

dass die Chipkarte mit dem Dump-Programm niemals die Fertigung verlassen darf. Würde diese Chipkarte vollständig personalisiert, könnten mit dem enthaltenen Dump-Programm alle geheimen Informationen ausgelesen werden. Deshalb besitzt diese Chipkarte entsprechende Mechanismen, damit sie auch bei Verwechslung oder im Falle eines Angriffs niemals zum Auslesen des Speichers missbraucht werden kann. Dies lässt sich einfach dadurch erreichen, dass sich beispielsweise das Dump-Programm beim nächsten Reset der Chipkarte selbstständig löscht. Damit kann die Chipkarte nur in einer einzigen Sitzung benutzt werden, da sie bei der ersten Spannungsunterbrechung ihre Dump-Funktionalität verliert.

Personalisierung/Individualisierung

Der nächste Schritt zur versandfertigen Chipkarte ist die Personalisierung, manchmal auch Individualisierung genannt. Der Begriff Personalisieren im weiteren Sinne bedeutet, dass alle Daten, die einer einzelnen Person oder einer einzelnen Karte zugeordnet sind, in die Chipkarte ein- bzw. aufgebracht werden. Dies können beispielsweise Name und Anschrift, aber auch kartenbezogene Schlüssel sein. Wichtig dabei ist nur, dass es kartenindividuelle Daten sind.

Grundsätzlich unterscheidet man dabei zwischen optischer und elektrischer Personalisierung. Das Prägen (*embossing*) von Zeichen oder die Lasergravur von Schrift und Bildern auf den Kartenkörper ist der optische Teil der Personalisierung. Der elektrische Teil ist dann das Laden der personenbezogenen Daten in den Mikrocontroller und das Schreiben von Daten auf den Magnetstreifen. Die Zeitdauer für die optische Personalisierung hängt sehr stark von den Besonderheiten ab und lässt sich nicht allgemein angeben. Die elektrische Personalisierung benötigt abhängig von der in die Chipkarte zu ladenden Datenmenge meistens zwischen 5 und 20 Sekunden.

Die Prägung von Namen oder ähnlichen kartenindividuellen und zeichenorientierten Daten erstellt eine Maschine, bei der mit hoher Geschwindigkeit und mit Kraftaufwand Metallbuchstaben auf die Kartenrückseite geschlagen werden. Da es sich dabei um ein relativ einfaches, aber sehr lautes und schwingungserzeugendes Verfahren handelt, werden die betreffenden Maschinen meist getrennt von der übrigen Fertigung aufgestellt. Die Lasergravur, bei der ein Laser Bereiche direkt unterhalb der Deckschicht des Kartenkörpers schwärzt, wird sehr oft statt einer Prägung eingesetzt. Dies ist auch ein taugliches Verfahren, wenn sich ein Schwarzweißfoto auf dem Kartenkörper befinden muss.

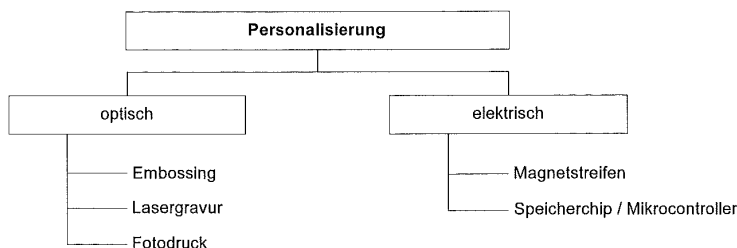


Bild 10.60 Aufstellung der auf einer Chipkarte zu personalisierenden Kartenelemente.

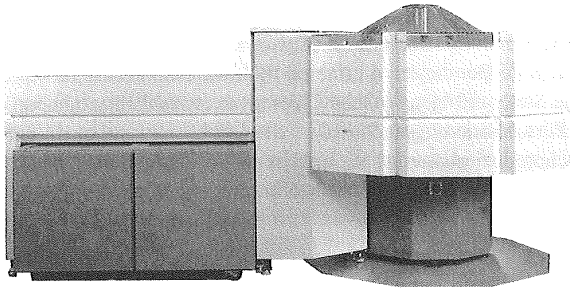


Bild 10.61 Modulares Personalisierungssystem Luchs 5000 mit u. a. integrierter Laserbeschriftungseinheit. (Giesecke & Devrient)

Die Daten für den Chip schreibt man analog zur Initialisierung in den Speicher. Sofern es sich jedoch dabei um geheime Schlüssel handelt, benutzt man oft eine kryptografisch gesicherte Datenübertragung,¹¹ sodass das Abhören der Übertragungsleitung zur Chipkarte einem Angreifer keine Vorteile bringt. Für Karten im Zahlungsverkehr wird manchmal ein noch aufwändigeres Verfahren eingesetzt. Dazu werden die vom Kartenherausgeber verschlüsselten Personalisierungsdaten in einem speziellen Sicherheitsmodul an der Personalisierungsmaschine umgeschlüsselt und anschließend direkt in die Karte geladen. Der Vorteil bei diesem Verfahren liegt darin, dass der Personalisierer die geheimen Daten der Karte nicht kennt und auch keine Möglichkeit hat, sie durch Abhören von Leitungen auszuspionieren.

Der technische Trend in der Chipkartenpersonalisierung geht immer stärker in die Richtung eines kryptografisch vollständig abgesicherten Prozesses, der dann im Prinzip auch von kostengünstigen Dienstleistungsfirmen in nicht gesicherten Räumen erledigt werden kann. Es existieren mittlerweile auch Verfahren, bei denen der Personalisierer die kartenindividuellen Daten auf einer gebrannten CD-ROM erhält. Der Fertigungsdatensatz mit dem dazugehörigen kartenindividuellen Schlüssel ist dabei fest mit der unigen Chipnummer des Mikrocontrollers gepaart, sodass es einem Personalisierer beispielsweise nicht mehr möglich ist, Duplikate von Karten herzustellen, es sei denn, er könnte das Chipkarten-

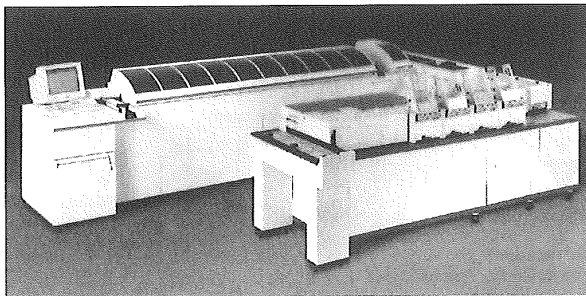


Bild 10.62 Modulares Personalisierungssystem Data Card 9000 mit u. a. integrierter Postverarbeitung. (Data Card)

¹¹ Siehe auch Abschnitt 6.6 „Sicherung der Datenübertragung“.

Betriebssystem manipulieren. Das Verfahren hat allerdings den Nachteil, dass bei fehlerhaften Chips nicht alle gelieferten Datensätze verwendet werden können, da die betreffenden Chips nicht mehr vorhanden sind. Bei diesem Verfahren muss deshalb auch eine Rückmeldung an die datengenerierende Stelle erfolgen, welche Chipkarten tatsächlich produziert wurden. Bei den heute üblichen Personalisierungsabläufen ist dies nicht notwendig, da eine fehlerhafte Karte problemlos nachproduziert werden kann. Dies ist im Übrigen auch der Grund, warum die Personalisierungsstätten der Kartenproduzenten immer Sicherheitsbereiche sind.

