das TC aufgedeckt. Andererseits kann das TC keinen Einfluss auf die Signatur nehmen, weil es nur die bereits erstelle Signatur erhält. Damit ist das Verfahren sicher gegen Signatur von ungewollten Dokumenten, solange nicht mindestens zwei Teilnehmer zusammenarbeiten.

[3] Unter der Annahme, dass die einmalige Erzeugung der PIN-Shares in einer sicheren Umgebung erfolgt.

Da das MPC Verfahren perfekte Sicherheit bietet und die Kommunikation über sichere Kanäle abläuft, ist die Sicherheit des gesamten Verfahrens durch die Sicherheit von AES bestimmt.

4.3 Rechtliche und regulatorische Aspekte

Für eine mögliche praktische Realisierung des verteilten Komfort-Chipkartenlesers und der verteilten PIN-Eingabe sind insbesondere auch die rechtlichen Rahmenbedingungen des Signaturgesetzes (SigG) und der Signaturverordnung (SigV) zu berücksichtigen. Dies umfasst insbesondere die Frage, ob und wie ein vCat-K von einer unabhängigen Instanz geprüft werden muss, ob die Verteilung der PIN grundsätzlich konform zu SigG und SigV ist und ob man mit einem derart verteilten System überhaupt eine *fortgeschrittene elektronische Signatur* erstellen kann, die ja nach § 2 Nr. 2 c) SigG mit Mitteln erzeugt sein muss, die der Signaturschlüsselinhaber unter seiner alleinigen Kontrolle halten kann.

Prüfungsanforderungen

Ein Cat-K Leser ist in rechtlicher Hinsicht eine Signaturanwendungskomponente nach § 2 Nr. 11 SigG, die die Anforderungen des § 17 Abs. 2 SigG und § 15 Abs. 2 SigV erfüllen muss. Während gemäß § 17 Abs. 4 Satz 2 SigG für die Erfüllung dieser Anforderungen eine Herstellererklärung genügen würde, ist in [BSI11, Anhang E] für Cat-K Leser eine Common Criteria Evaluation der Stufe EAL 3+ vorgesehen, bei der die Sicherheitsmaßnahmen gegen ein hohes Angriffspotenzial zu prüfen sind und eine vollständige Missbrauchsanalyse durchzuführen ist (vgl. Anlage 1 zu SigV). Diese Anforderungen müssen auch von einem vCat-K erfüllt werden und obwohl die Prüfung und Bestätigung eines vCat-K sicherlich kein einfaches Unterfangen ist, erscheint dies nicht grundsätzlich ausgeschlossen.

Zulässigkeit der PIN-Verteilung

Eine *sichere Signaturerstellungseinheit* (SSEE) müssen gemäß § 17 Abs. 1 SigG vor einer unberechtigten Nutzung des Signaturschlüssels schützen und gemäß § 15 Abs. 1 SigV sicherstellen, dass der Signaturschlüssel erst nach Identifikation des Inhabers durch Besitz und Wissen oder durch Besitz und ein oder mehrere biometrische Merkmale (siehe auch [Hü07]) angewendet werden kann. Gemäß [Bund09] muss eine PIN als signaturauslösendes Wissensdatum gemäß SigV bestimmte Anforderungen[4] erfüllen, damit die in Anlage 1 von SigV geforderte Widerstandsfähigkeit gegen einen Angreifer mit hohem Angriffspotential erreicht werden kann.

Falls der Zertifizierungsdiensteanbieter (ZDA) die signaturauslösende PIN erzeugt und an den Signaturschlüsselinhaber übergibt, muss sichergestellt sein, dass

1. „sie zu keinem Zeitpunkt nach Einbringen in die SSEE bzw. Erzeugen in der SSEE außerhalb der SSEE gespeichert ist und
2. der ZDA dem Signaturschlüsselinhalber ein abgeleitetes Datum (bzgl. PIN) auf einem geeigneten Datenträger übergibt. Hierbei muss ein unberechtigter Versuch, Kenntnis vom abgeleiteten Datum zu nehmen, leicht durch Inaugenscheinnahme des Datenträgers erkannt werden können.