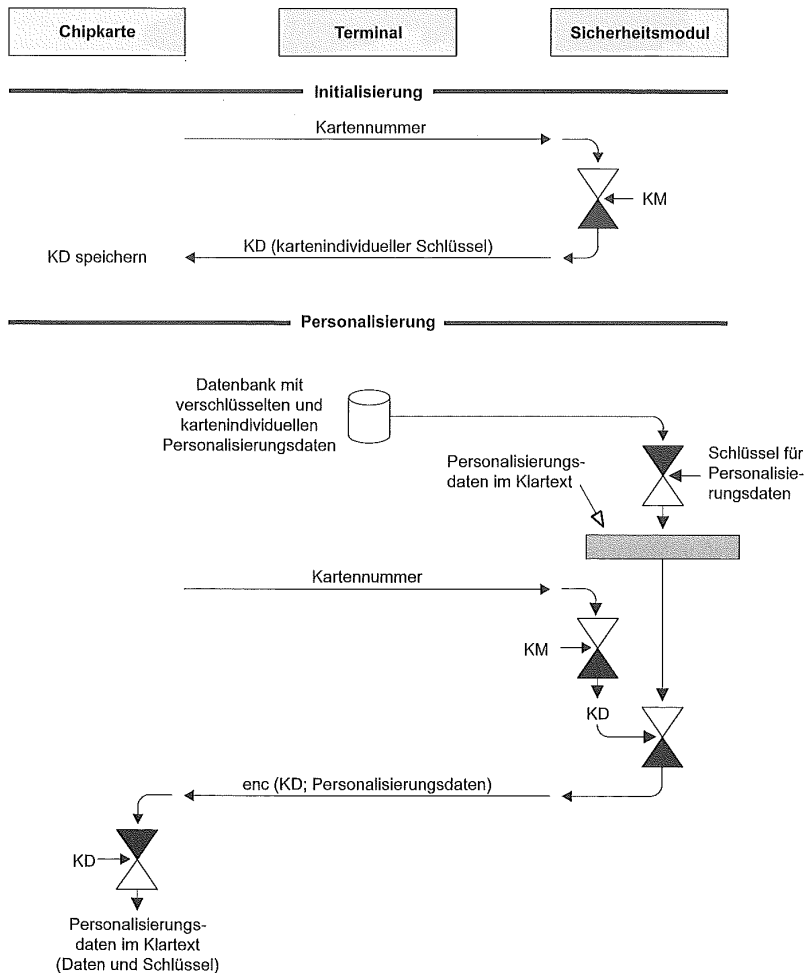


Bild 10.63 Schematische Darstellung eines üblichen Ablaufs von Initialisierung und Personalisierung mit kryptografischer Absicherung der Übertragung von Daten und Schlüsseln. KM bezeichnet den Hauptschlüssel, der zur Ableitung der kartenindividuellen Schlüssel KD verwendet wird. Es sind nur die kryptografisch relevanten Abläufe eingezeichnet.

Leider sind die im Personalisierungsbereich eingesetzten kryptografischen und sicherheitstechnischen Verfahren größtenteils geheim, weshalb hier keine bestimmte Anwendung beschrieben werden kann. Bild 10.63 zeigt jedoch beispielhaft aus kryptografischer Sicht den Ablauf einer Initialisierung mit anschließender Personalisierung. Beide Fertigungsschritte müssen in unterschiedlichen Räumen mit unterschiedlichem Personal stattfinden, damit die kryptografische Absicherung greifen kann.

Das Prinzip funktioniert dabei folgendermaßen: Bei der Initialisierung wird anhand einer einzigartigen Kartennummer und eines Hauptschlüssels KM in einem Sicherheitsmodul

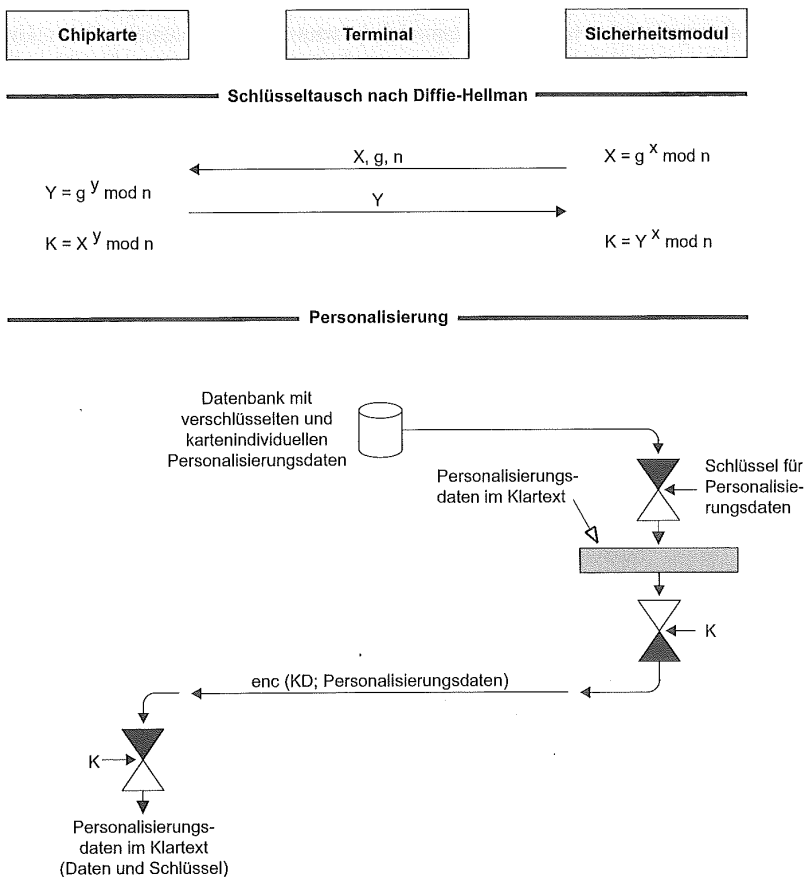


Bild 10.64 Schematische Darstellung eines möglichen Ablaufs der Personalisierung mit kryptografischer Absicherung der Übertragung von Daten und Schlüsseln. Bei diesem speziellen Ablauf werden die Schlüssel für das verschlüsselte Laden der Daten vorab mittels Diffie-Hellman-Schlüsselaustausch ausgehandelt. Damit ist es nicht mehr notwendig, in einem zur Personalisierung vorgelagerten und separierten Schritt symmetrische Schlüssel im Klartext in die Chipkarte einzubringen. Es sind nur die kryptografisch relevanten Abläufe eingezeichnet.

ein kartenindividueller Schlüssel KD abgeleitet und im Klartext zur Chipkarte gesendet, welche ihn abspeichert. Natürlich müssen bei der Initialisierung noch viele weitere Daten in die Chipkarte geschrieben werden, doch die Erzeugung und Speicherung des kartenindividuellen Schlüssels KD ist der einzige kryptografisch relevante Schritt.

Anschließend wird die Karte personalisiert, was entweder unmittelbar danach oder auch einige Wochen nach der Initialisierung erfolgen kann. Wichtig dabei ist nur, dass die Personalisierung vollständig von der Initialisierung getrennt ist, damit ein dort abgehörter KD während der Personalisierung nicht zur Entschlüsselung der kartenindividuellen Daten verwendet werden kann.

In der Personalisierung werden die mit einem einheitlichen Schlüssel verschlüsselten Personalisierungsdaten vom Sicherheitsmodul kartenweise entschlüsselt. Diese Umschlüsselung ist notwendig, da dem Generierer der Personalisierungsdaten die chipindividuelle Nummer nicht bekannt ist, weil diese vom Halbleiterhersteller unabhängig erzeugt wird. Aus der übergebenen Kartennummer und dem Hauptschlüssel KM errechnet das Sicherheitsmodul den kartenindividuellen Schlüssel KD. Nun haben das Sicherheitsmodul und die Chipkarte ein gemeinsames Geheimnis und nutzen dieses, d.h. den Schlüssel KD, zur Verschlüsselung der Personalisierungsdaten. Diese werden nun an die Chipkarte verschlüsselt übertragen, dort entschlüsselt und an die entsprechenden Stellen im EEPROM geschrieben. Dieses Verfahren ist ein vollständiger kryptografischer Schutz des Personalisierungsvorgangs. Er bewahrt die zu personalisierenden Daten vor Ausspähung, solange der Schlüssel KD, der während der Initialisierung in die Karte geschrieben wird, geheim bleibt.

Bild 10.64 zeigt eine alternative Methode der sicheren Einbringung von Daten in Chipkarten. Dabei wird als erster Schritt mit dem Schlüsselaustauschverfahren nach Diffie-Hellmann zwischen Chipkarte und Terminal ein gemeinsamer geheimer Schlüssel vereinbart. Anschließend überträgt man mit diesem Schlüssel die Daten verschlüsselt zur Chipkarte. Der große Vorteil dabei ist, dass nie ein geheimer Schlüssel unverschlüsselt übertragen wird.

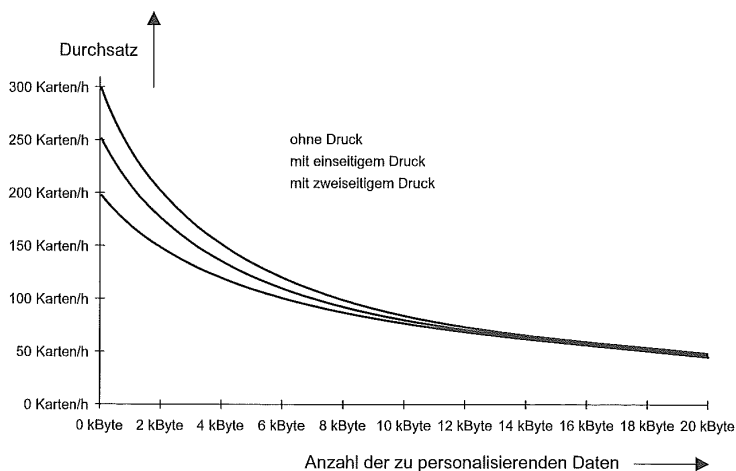


Bild 10.65 Durchsatzdiagramm für elektrische Personalisierung in Verbindung mit ein- oder zweiseitigem Kartendruck bei einer Desktop-Personalisierungsmaschine.

Am Ende des Personalisierungsprozesses werden zur Qualitätssicherung auf der Personalisierungsmaschine noch verschiedene Prüfungen der fertigen Chipkarte durchgeführt. So wird beispielsweise auf den neuesten Maschinen die Chipkarte mit einer Kamera aufgenommen, die optische Personalisierung mit einem Computer ausgewertet und mit der Produktionsdatenbank gegengeprüft. Im Fehlerfall wird diese Karte in ein Fehlerfach angesteuert und automatisch eine neue Karte nachproduziert. In der Regel findet auch noch eine Prüfung der personalisierten Daten im Mikrocontroller statt. Dies ist jedoch technisch schwer zu bewerkstelligen, da auf viele Dateien kein lesender Zugriff mehr möglich ist. Für diese Personalisierungstests findet man deshalb oft spezielle Sicherheitsmodule an den Personalisierungsmaschinen, die geheime Hauptschlüssel enthalten. Damit lassen sich die in die Chipkarten personalisierten Schlüssel auf Korrektheit prüfen, etwa durch Authentisierung.

Bei einem anderen Verfahren erhält der Personalisierer für jede Chipkarte individuelle Kommando- und dazugehörige Antwortstrings. Diese Kommandos sendet er dann in der richtigen Reihenfolge zur Chipkarte und vergleicht die von der Karte zurückgegebene Antwort mit der erhaltenen Vorgabe. Bei Ungleichheit verhält sich die Chipkarte nicht wie erwartet, und es muss ein Personalisierungsfehler aufgetreten sein. Bei dieser Methode erübrigt sich ein spezielles Sicherheitsmodul für die Tests in der Personalisierung.

Sind Chipkarten einmal personalisiert, kann dies im Regelfall nie mehr rückgängig gemacht werden, was bedeutet, dass fehlerhaft personalisierte Chipkarten Müll sind. Am fehleranfälligsten ist die elektrische Personalisierung; wenn bei einer größeren Charge ein Fehler passiert, sind der finanzielle Schaden und der Zeitverzug groß. Deshalb gibt es vereinzelt Chipkarten-Betriebssysteme, bei denen die Möglichkeit besteht, die komplette Personalisierung nach einer entsprechenden Authentisierung vollständig zu löschen. Seitens des Betriebssystems verhalten sich diese Chipkarten dann wie nach der Halbleiterfertigung oder nach der Komplettierung. Bei Testkarten wird diese Funktion manchmal benutzt, da man dann die Software in den Chipkarten ändern kann, ohne diese jedes Mal auszumustern. Verschiedentlich werden solche Mechanismen des Chipkarten-Betriebssystems auch für Serienkarten freigeschaltet, sodass man gegebenenfalls eine Depersonalisierung vornehmen könnte.

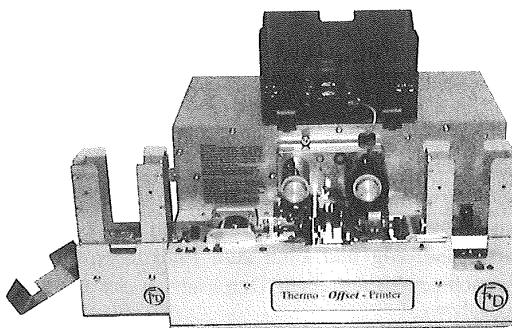


Bild 10.66 Beispiel für eine Desktop-Personalisierungsanlage für elektrische Personalisierung und doppelseitigen Farbdruck mit 300 dpi Auflösung. Auf der rechten Seite befindet sich der Karteneingangsstapel, auf der linken Seite der Stapel für die Gutkarten und das Fach für die Schlechkarten. (F&O)

In der Regel ist die Personalisierung von Chipkarten ein Vorgang, der typischerweise ab zehntausend Karten aufwärts anfängt. Bei vielen Anwendungen muss jedoch gewährleistet sein, dass auch einzelne und kundenindividuelle Karten nachproduziert werden können. So muss etwa eine kaputte oder verloren gegangene ec-Karte mit Chip innerhalb weniger Tage ersetzt werden können, da sonst der Kartenbenutzer kein Geld mehr am Geldausgabeautomaten abheben kann. Diese Just-In-Time-Personalisierung gewinnt mit steigender Kundennähe immer mehr an Gewicht. Sie wird in der Regel auf einer Fertigungslinie neben der Massenfertigung eingerichtet, erhält die Kartendaten per Datenfernübertragung und verwendet bereits initialisierte Chipkarten als eine Art Halbzeug. Mit dieser Art der Kartenproduktion kann garantiert werden, dass der Endkunde, also der Kartenbesitzer, bei Bedarf innerhalb von 24 Stunden eine neue Karte in Händen hält. Selbstverständlich ist so eine antwortzeitoptimierte Fertigung nicht für die Massenproduktion von Chipkarten geeignet.

Kuvertieren und Versenden

Der letzte Fertigungsschritt in der Kartenproduktion ist das Kuvertieren und Versenden der Chipkarten. Nicht bei allen Kartentypen, wie etwa bei Telefonkarten, ist dies notwendig, da die Lieferung oft als Massengut direkt zum Kartenherausgeber erfolgt. Für die aufwendigeren und teureren Karten ist es jedoch die Regel, dass der Karteninhaber per Post ein persönliches Anschreiben mit seiner neuen Karte erhält. Bei manchen Anwendungen, wie beispielsweise bei Kreditkarten, erhält der Kartenbenutzer auch noch einen Brief mit seiner PIN. Dieser wird aus Sicherheitsgründen mit getrennter Post und zeitlich um einige Tage versetzt verschickt. Der Bereich in der Kartenfertigung, der alle diese Verrichtungen ausführt, wird oft auch Lettershop genannt.

Der PIN-Brief selber ist so aufgebaut, dass sich in seinem Innern ein Blatt Papier befindet, das sich über ein an der Innenseite des PIN-Briefs befindliches Kohlepapier von außen mit einem Nadeldrucker bedrucken lässt. Der Brief ist so gestaltet, dass unbefugtes Lesen der gedruckten PIN ohne sichtbare Beschädigung des Umschlags nicht möglich ist. Durch diese Mechanismen wird sichergestellt, dass das unbemerkte Ausspähen der PIN selbst bei der Erstellung des PIN-Briefs nicht möglich ist. Hochleistungsdrucksysteme für PIN-Briefe können bis zu 34000 Dokumente pro Stunde drucken.

Für den Versand der Karten werden je nach Kartentyp die persönlichen Daten wie Name und Anschrift aus der Chipkarte ausgelesen oder aus der Fertigungsdatenbank geholt und auf einen Kartenträger, d. h. einen vorgedruckten Brief, mit einem Hochleistungslaserdrucker aufgedruckt. Der Brief kann zwei gestanzte Laschen haben, in denen die Chipkarte eingeschoben und fixiert ist. Alternativ zu dieser Technik wird oft auch ein rückstandsfrei lösbares Adhäsionsklebeband verwendet. Anschließend folgen die Faltung dieses Kartenträgers und die Kuvertierung. Nach der folgenden Frankierung ist die Chipkarte mit dem persönlichen Anschreiben versandfertig und kann zum Endkunden verschickt werden. Leistungsfähige Kuvertiermaschinen haben einen Durchsatz in der Größenordnung von 7000 Briefen pro Stunde.

Als letzte Qualitätskontrolle werden die fertig erstellten Briefe mit der in ihnen enthaltenen Karte noch einzeln automatisch gewogen. Das Kartengewicht von ca. 6 Gramm reicht mühelos aus, um zweifelsfrei sicherzustellen, dass jeder Brief auch eine Karte enthält.

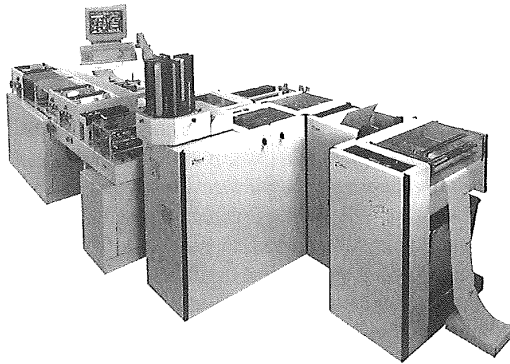


Bild 10.67 Anlage für das Applizieren von Karten auf dazugehörigen Anschreiben. Diese werden dann, falls erforderlich, mit zusätzlichen Beilagen versehen und kuvertiert. Die Maschine besitzt eine Kapazität von 7 000 Kuvertierungen pro Stunde. (Böwe Systec)

Um die Portokosten so niedrig wie möglich zu halten, ist es üblich, die Briefe nach Postleitzahlen vorsortiert der Post zu übergeben. Diese Postoptimierung ist am einfachsten, wenn die Produktionsreihenfolge der Chipkarten in der Fertigung nach dem von der Post verlangten Sortierkriterium (üblich sind Postleitzahlen und Zustellbezirk) ausgerichtet ist.