Das dedizierte Verfahren zur Übergabe der PIN bedarf weiterer Konkretisierung und ist im Sicherheitskonzept des ZDA detailliert darzulegen.“

Vor dem Hintergrund dieser derzeit gültigen AGAB[5]-Beschlüsse [Bund09] ist daher die Vertei-

[4] Dort ist u.a. festgelegt, dass eine PIN mindestens sechs Zeichen (mindestens aus dem Vorrat 0...9) umfassen und bei einer sechsstelligen PIN einen Fehlbedienungszähler von höchstens drei besitzen muss.

[5] Arbeitsgemeinschaft der anerkannten Bestätigungsstellen gemäß § 18 SigG.

lung der signaturauslösenden PIN nicht grundsätzlich ausgeschlossen, sondern lediglich an die oben genannten Bedingungen geknüpft.

Alleinige Kontrolle im verteilten System

Schließlich bleibt die Frage, ob mit einem vCat-K überhaupt eine fortgeschrittene elektronische Signatur gemäß § 2 Nr. 2 SigG erstellt werden kann, da ja in § 2 Nr. 2 SigG gefordert ist, dass eine solche Signatur mit Mitteln erzeugt werden muss, die der Signaturschlüssel-Inhaber unter seiner „alleinigen Kontrolle“ halten kann.

Was genau unter der alleinigen Kontrolle zu verstehen ist, wird im Signaturgesetz nicht näher definiert und die amtliche Begründung führt hierzu nur aus, dass der „Signaturschlüsselinhaber seine Signaturerstellungseinheit vor unbefugter Nutzung schützen können muss.“ Die „unbefugte Nutzung“ der Signaturerstellungseinheit scheint sich hier vor allem auf die unberechtigte Erzeugung einer Signatur zu beziehen. Diese ist auch dann wirksam verhindert, wenn ein Angreifer lediglich über eine Teilmenge der zur Signaturerstellung benötigten Komponenten und Geheimnisse verfügt, weil er nur einen Teil[6] der PIN kennt.

Die Bundesnetzagentur scheint § 2 Nr. 2 c) SigG übrigens ähnlich zu interpretieren, da in der Antwort zu Frage 18a ihrer FAQ[7] erläutert wird, dass bei einer so genannten „Telesignatur“ die *eigentlich fehlende alleinige Kontrolle* des Inhabers über seine SSEE durch eine Kombination aus anderen technischen und organisatorischen Sicherheitsmaßnahmen (Zutrittsregelungen, Split-PIN, Vieraugenprinzip etc.) nachgebildet werden muss.

Vor diesem Hintgergrund erscheint es also insgesamt nicht ausgeschlossen, ein System für den verteilten Komfort-Chipkartenleser zu entwickeln, das alle Anforderungen des Signaturgesetzes und der Signaturverordnung erfüllt.

4.4 Vergleich zu anderen Ansätzen

Mediated Signatures

In [KKLDT10] wird vorgeschlagen, sogenannte *Mediated Signatures* [TsWo01] und *Hash-Chains* anstelle einer QES zu verwenden.[8] An der Signaturerstellung sind mindestens zwei Parteien beteiligt: Der Benutzer und ein Server-Dienst, welcher *Security Mediator* genannt wird.

Benutzer und Security Mediator besitzen jeweils einen Teil des Signaturschlüssels. Zur Signaturerstellung erzeugt einer der Beteiligten mit seinem Teilschlüssel eine Teilsignatur über das zu signierende Dokument. Mit dieser Teilsignatur und dem anderen Teilschlüssel erzeugt der zweite Beteiligte eine weitere Teilsignatur und kombiniert beide zu einer kompletten Signatur.

Aus der Kombination von Mediated Signatures mit ID-basierten Kryptosystemen wie es z.B. in [YuYS07] vorgeschlagen wird, ergibt sich ein weiterer interessanter Anwendungsfall. Der in beiden vorgenannten Arbeiten dargestellte Vorteil der sofortigen Durchsetzbarkeit von Revokationen durch Verweigerung des Serverdienstes ist beim vCat-K genauso möglich.