Die Praxis fördert auch bei so einfachen Dingen wie dem Postversand immer wieder interessante Problemstellungen zutage. So traten bei einem großen Chipkartenhersteller eines Tages plötzlich Ausfälle von per Briefpost versendeten Chipkarten auf. Nach einigen Tagen Ursachenforschung stellte sich heraus, dass im zuständigen Postverteilzentrum die Sortiermaschinen mit einer neuen Anordnung der Förderrollen umgestellt wurden. Diese hatten nun beim Sortieren die Briefe mit den Chipkarten so stark gebogen, dass es bei einigen Karten zum Bruch des Chips im Modul kam. Das Problem konnte gelöst werden, indem die Chipkarten auf dem Träger um einige Zentimeter versetzt wurden. Aus diesen und ähnlichen Gründen werden vor großen Versandaktionen oft einige hundert Testbriefe ins Zielgebiet versendet und anschließend analysiert, um sicherzustellen, dass die Chipkarten in den Briefen nicht auf dem Transport oder beim Sortieren beschädigt werden.

Bauteil/ Fertigungsschritt	Kostenanteile einer Chipkarte mit: ≈ 6 kByte ROM ≈ 1 kByte EEPROM ≈ 128 Byte RAM)	Kostenanteile einer Chipkarte mit: ≈ 16 kByte ROM ≈ 8 kByte EEPROM ≈ 256 Byte RAM
Die	50,0 %	65,0 %
Modul	25,0 %	15,0 %
Kartenkörper	12,5 %	10,0 %
Initialisierung/ Personalisierung	12,5 %	10,0 %

Bild 10.68 Überblick der Kostenanteile bei zwei Typen von Chipkarten mit unterschiedlich leistungsfähigen Mikrocontrollern.

Die auf den vorangegangenen Seiten beschriebenen Fertigungsschritte und -phasen zeigen den Ablauf einer Massenproduktion, wie er z. B. für eine GSM-Karte oder Kreditkarte mit Chip üblich ist. Andere Anwendungen oder Kartenherausgeber erfordern bei der Produktion auch andere Prämissen. So werden zum Teil Chipkarten für GSM vor Ort im Geschäft personalisiert und dann direkt dem Kunden übergeben. Aus Kundensicht drückt dies natürlich Kompetenz und Leistungsfähigkeit aus, wenn er seine persönliche Chipkarte sofort nach der Anmeldung und Bezahlung erhält. Es ist dies aber letztendlich auch ein vom jeweiligen Marketing und den Sicherheitsvorgaben des Kartenherausgebers stark beeinflusster Sachverhalt. Im Gegensatz zu Vorgenanntem ist die Kartenkörper- und Modulherstellung im Prinzip unabhängig vom späteren Kartenherausgeber oder dessen Vertriebsaspekten und daher auch weitgehend bei allen Anwendungen gleich.

10.5 Phase 4 des Lebenszyklus im Detail

Die Phase 4 einer Chipkarte ist dem normalen Kartenbenutzer aus dem täglichen Umgang mit seinen Karten bekannt. Neue Anwendungen werden geladen oder freigeschaltet und bereits auf der Karte befindliche Anwendungen unter Umständen deaktiviert. Da der Großteil dieses Buches von Phase 4 handelt, wird – außer bei den Kartenmanagementsystemen – hier nicht mehr näher darauf eingegangen.

Kartenmanagementsysteme

Verwaltungssysteme für Karten sind bei verschiedenen Kartenherausgebern bereits seit vielen Jahren im Einsatz. Dabei lag der Fokus bisher jedoch auf der Bestandsverwaltung und Zuordnung der Karten zu bestimmten Personen. Mit der Verbreitung von Chipkarten, die in der Lage sind, Anwendungen zu ändern, nachzuladen oder zu löschen, erweitern sich die Aufgaben von Verwaltungssystemen für Karten grundlegend, da nun auch die Aspekte von kartenindividuellen Anwendungen abgedeckt werden müssen. Man bezeichnet diese Art von Systemen auch als Kartenmanagementsystem (CMS – *card management system*), Applet-Managementsysteme (AMS) oder zum Teil auch als Dateimanagementsysteme (FMS – *file management system*). Im Folgenden wird der Begriff Kartenmanagementsystem verwendet.

Die Voraussetzung für eine funktionierende Kartenverwaltung ist zum einen eine leistungsfähige Datenbank mit allen notwendigen Informationen über die ausgegebenen Karten und zum anderen eine zumindest zeitweise Online-Anbindung der zu verwaltenden Karten. Aus diesem Grunde eignen sich für Kartenmanagementsysteme auch sehr gut die bestehenden Chipkarten in der Telekommunikation, da sie bei der Benutzung ständig online mit dem Hintergrundsystem verbunden sind. Im zum Teil offline arbeitenden Zahlungsverkehr kann man jedoch beispielsweise die temporäre Online-Anbindung der Chipkarten im Geldausgabeautomaten oder Händlerterminal nutzen. Essenzielle Voraussetzung für jede Online-Anbindung ist eine sichere Ende-zu-Ende-Verbindung zwischen Chipkarte und dem Verwaltungssystem.

Die Aufgaben eines Kartenmanagementsystems können eine sehr große Bandbreite abdecken. Im einfachsten Fall werden Daten in Dateien von bestimmten Chipkarten aktualisiert (RFM – *remote file management*). Dazu verwendet man die üblichen Chipkarten-Kommandos, die über einen sicheren Kanal zur Chipkarte übertragen werden. Etwas aufwendiger ist die Verwaltung von Dateien, also das Löschen bestehender und Anlegen neuer Dateien, wobei die Mechanismen analog dem der Aktualisierung von Dateiinhalten sind.

Deutlich größere Dateimengen fallen an, wenn auf einer Chipkarte neue Anwendungen angelegt werden sollen. Sind diese dateibasiert, müssen alle entsprechenden Dateien auf der Chipkarte erzeugt und mit Daten gefüllt werden. Ist die neue Anwendung Programmcode-basiert, so muss das Programm in die Chipkarte geladen werden. Im Falle von Java Card kommt dabei vor allem der OP-Lader¹² zum Einsatz. Es kann jedoch auch vorkommen, dass Anwendungen durch andere oder neue Versionen ersetzt werden müssen. Vorbereitend dazu sind zuvor die Daten der Anwendungen auf der Chipkarte zu sichern. Anschließend muss die alte Anwendung gelöscht, die neue Anwendung auf der Chipkarte erzeugt werden und schlussendlich sind die gesicherten Daten, evtl. noch auf neue Formate konvertiert, in die neue Anwendung einzubringen.

Bei den beschriebenen Aufgaben eines Kartenmanagementsystems wurde die Zeitspanne nach Ausgabe der Chipkarten an den Endkunden betrachtet. Man kann dies jedoch noch deutlich ausweiten, sodass ein Kartenmanagementsystem die gesamte Lebenszyklus-Verwaltung (*life cycle management*) der Chipkarte durchführt. Dies beginnt bei der Komplettierung des Chipkarten-Betriebssystems, erstreckt sich über Initialisierung und Personalisierung bis hin zur Benutzung der Chipkarte und der irgendwann notwendigen Deaktivierung mit Übertragung der Daten auf eine neue Chipkarte.

Diese enorme Vielfalt an Aufgaben führt natürlich auch zu einer erheblichen Komplexität eines Kartenmanagementsystems. Des Weiteren sollte man nicht außer Acht lassen, dass in der Praxis in den seltensten Fällen eine homogene Masse von Chipkarten zu verwalten ist. Der Regelfall wird dabei ein höchst heterogenes Sammelsurium von unterschiedlichen Chipkarten-Betriebssystemen in mehreren Versionen auf Hardware-Plattformen mit differierenden Speichergrößen sein. Auch die zu administrierenden Anwendungen werden ebenfalls eine gewisse Bandbreite an Versionen haben.

Als griffiges Beispiel zur Darstellung der Komplexität könnte man einen Telekommunikationsbetreiber annehmen, der SIMs mit 3 Betriebssystemversionen auf 3 Hardwareplattformen mit 3 unterschiedlichen Versionen einer Anwendung im Feld hat. Im ungünstigsten Fall muss hier das Kartenmanagementsystem $27 (= 3^3)$ unterschiedliche Zugriffe auf die Anwendung durchführen. Für den Kartenbenutzer stellen sich diese 27 Varianten jedoch nur als eine einzige Anwendung auf seiner SIM dar.

Neben dieser sich sehr schnell einstellenden enormen Vielfalt an Varianten müssen auch in der zu verwaltenden Chipkarte gewisse Rahmenbedingungen gegeben sein. Im Prinzip muss der gesamte Administrationsvorgang durch das Kartenmanagementsystem atomar durchgeführt werden, denn wenn durch einen irgendwie bedingten Abbruch die Administration nicht vollständig durchgeführt werden kann, muss der ursprüngliche Zustand wiederhergestellt werden können. Man denke dabei etwa an das Nachladen eines Java-Applets in die SIM über die Luftschnittstelle. Bricht die Verbindung ab, weil z. B. in einem Tunnel ein Funkloch auftritt, dann darf dies nicht zu irgendwelchen Auswirkungen auf die bestehende SIM-Funktionalität führen. Dies alles ist technisch mit heutigen Mechanismen und Verfahren lösbar, doch ist der dafür notwendige Aufwand erheblich.

Es gibt auf dem Markt Kartenmanagementsysteme, welche verschiedene oben genannte Aufgaben erfüllen. Werden jedoch großflächig bei einer Anwendung Chipkarten eingesetzt,

¹² Siehe auch Abschnitt 5.11 „Open Platform“.

auf denen Anwendungen, gleich welcher Art, dynamisch verwaltet werden sollen, müssen die aktuellen Kartenmanagementsysteme an einigen Stellen noch erheblich erweitert werden.

10.6 Phase 5 des Lebenszyklus im Detail

Die Phase 5 des Chipkarten-Lebenszyklus nach ISO 10202-1 legt alle Maßnahmen beim Ende der Kartenbenutzung fest. Im Detail sind dies die Deaktivierung der Anwendung(en) auf der Chipkarte und anschließend die Deaktivierung der Chipkarte selber. Diese beiden Prozesse sind jedoch bei den meisten Chipkarten schlichte Theorie. Die Karten werden in der Praxis entweder in den Müll geworfen oder von Sammlern sorgfältig beschriftet auf unbestimmte Zeit in Ordnern eingelagert. Eine Rückgabe an den Kartenherausgeber ist in der Regel eher selten.

Es gäbe jedoch durchaus Kommandos für das Deaktivieren einzelner Anwendungen oder der gesamten Chipkarte. So sind die ISO/IEC 7816-9 Kommandos DELETE FILE, DEACTIVATE FILE, TERMINATE DF oder TERMINATE CARD USAGE explizit dafür vorgesehen, den letzten Teil des Lebenszyklus von Anwendungen einzuläuten.¹³

Diese Kommandos sind vor allem für Multiapplication-Karten zur Verwaltung der einzelnen Anwendungen unabdingbar, doch bei den heutigen Chipkarten, die meist doch nur mehr oder minder eine einzige Anwendung tragen, werden sie selten verwendet. Die einfachste und von jedermann problemlos durchführbare Methode ist natürlich das Auseinanderschneiden der Karte, die von manchen Herausgebern auch zur „Terminierung“ der Karte empfohlen wird.

Aus Gründen der Systemsicherheit hätte jedoch die Rückgabe der Chipkarten in manchen Fällen durchaus seine Berechtigung. Die Karten enthalten zum Teil noch gültige geheime Schlüssel. Falls ein potenzieller Angreifer im Besitz von mehreren hundert oder gar tausend Karten ist, hat er eine wesentlich größere Datengrundlage für die Analyse von Hard- und Software, als dies bei einigen wenigen Karten der Fall wäre. Statistische Untersuchungen auf der Grundlage großer Stückzahlen sind immer wesentlich aufschlussreicher als bei einzelnen Chipkarten.

Manche Kartenherausgeber gehen deshalb und auch aus bekannten umweltpolitischen Erwägungen dazu über, bei der Ausgabe neuer Karten die alten, abgelaufenen wieder einzusammeln. Oft befinden sich auch in der Nähe von Kartentelefonen Sammelbehälter für verbrauchte Telefonkarten. Erst dieses Zusammentragen von Karten schafft die Grundlage für ein funktionsfähiges Recycling.

Recycling

Zum Recycling von Chipkarten muss man ehrlicherweise anmerken, dass es nicht sehr weit fortgeschritten ist. Einerseits gibt es zurzeit einfach nicht genügend eingesammelte Chipkarten für einen Recyclingprozess, und andererseits ist die zu recycelnde Menge an Kartenmaterial auch nicht sehr groß. Im Jahr 1997 wurden weltweit etwa 40 000 Tonnen Kunststoff für die Herstellung von Karten verwendet. Selbst unter der (völlig idealistischen) Annahme, dass die gleiche Gewichtsmenge an Karten ausgemustert und vollständig in einen Recyclingprozess rückgeführt werden kann, ist dies gegenüber der Weltkunst-

¹³ Siehe auch Abschnitt 7.8 „Kommandos zur Verwaltung von Dateien“.

stoffproduktion, die alleine bei PVC ungefähr 13 Millionen Tonnen im selben Jahr betrug, eine verschwindend geringe Menge.

Mit steigender Verbreitung der Karten wird sich dies aber ändern. Recycling ist gerade bei Chipkarten ein problematisches Thema. Die aus mehreren Einzellagen aus teilweise verschiedenen Kunststoffen laminierten Chipkarten bilden ein höchst heterogenes Material. Außerdem sind die Karten mit unterschiedlichen Farben bedruckt und mit Hologrammen, Unterschrifts- und Magnetstreifen versehen, was zu weiteren Stoffen in dem Mischmaterial führt. Überwiegend sortenreine Stoffe fallen lediglich in der Kartenproduktion an, beispielsweise als Verschnitt beim Stanzen der Kartenkörper aus einer Einschnittfolie. Diese Kunststoffe können dann verhältnismäßig einfach wieder verwendet werden, was bei vielen Kartenherstellern auch der Fall ist.

Bei Chipkarten, die nicht mehr verwendet werden, ist jedoch eine Trennung in sortenreine homogene Stoffe momentan praktisch unmöglich. Bei den bislang vorgeschlagenen Verfahren ist geplant, das Modul aus der Karte zu stanzen und den restlichen Kartenkörper dann zu shreddern. Die dabei entstehenden Kunststoffschrottel können dann für niederwertige Kunststoffteile weiterverwendet werden. Gartenzweige sind ein typisches Beispiel für diese Art des Recyclings. Die Module werden ebenfalls fein zerkleinert und die in ihnen enthaltenen Metalle auf elektrolytischem Wege wiedergewonnen. Man setzt diese Verfahren großtechnisch aber momentan nirgends ein. Zudem ist nicht eindeutig sichergestellt, ob diese Art von aufwendigem Recycling wirklich umweltschonender ist als eine bloße Verbrennung oder Deponierung.

Bei kontaktlosen Chipkarten mit im Kartenkörper eingebetteten Spulen (Kupferdraht oder elektrisch leitende Farbe) ist *de facto* eine sortenreine Trennung in die einzelnen Kunststoffe nicht mehr möglich.

In der Praxis bleibt deshalb vor allem bei Mehrlagenkarten nur der Weg, sie bei hoher Temperatur einer Verbrennung zuzuführen, was manche Leute oft großspurig als Energierecycling bezeichnen. Ist die Temperatur dabei genügend hoch, entstehen auch wenig Schadstoffe. Ob diese Lösung langfristig als akzeptabel angesehen wird, bleibt abzuwarten. Jedenfalls ergeben 1 Million Chipkarten bei einem Einzelgewicht von 6 Gramm ein Gesamtgewicht von 6 Tonnen.

Bauteil	Material	Gewicht
Kartenkörper	diverse Kunststoffe (z. B.: PVC, PC, ABS)	4,400 g
Farben auf Kartenkörper	Harze, Pigmente	sehr gering
Magnetstreifen	Eisenoxid o. Ä., Farbe, Klebstoff	sehr gering
Hologramm	Aluminium, Klebstoff	sehr gering
Mikrocontroller (10 mm ²)	Silizium mit verschiedenen Dotierungen	0,009 g
Bonddrähte	Gold oder Aluminium	sehr gering
Vergussmasse für Mikrocontroller	Epoxidharz	0,010 g
Kleber für Modul in Kartenkörper	Epoxidharz	sehr gering
Modul mit 6 Kontaktf Flächen	Epoxidharz, Glasfaser, Nickel, Aluminium, Gold	0,170 g
Modul mit 8 Kontaktf Flächen	Epoxidharz, Glasfaser, Nickel, Aluminium, Gold	0,180 g

Bild 10.69 Überblick der wesentlichen Gewichtsanteile der Bauteile von Chipkarten.

11 Chipkarten-Terminals

11.1 Mechanische Eigenschaften	666
11.2 Elektrische Eigenschaften.	670
11.3 Sicherheitstechnik	671
11.4 Anbindung von Terminals an übergeordnete Systeme	674
11.4.1 PC/SC	674
11.4.2 OCF	678
11.4.3 MKT	678
11.4.4 MUSCLE	679

Die einzige Verbindung zur äußeren Welt bei Chipkarten ist die serielle Schnittstelle. Es gibt keine andere Möglichkeit des Datenaustausches. Deshalb benötigt man ein zusätzliches Gerät, das die elektrische Verbindung zur Chipkarte herstellt. In diesem Buch wird dieses Gerät einheitlich als „Terminal“ bezeichnet. Es existieren jedoch auch noch andere Bezeichnungen, wie IFD (*interface device*), CAD (*chip accepting device*), CCR (*chip card reader*), Smart Card Reader¹ oder Smart Card Adapter. Die Grundfunktionalität, nämlich die Chipkarte elektrisch zu versorgen und eine datentechnische Verbindung herzustellen, ist aber bei all diesen Geräten gleich.