Mediated Signatures erfordern spezielle Eigenschaften des zugrundeliegende Signaturverfahrens, wie bspw. die Multiplikativität bei RSA-Signaturen, um die verteilte Signaturerzeugung zu ermöglichen. Da bei vCat-K nur die eSign-PIN bzw. deren Verschlüsselung gemeinsam be-

---

[6] Der Systematik aus [Bund09] folgend muss bei der Verteilung der PIN insbesondere sichergestellt sein, dass die Wahrscheinlichkeit für das Erraten der fehlenden Bestandteile höchstens gleich $3 \cdot 10^{-6}$ ist.

[7] Siehe `http://www.bundesnetzagentur.de/cln_1932/SharedDocs/FAQs/DE/BNetzA/QES/18aUnterschiedlicheBegriffe.html?nn=75924`.

[8] In Polen gibt es hierfür eine kommerzielle Lösung unter dem Namen PKI2.0 (`http://www.pki2.eu`)

rechnet wird, ist das Verfahren direkt für beliebige Signaturalgorithmen einsetzbar und könnte darüber hinaus analog auf andere MPC-fähige Verschlüsselungsmethoden für den Secure Messaging Kanal angepasst werden.

Im Gegensatz zu Mediated Signatures verbleibt der Schlüssel bei vCat-K im alleinigen Besitz des Benutzers. Der Benutzer kann auch ohne Mitwirkung weiterer Parteien Signaturen erstellen und den anderen Parteien durch Ändern der eSign-PIN sämtliche Rechte entziehen. Des Weiteren verhindert vCat-K die Übermittlung eines gefälschten Hashwertes durch einen kompromittierten Client und bietet ein Einmal-Passwort Verfahren.

Mobile qualifizierte elektronische Signaturen

Zur Erzeugung mobiler qualifizierter Siganturen bestehen verschiedene Ansätze. Ein Ansatz ist die Erzeugung der Signatur auf einem speziellen dafür vorgesehenen Server und anschließende Versendung der Signatur zu einem mobilen Endgerät. Bei den in [OrCK10] vorgestellten *Qualified Mobile Server Signature* sind die Signatur-Schlüssel der Benutzer in einem Hardware-Sicherheitsmodul auf dem Server gespeichert und mit einer privaten PIN geschützt. Die Signaturerstellung führt der Benutzer über die Web-Anwendung des Servers durch. Zusätzlich zur Auslösen der Signaturerzeugung per PIN muss der Benutzer einen Bestätigungscode eingeben, der zuvor per SMS an sein Mobiltelefon gesendet wurde.

Fritsch et al. [FrRR03, Ross04] kommen jedoch zu dem Schluss, dass ein solcher Ansatz die gesetzlichen Anforderungen zur Erstellung qualifizierter Signaturen nicht erfüllen kann, da die Schlüssel nicht unter der alleinigen Kontrolle des Nutzers stehen. Hier wird der Ansatz vertreten die Signatur direkt auf dem mobilen Endgerät unter Zuhilfenahme einer sicheren Signaturerstellungseinheit zu erstellen. Hierfür wird vorgeschlagen, eine Signaturerstellungseinheit auf einer USIM Karte in mobile Endgeräte zu integrieren, was jedoch zusätzliche Investitionen, z.B. des Mobilfunkanbieters, erfordern würde.

Um die Sicherheit gegen softwareseitige Manipulation des mobilen Endgerätes und damit möglichen Signaturfälschungen zu gewährleisten, empfiehlt Rossnagel in [Ross04] den Einsatz eines sicheren Microkernels wie *PERSEUS* [PfSt01] auf dem mobilen Endgerät, welcher die Signaturanwendung ausführt und Display-Ausgaben sowie Tastatur-Eingaben absichert.

vCat-K ermöglicht mobile QES durch den Einsatz NFC-fähiger Mobiltelefone als Kartenleser für den nPA. Dadurch können mobile qualifizierte Signaturen unabhängig und ohne Investitionen von Mobilfunkanbietern implementiert werden, wodurch eine maßgebliche Hürde für den flächendeckenden Einsatz beseitigt wird. Die Sicherheit gegen Signaturfälschungen und das Abhören der PIN stellt vCat-K durch das iPIN Verfahren und die Verifikation des zu signierenden Dokumentes über einen unabhängigen Kanal sicher. Damit entfällt die Notwendigkeit der Installation eines sicheren Microkernels oder eines speziellen Hardware-Sicherheitsmoduls, wodurch mobile Standard-Endgeräte ohne Veränderungen verwendet werden können.

Im Gegensatz zu *Qualified Mobile Server Signature* verbleiben die Signaturschlüssel unter der Kontrolle des Ausweisinhabers. Das Problem des Nachweises, dass eine Signatur tatsächlich vom Nutzer autorisiert wurde entfällt damit.

5 Ausblick

Die in Abschnitt 4 vorgestellte Architektur des vCat-K ermöglicht die Erzeugung von blinden Signaturen wie sie als erstes von Chaum [Chau82] vorgeschlagen wurden. Betrachtet man den uIdP als Nutzer, den Ausweisinhaber als Signierenden und verzichtet auf Rückbestätigung sowie Signaturverifikation durch das TC, so erhält man ein blindes Signaturschema. Die Blindheit der Signatur ist dabei gegeben, solange AES als sicher anzusehen ist. Die einmalige Frei-

gabe der Signaturfunktion ist durch iPINs sichergestellt, womit garantiert werden kann, dass der Nutzer keine zusätzlichen Signaturen erstellen kann. Im Gegensatz zu anderen blinden Signaturverfahren, wie bspw. das in [Chau82] vorgeschlagene RSA-basierte Verfahren, ist der vCat-K durch vertrauenswürdige Hardware unterstützt und kann daher mit jedem beliebigen Signaturverfahren umgesetzt werden, da keine speziellen Eigenschaften des Signaturverfahrens ausgenützt werden.

Bei ressourcen-beschränkten Geräten sind insbesondere die Operationen auf elliptischen Kurven beim PACE-Protokoll eine Herausforderung [WHBK+11]. Auch bieten gängige Klassenbibliotheken keine Funktionalität für solche Operationen, sodass auf einen externen Krypto-Provider zurück gegriffen werden muss. Denkbar ist, hier ein *verteiltes PACE* einzusetzen. Dabei entschlüsselt der Client ausschließlich die im ersten Protokollschritt vom nPA empfangene Zufallszahl und übergibt sie an den uIdP. Dieser führt dann die weiteren Schritte des Protokolls aus. Damit bleibt die CAN, oder die sonstigen für PACE verwendbaren Passwörter, auf dem Client und die Operationen werden vom Server durchgeführt. Damit reduzieren sich auch die Anforderungen an den Client und die benötigten kryptographischen Funktionen beschränken sich nur noch auf AES und SHA-1. Denkbar wäre auch ein verteiltes PACE mit iPINs zu realisieren. Dadurch könnte man dem uIdP bei jedem Authentisierungsvorgang eine iPIN schicken, dieser könnte die PIN rekonstruieren um PACE durchzuführen. Damit würde ein Abhören der PIN auf dem Client verhindert.

6 Fazit

Wir haben gezeigt, dass NFC-fähige Mobiltelefone eine kostengünstige und effektive Alternative zu herkömmlichen Kartenlesern sein können. Hervorzuheben ist die Möglichkeit nicht nur auf herkömmliche Kartenleser, sondern auch auf PCs zu verzichten, um Authentisierungen oder Signaturen mit Chipkarten durchzuführen. In geeigneten Fällen kann sogar auf die Chipkarte verzichtet werden, da diese durch das Mobiltelefon emuliert werden kann.