Sobald Terminals technisch etwas komplizierter aufgebaut sind als bloße Kontaktierungseinheiten, Spannungswandler und Takterzeuger, verfügen sie über einen eigenen Prozessor (üblicherweise 8/16-Bit-Architektur) und einen dazugehörigen Speicher. Der Prozessor kann bei einfachen Geräten Teil eines Mikrocontrollers sein, ist aber oft nur Bestandteil eines Einplatinencomputers. Programmiert werden Terminals in der Regel nur vom Terminalhersteller in der Programmiersprache C, C++ oder Java [JavaPOS]. Im Bereich Mobiltelefone, die ebenfalls Chipkarten-Terminals sind, wird in Zukunft eine Variante von Java (J2ME – *Java 2 Micro Edition*) große Bedeutung für Anwendungsprogrammierung gewinnen.

Die Problematik bei der Terminalprogrammierung durch Dritte ist indessen die gleiche wie bei Chipkarten mit ausführbarem Programmcode, sodass wohl auch die Lösungen in die gleiche Entwicklungsrichtung münden werden. Europays Open Terminal Architecture (OTA) mit einem Forth Interpreter war 1996 einer der ersten Ansätze dazu [Europay 96], und Java für Terminals ist der nächste Schritt. In der EMV-Spezifikation ist bereits explizit ein Konzept für nachladbaren Programmcode verankert.

Terminals besitzen keine eigene Festplatte und müssen deshalb ihre Programme und Daten in batteriegepuffertem RAM, EEPROM oder Flash-EEPROM ablegen. Der dabei zur Verfügung stehende Speicher bewegt sich in der Regel in der Größenordnung von einigen wenigen Megabytes.

Im Gegensatz zu Chipkarten, die alle technisch ähnlich aufgebaut sind, gibt es bei den Terminals sehr unterschiedliche Ausführungen. Man unterscheidet grundsätzlich zwischen tragbaren und stationären Terminals. Die tragbaren beziehen ihre Energie aus Batterien und die stationären vorzugsweise aus dem Stromnetz oder über die Datenschnittstelle.

¹ Der Begriff Kartenleser oder Smart Card Reader ist nicht so zu verstehen, dass damit aus Chipkarten nur Daten gelesen werden können. Ein Schreibvorgang ist natürlich ebenfalls möglich.

Auch lassen sich Terminals nach ihrer vorhandenen Benutzerschnittstelle einteilen. Gerade bei tragbaren Geräten ist oft zusätzlich ein Display mit einer einfachen Tastatur vorhanden, sodass man auf die wichtigsten Funktionen direkt vor Ort zugreifen kann.

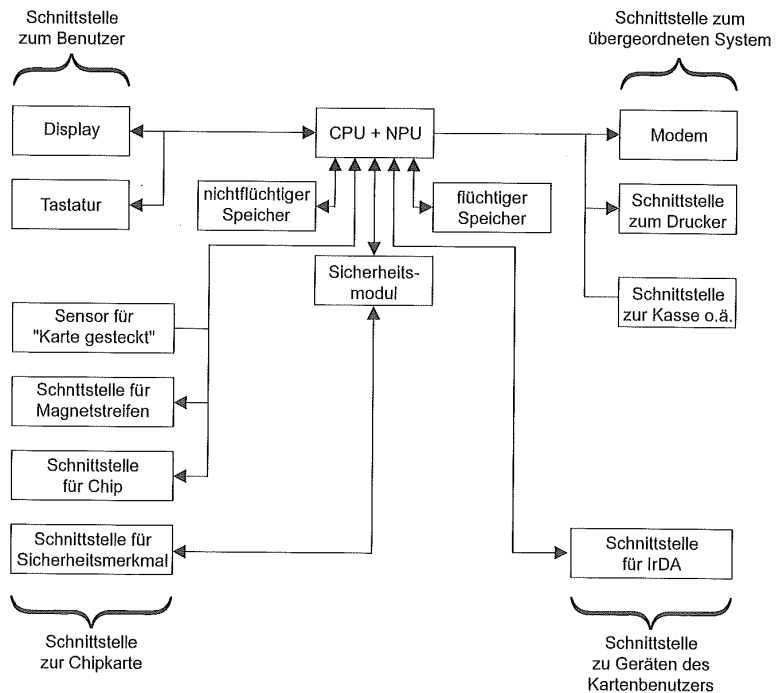


Bild 11.1 Typische Architektur eines Chipkartenterminals mit Display, Tastatur, Magnetstreifenleser und Sicherheitsmodul. Terminals in dieser Ausstattung werden häufig an Kassen für die Bezahlung mit unterschiedlichsten Kartentypen (Kredit-, Debitkarte, elektronische Geldbörsen) verwendet. Die Tastatur kann unter Umständen auch speziell gegen Manipulationen gesichert sein (PIN-Pad). Die Abbildung zeigt grundlegende Informations- und Energieflüsse und ist kein Stromlaufplan.

Stationäre Terminals sind zwar oft ebenfalls mit Display und Tastatur ausgestattet, doch auch noch zusätzlich mit einer permanenten Verbindung zu einem übergeordneten Rechnersystem. Ist am Terminal keine Mensch-Maschine-Schnittstelle vorhanden (d. h. kein Display und keine Tastatur), muss zwangsläufig eine direkte Verbindung zu einem Rechner bestehen, um so die Verbindung Chipkarte-Benutzer herzustellen.

In einer Spezifikation des deutschen ZKAs existiert eine allgemeine und sehr praxisnahe Charakterisierung von Terminalklassen. Dabei gibt es vier Klassen. Ein Klasse 1 Terminal ist demnach ein Terminal, das im Wesentlichen aus einer Kontaktiereinheit ohne weitere Funktionselemente sowie einer Schnittstelle (z. B. USB) zu anderen Systemen besteht. Die Klasse 2 schließt alle Funktionalitäten von Klasse 1 ein mit dem Zusatz eines Displays. Die nächsthöhere Klasse 3 besitzt zusätzlich zu den Klasse-2-Elementen noch eine Tasta-

tur. Klasse 4 ist die aufwendigste Klasse mit allen Funktionselementen von Klasse 3 sowie einem Sicherheitsmodul (HSM) mit RSA-Funktionalität.

Tabelle 11.1 Die Einteilung der Terminalklassen nach ZKA

Klasse	Funktionselemente
1	Kontaktiereinheit, Schnittstelle zu anderen Systemen
2	Klasse 1 Funktionselemente + Display
3	Klasse 2 Funktionselemente + Tastatur
4	Klasse 3 Funktionselemente + Sicherheitsmodul

Vereinzelnd findet man Terminals, die mit einer IrDA (*infrared data association*) Infrarotschnittstelle oder Bluetooth ausgestattet sind. Diese können für die direkte Kommunikation zwischen Terminal und etwa digitalen Assistenten (*personal digital assistant – PDA*) oder Mobiltelefonen benutzt werden. Der Vorteil ist: Man kann davon ausgehen, dass das eigene Gerät vertrauenswürdig ist und beispielsweise die PIN nicht mehr an einem fremden Terminal eingegeben werden muss.

Die Aufteilung in portable und stationäre Terminals führt zu einem weiteren Unterscheidungsmerkmal, nämlich der Art, wie sie betrieben werden: Ein Online-Terminal hat während des Betriebes eine ununterbrochene Verbindung zu einem entfernten Computer, der dann einen Teil der Steuerung übernimmt. Die typischen Vertreter dieser Gattung sind Terminals für Zutrittskontrolle, da sie vollständig von einem permanent verbundenen Hintergrundsystem angesteuert werden.

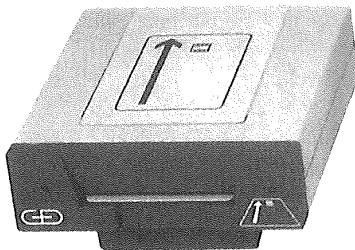


Bild 11.2 Typisches Chipkarten-Terminal für Anbindung über eine serielle Schnittstelle an einen Rechner. (Giesecke & Devrient – CCR2)

Das Gegenstück dazu, die Offline-Terminals, arbeiten völlig autark in Bezug auf ein übergeordnetes System. Allerdings gibt es zwar viele Online-Terminals, doch praktisch keine reinen Offline-Geräte. Zumindest gelegentlich tauschen Offline-Terminals über eine Datenleitung Informationen mit einem Hintergrundsystem aus, und sei es nur, um eine neue Sperrliste oder ein Update der Terminalsoftware anzufordern.