Insbesondere in der hier speziell untersuchten Kombination mit dem nPA ergeben sich interessante Anwendungszenarien und Geschäftsmodelle. Das Mobiltelefon kann als Basis-, Standard-, oder Komfortleser an ein PC angeschlossen werden oder vollständig mobil (ohne PC) genutzt werden, um die elektronischen Funktionalitäten des nPA zu nutzen. Die Vorteile dieser Lösungen sind signifikante Kostenersparnisse und erweiterte Flexibilität für den Benutzer. Deshalb darf erwartet werden, dass hierdurch die Akzeptanz von elektronischen Identitäten und Signaturen mit dem nPA im Speziellen und in der Folge davon im Allgemeinen gesteigert wird.

Im Falle von QES mit dem nPA werden besondere Sicherheitsanforderungen an den Kartenleser gestellt, die im herkömmlichen Fall durch den Einsatz von Komfort-Chipkartenleser sichergestellt werden. Die hier vorgestellte Lösung des vCat-K entspricht nicht der technischen Spezifikation eines Komfortlesers, bietet jedoch in jedem Fall das geforderte Sicherheistniveau. Wird das Mobiltelefon als Kartenleser zusammen einem PC verwendet, wird das geforderte Sicherheitsniveau übertroffen, und zusätzlich die Benutzerfreundlichkeit des Systems erhöht. Aus technischer Sicht steht dem Einsatz der vorgestellten Lösungen also nichts entgegen.

Auch hinsichtlich der rechtlichen und regulatorischen Aspekte im Zusammenhang mit qualifizierten elektronischen Signaturen stell sich die Lage positiv dar, da durch die verschiedenen technischen und organisatorischen Sicherheitsmaßnahmen die alleinige Kontrolle des Benutzers über seine Signaturerstellungseinheit sichergestellt werden kann. Dies beruht nicht zuletzt darauf, dass im hier vorgestellten System, im Gegensatz zu anderen Lösungen, nur die PIN verteilt wird, und der Schlüssel im alleinigen Besitz des Benutzers verbleibt.

Literatur

[BBI10] Die Beauftragte der Bundesregierung für Informationstechnik: IT-Sicherheitskit für Bürgerinnen und Bürger (2010).

[BeGW88] M. Ben-Or, S. Goldwasser, A. Wigderson: Completeness theorems for non-cryptographic fault-tolerant distributed computation. *In: Proceedings of the twentieth annual ACM symposium on Theory of computing*, STOC '88, ACM, New York, NY, USA (1988), 1–10.

[Beil09] K. Beilke: NFCBTPCSC - Nokia NFC Bluetooth PCSC Reader (2009).

[BSI10a] Bundesamt für Sicherheit in der Informationstechnik: Advanced Security Mechanism for Machine Readable Travel Documents - Extended Access Control (EAC), Password Authenticated Connection Establishment (PACE), and Restricted Identification (RI). BSI-TR-03110, Version 2.05 (2010).

[BSI10b] Bundesamt für Sicherheit in der Informationstechnik: Technische Richtlinie eID-Server. BSI-TR-03130, Version 1.4.1 (2010).

[BSI11] Bundesamt für Sicherheit in der Informationstechnik: Anforderungen an Chipkartenleser mit nPA Unterstützung. BSI-TR-03119, Version 1.2 (2011).

[Bund09] Bundesnetzagentur: Beschlüsse der 37. Sitzung der Arbeitsgemeinschaft anerkannter Bestätigungsstellen (AGAB) vom 04.03.2009 – PIN-/PUK-Techniken bei sicheren Signaturerstellungseinheiten nach § 2 Nr. 10 SigG (2009).

[Chau82] D. Chaum: Blind Signatures for Untraceable Payments. *In: International Crytology Conference* (1982), 199–203.

[CrDM00] R. Cramer, I. Damgård, U. Maurer: General secure multi-party computation from any linear secret-sharing scheme. *In: Proceedings of the 19th international conference on Theory and application of cryptographic techniques*, EUROCRYPT'00, Springer-Verlag, Berlin, Heidelberg (2000), 316–334.