Die Verbindungsstrecke zwischen Terminal und einem weiteren Rechner ist bei typischen Anwendungen innerhalb eines Gebäudes eine elektrische Leitung oder ein Lichtwellenleiter. Es kann aber, wie im Bereich des Zahlungsverkehrs bei Kassenterminals, auch



Bild 11.3 Beispiel für ein tragbares Chipkarten-Terminal für den elektronischen Zahlungsverkehr mit Kreditkarten, Debitkarten und elektronischen Geldbörsen. Das Terminal besitzt ein integriertes Sicherheitsmodul, einen Drucker und ist Offline-fähig. (Giesecke & Devrient – ZVT 900)

eine Telefonleitung zum nächsten Rechenzentrum sein. Ob nun dieses vom Terminal bei Bedarf angewählt wird oder ob eine Standleitung geschaltet ist, hängt von der jeweiligen Anwendung ab. Da die Bereitstellung von Standleitungen aber teuer ist, geht man aus Gründen der Kostenreduktion immer mehr dazu über, Telefonleitungen nur bei Bedarf zu nutzen, d. h. das Terminal muss dann mit einem Wählmodem ausgerüstet sein.

In die vorangegangene Klassifizierung lassen sich Chipkarten-Terminals in Form einer PC-Card (ex PCMCIA-Karte) nur sehr schwierig unterbringen. Sie können sowohl online als auch offline betrieben werden und lassen sich sowohl an Arbeitsplatzrechnern als auch in Notebook-Computern einsetzen. Im Prinzip sind sie nur eine einfache und meist auch preisgünstige Hardwareschnittstelle zwischen Chipkarte und Computer. Voraussetzung für die Verwendung ist lediglich ein PC-Card-Steckplatz, der je nach Hersteller vom Typ I (3,3 mm Bauhöhe) oder vom Typ II (5 mm Bauhöhe) sein muss. Manche PC-Card-Chipkarten-Terminals besitzen zusätzlich zum Chipkarteninterface einen internen Ergänzungsspeicher für die Chipkarte und Recheneinheiten für Massendatenver- und entschlüsselung. Diese nur wenige Millimeter hohen Terminals sind die wohl am flexibelsten einsetzbaren überhaupt, da sie Chipkarten zum Teil völlig neue Anwendungsfelder erschließen. So ist es

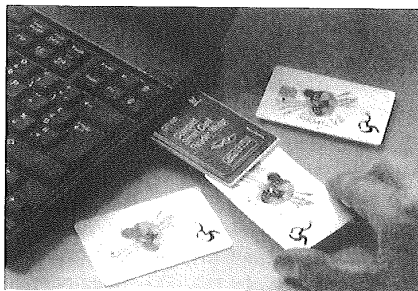


Bild 11.4 Typisches Chipkarten-Terminal in Form einer PC-Card. (Gemplus – GPR400)

mit ihnen ohne zusätzliche Kabel, Spannungsversorgung und externe Hardware erstmals möglich, dass Chipkarten und Standard-PCs inklusive Standardsoftware zusammenarbeiten. Das Einsatzspektrum ist sehr groß, es geht vom Zugriffsschutz auf bestimmte PC-Funktionen über Kopierschutz für Software bis zur Sicherung und Unterschrift von E-Mails bei der Datenübertragung.

Um den einfachen Datenaustausch zwischen Chipkarten und PCs zu ermöglichen, gibt es auch Disketten-Terminals. Dazu ist in einer 3½-Zoll-Diskette eine sehr dünne Kartenkontaktiereinheit, eine Anschaltelektronik, eine Batterie und eine Spule zur Datenübertragung zum Schreib-/Lesekopf des Diskettenlaufwerks integriert. In diesem 3,3 mm dicken Disketten-Terminal ist ausreichend Platz, um eine Chipkarte einzustecken. Auf dem PC ist dann nur noch ein entsprechender Softwaretreiber für den Datenaustausch notwendig. Dies ist eine Lösung, um auf unkomplizierte und kostengünstige Weise Chipkarten in bestehende Systeme zu integrieren, wobei diese Lösung in der Praxis keine Relevanz erreichen konnte.



Bild 11.5 Chipkarten-Terminal in Form eines USB-Steckers für Chipkarten im ID-000 (Plug-In) Format.

Zwischen den ersten Chipkarten in der Zwei-Chip-Technik und heutigen Varianten, die mit einem sehr leistungsfähigen Mikrocontroller ausgestattet sind, liegen viele Jahre Entwicklungstätigkeit. Eine ähnliche technische Evolution haben parallel dazu die Terminals erfahren. Anfänglich waren sie oft aus Unkenntnis mechanisch und elektrisch sehr primitiv aufgebaut, was zur Folge hatte, dass die Mikrocontroller in den Chipkarten oft beschädigt wurden und vorzeitig ausfielen. Mittlerweile sind diese „Kinderkrankheiten“ bei den meisten Terminal-Herstellern jedoch überwunden, und man hat jene Phase der Entwicklung erreicht, in der weniger die überall gleichen technischen Eigenschaften, die bei allen Herstellern ähnlich sind, die Kaufentscheidung beeinflussen als das äußere Design des Gerätes.

Funktional besteht ein Chipkarten-Terminal aus zwei Komponenten – der Kartenkontaktiereinheit und dem Terminalcomputer. Der Kartenleser, in den die Chipkarte gesteckt und dann elektrisch kontaktiert wird, hat im Wesentlichen nur mechanische Funktion. Um die Kontaktiereinheit elektrisch anzusteuern, die Benutzerschnittstelle zu verwalten und die Verbindung zu übergeordneten Systemen herzustellen, benötigt man den Terminalcomputer. Dies kann im einfachsten Fall ein einzelner Mikrocontroller sein oder – bei technisch aufwendigeren Lösungen – ein Einplatinencomputer.

11.1 Mechanische Eigenschaften

Wird eine Chipkarte in ein Terminal gesteckt, geschieht mechanisch gesehen zweierlei: Die Kontaktflächen der Chipkarte müssen galvanisch mit dem Terminalcomputer verbunden werden, wozu die Kontaktierungseinheit dient. Auch muss das Stecken der Karte durch das Terminal erkennbar sein, was beispielsweise ein Mikroschalter oder eine Lichtschranke übernehmen kann. Die Lichtschranke hat den Nachteil, dass sie durch Verschmutzung oder bei transparenten Kartenmaterialien eine gesteckte Karte nicht mehr sicher erkennt. Ein auf mechanischer Basis arbeitender Schalter funktioniert hier nach wie vor am zuverlässigsten.

Im Bereich der Kontakte und Kontaktierungseinheiten bestehen bei den Terminals die größten Unterschiede. Für die Form der Kontakte sowie die maximal einzuleitenden Kräfte sind in der GSM 11.11 Spezifikation Grenzwerte angegeben, die die Grundlage für fast alle Terminals darstellen. Laut dieser Spezifikation sollen die Kontaktelemente des Terminals abgerundet sein und nicht spitz verlaufen. Der Radius dieser Rundung muss mindestens 0,8 mm betragen. Damit können Kratzer auf den Kontaktflächen der Chipkarte weitgehend vermieden werden. Zudem ist die Einleitung der Kraft in die Kontaktflächen durch vorne abgerundete Kontaktelemente wesentlich günstiger als bei spitz gestalteten.

Die maximale Andruckkraft eines einzelnen Kontaktelements darf nach GSM unter keinen Umständen größer als 0,5 N sein (nach EMV: 0,6 N). Dadurch soll der Chip aus einem Siliziumkristall, der unter den Kontaktflächen liegt, geschützt werden, da er bei größeren Kräften brechen könnte.

Die Position der Kontakte auf der Chipkarte ist zwar international durch die ISO genormt und sollte dadurch weltweit eigentlich gleich sein, doch gibt es in Frankreich eine nationale AFNOR-Norm, die eine Chipposition näher an der oberen Kartenkante als ISO vorsieht.

Aus diesem Grund wurden Terminals konstruiert, die mit zwei Kontaktköpfen ausgerüstet sind. Dadurch ist es möglich, sowohl die ISO- als auch die AFNOR-Position der Kontakte zu unterstützen. Diese technisch aufwendige Lösung ist für Systeme von Interesse, bei denen parallel Chipkarten mit ISO- und AFNOR-Kontaktposition benutzt werden. Dies betrifft nur eine Übergangszeit, da die AFNOR-Position nach ISO eigentlich nicht mehr verwendet werden soll. So sind etwa bei einigen französischen Bankanwendungen Terminals mit doppelten Kontaktköpfen im Einsatz. Damit können dann für die Zeit der Umstellung sowohl die alte AFNOR-Position als auch die Kontakte von neueren Chipkarten in ISO-Position angefahren werden.

Vor allem beim Einsatz von tragbaren oder in Fahrzeugen eingebauten Terminals kommt es manchmal zu Schwierigkeiten mit der Kontaktierung zwischen Terminal und Chipkarte. Diese Art von Terminals ist gerade in Fahrzeugen oft hohen Beschleunigungen ausgesetzt, wodurch es zu einem kurzzeitigen Abheben der Kontakte kommen kann. Man stelle sich nur die Fahrt über Kopfsteinpflaster mit einer Geschwindigkeit vor, bei der die federbelasteten Kontaktelemente in Resonanzschwingung geraten. Ist die Chipkarte dabei elektrisch aktiviert, ist schlichtweg nicht mehr vorhersagbar, was passiert.