[DaKe10] I. Damgård, M. Keller: Secure Multiparty AES. *In: Financial Cryptography* (2010), 367–374.

[FrRR03] L. Fritsch, J. Ranke, H. Rossnagel: Qualified mobile electronic signatures: Possible, but worth a try? *In: Information Security Solutions Europe (ISSE) 2003 Conference*, Vienna (2003).

[Hors09] M. Horsch: MobilePACE - Password Authenticated Connection Establishment implementation on mobile devices. Bachelor Thesis, TU Darmstadt (2009).

[Hors11] M. Horsch: MONA - Mobile Authentisierung mit den neuen Personalausweis. Master Thesis, TU Darmstadt (2011), In preparation.

[Hü07] D. Hühnlein: Rechtliche Rahmenbedingungen der Komfortsignatur. *In: P. Horster (Hrsg.), $D \cdot A \cdot CH$-Security 2003*, IT-Verlag (2007), 189–200.

[ISO01] ISO/IEC 14443: Identification cards - Contactless integrated circuit(s) cards - Proximity cards - Part 1-4. International Standard (2001).

[ISO04] ISO/IEC 18092: Information technology – Telecommunications and information exchange between systems – Near Field Communication – Interface and Protocol (NFCIP-1). International Standard (2004).

[ISO05a] ISO/IEC 21481: Information technology – Telecommunications and information exchange between systems – Near Field Communication Interface and Protocol -2 (NFCIP-2). International Standard (2005).

[ISO05b] ISO/IEC 7816: Identification cards – Integrated circuit cards – Part 4: Organization, security and commands for interchange. International Standard (2005).

[ISO09] ISO/IEC 9798: Information Technology – Security Techniques – Entity Authentication – Part 1-6. International Standards (1997-2009).

[JuR11] Juniper Research: Press Release: 1 in 5 Smartphones will have NFC by 2014, Spurred by Recent Breakthroughs: New Juniper Research Report (2011).

[KKLDT10] P. Kubiak, M. Kutylowski, A. Lauks-Dutka, M. Tabor: Mediated Signatures - Towards Undeniability of Digital Data in Technical and Legal Framework. *In: BIS (Workshops)* (2010), 298–309.

[Nick09] C. Nickel: Authentisierung an mobilen Geräten mittels Gangerkennung. In: *Datenschutz und Datensicherheit (DuD)*, 5 (2009), 280–283.

[OrCK10] C. Orthacker, M. Centner, C. Kittl: Qualified mobile server signature. In: *IFIP Advances in Information and Communication Technology*, 330 (2010), 103–111.

[Pail99] P. Paillier: Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. *In: J. Stern (Hrsg.), Advances in Cryptology– EUROCRYPT '99*, Springer Berlin / Heidelberg, *Lecture Notes in Computer Science*, Bd. 1592 (1999), 223–238.

[PfSt01] B. Pfitzmann, C. Stüble: PERSEUS: A Quick Open-source Path to Secure Signatures. *In: 2nd Workshop on Microkernel-based Systems* (2001).

[Ross04] H. Rossnagel: Mobile Qualified Electronic Signatures and Certification on Demand. *In: EuroPKI* (2004), 274–286.

[Sham79] A. Shamir: How to share a secret. In: *Communications of the ACM*, 22 (1979), 612–613.

[TsWo01] D. B. X. D. G. Tsudik, C. Wong: A method for fast revocation of public key certificates and security capabilities. *In: Proceedings of the 10th USENIX Security Symposium* (2001), 297–308.

[WHBK+11] A. Wiesmaier, M. Horsch, J. Braun, F. Kiefer, D. Hühnlein, F. Strenzke, J. Buchmann: An efficient PACE Implementation for mobile Devices. *In: ASIA CCS '11: 6th ACM Symposium on Information, Computer and Communications Security* (2011).

[YuYS07] Y. Yu, B. Yang, Y. Sun: ID-Based Threshold Signature and Mediated Signature Schemes. *In: Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing* (2007), 473–478.