Sollten im Extremfall alle Kontakte gleichzeitig abheben und wieder aufsetzen, wird die Chipkarte eine Einschaltsequenz ausführen und dann einen ATR aussenden. Allerdings ist

die elektrische Anschaltsequenz in diesem Fall sicherlich nicht nach ISO, was bei häufiger Wiederholung zum Ausfall der Chipkarte führen kann. Unabhängig davon sind natürlich alle erreichten Zustände der laufenden Sitzung in der Chipkarte durch den kurzzeitigen Spannungsausfall verloren. Dies bedeutet, dass je nach Anwendung die PIN neu eingegeben oder die Kommunikationsteilnehmer nochmals authentisiert werden müssen.

Beim Abheben eines einzelnen Kontaktes hängt die Auswirkung stark davon ab, um welchen Kontakt es sich handelt. Im Fall des I/O-Kontaktes ist nur die Kommunikationsverbindung zeitweise gestört. Mit den üblichen Mechanismen zur Fehlerbehandlung kann diese Störung überbrückt werden. Hebt ein anderer Kontakt ab, kommt es zu einem Reset der Chipkarte, was zu einem kompletten Neuaufbau der Verbindung führt.

Um ein Abheben der Kontakte durch Beschleunigungskräfte zu verhindern, kann die Anpresskraft verstärkt werden, wobei aber 0,5 N pro Kontaktfläche die obere Grenze sind. Technisch gesehen gibt es keine einfache und zufrieden stellende Lösung für dieses Problem, doch durch die sinnvolle Anordnung des Terminals kann die Wahrscheinlichkeit eines Abhebens der Kontakte minimiert werden. Eine Möglichkeit besteht z. B. darin, die Kontakte im rechten Winkel zur Hauptbeschleunigungsrichtung zu positionieren.

Allerdings muss die Terminal-Software selbstverständlich fähig sein, eine Verbindung wieder aufzubauen, wenn die Kontakte kurz abgehoben haben. Die Millionen GSM-Endgeräte sind ein Beweis dafür, dass der Einsatz von Chipkarten in tragbaren Geräten problemlos möglich ist.

Die Standzeiten für die Kontakte und den technischen Aufbau bei Terminals sind höchst unterschiedlich. Sie werden auch sehr stark durch die Umweltbedingungen – wie Temperatur, Luftfeuchtigkeit und Ähnliches – beeinträchtigt. Eine MTBF (*mean time between failure*) von 150000 Steckzyklen wird aber als ein durchaus üblicher Wert für Terminals angenommen.

Kontaktiereinheit mit Schleifkontakten

Die technisch einfachsten und somit auch preisgünstigsten Terminals verfügen lediglich über Schleifkontakte in der Form von Blatt- oder Tellerfedern. Weitere mechanische Kontaktelemente sind bei diesen einfachen Lesern nicht vorhanden. Eine Kontaktiereinheit auf Federbasis führt aber dazu, dass beim Stecken und Herausziehen der Chipkarte die Kontaktelemente über einen Teil der Karte und der Kontaktflächen schleifen und dort Kratzspuren hinterlassen. Diese sind nicht nur aus ästhetischen, sondern auch aus technischen Gründen unerwünscht.

Durch das Verkratzen der vergoldeten Kontaktflächen der Chipkarte wird nach und nach der Oxidationsschutz, der aus einer vergoldeten Oberfläche besteht, abgetragen. Die nun freigelegte Metallschicht oxidiert. Darunter leidet die elektrische Verbindung, und der Benutzer muss seine Chipkarte unter Umständen mehrmals stecken und wieder ziehen, bis die Oxidschicht abgetragen und dann wieder eine zufrieden stellende elektrische Verbindung hergestellt ist.

Mechanisch angetriebene Kontaktiereinheit

Die nächste Kategorie der Terminals hat keine fest positionierten Schleifkontakte, sondern verfügt über eine Mechanik, die beim Stecken der Chipkarte die Kontaktiereinheit auf die Kontaktflächen aufsetzt. Durch die Kraft beim Einschieben der Chipkarte wird über einen

Hebelmechanismus die Kontaktiereinheit im rechten Winkel auf die Kontaktflächen aufgesetzt.

Eine optimal gestaltete Mechanik ist dann gegeben, wenn beim Aufsetzen der Kontakte eine sehr kleine Relativbewegung parallel zu den Kontaktflächen der Chipkarte besteht. Damit kommt die elektrische Verbindung sicher zustande. Hierdurch können auch leichte Verschmutzungen auf den Kontaktoberflächen weggerieben werden. Die Kontaktelemente selbst sind zusätzlich noch einzeln gefedert, um so eine definierte Anpresskraft für jede einzelne Kontaktfläche sicherzustellen.

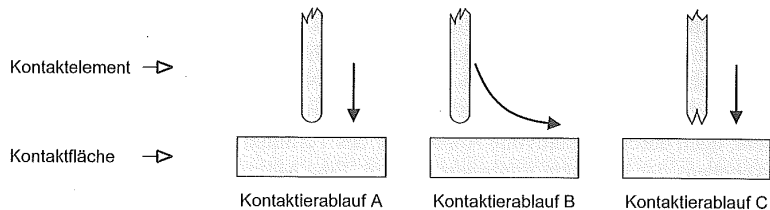


Bild 11.6 Verfahren zur elektrischen Kontaktierung von Chipkarten. Kontaktierablauf A mit gerundeten Kontaktelementen ist ungünstig, da Verschmutzungen der Kontaktfläche eine sichere elektrische Verbindung beeinträchtigen. Ablauf B und C stellen für die jeweilige Form der Spitze des Kontaktelements gut geeignete Lösungen dar. Die scharfkantigen Kontakte von C graben sich leicht in die Oberfläche der Kontaktfläche ein, was unter dem Mikroskop in Form kleiner Kerben sichtbar ist.

Elektrisch angetriebene Kontaktiereinheit

Die technisch aufwendigste und mechanisch beste Lösung sind Terminals mit elektrisch angetriebener Kontaktiereinheit. Parallel geführte Kontaktelemente werden dabei von einem Motor oder Elektromagneten angetrieben während die Chipkarte senkrecht von oben, mit geringer seitlicher Relativbewegung, kontaktiert wird. Bedingt durch diesen großen Aufwand an elektromechanischen Bauteilen, fallen die Abmessungen des Terminals relativ groß aus.

Für professionelle Lösungen, bei denen mehrere Millionen Kontaktierungszyklen wartungsfrei erreicht werden müssen, sind diese Geräte jedoch bestens geeignet. Typische Einsatzgebiete sind deshalb Geldausgabeautomaten oder Personalisierungsmaschinen in der Chipkarten-Fertigung.

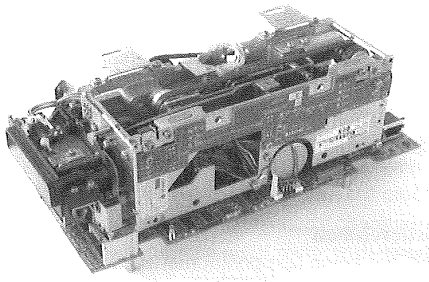


Bild 11.7 Typischer Einzugsleser für Geldausgabeautomaten mit Shutter und Magnetstreifenleser.

Kartenauswurf

Gesteckt wird eine Chipkarte meistens manuell, also ohne weitere Unterstützung durch das Terminal. Lediglich in Geldausgabeautomaten sind Einzugleser vorhanden, die dann die Chipkarte mit einem Fördermechanismus an den Kontaktkopf innerhalb des Automaten heranführen. Bei einem normalen Terminal ist das aber nicht der Fall. Dort gibt es lediglich beim Auswurf der Chipkarte Unterschiede. Einfache Leser haben keinen automatischen Auswurf, die Chipkarte muss also manuell aus dem Leser entnommen werden. Für diese Entnahme gibt es wiederum zwei Prinzipien: Push-Push und Push-Pull. Bei einer Push-Push-Kontaktiereinheit wird die Karte wie üblich von Hand gesteckt, und zur Entnahme muss dann ein weiteres Mal auf die Karte gedrückt und dann gezogen werden. Dies ist ergonomisch unvorteilhaft, da es dem natürlichen Bewegungsablauf widerspricht. Entsprechend oft ziehen viele Menschen die Chipkarte einfach mit Gewalt aus dem Terminal, was dazu führt, dass die Kontakte durch die Mechanik nicht abgehoben werden und über die Kontaktflächen und den Kartenkörper schleifen. Kontaktiereinheiten in der Push-Pull-Ausführung entsprechen dem gewohnten Bewegungsablauf eher, da bei ihnen die Chipkarte beim Stecken in den Leser geschoben und beim Entnehmen einfach herausgezogen wird.

Bei Terminals mit automatischem Auswurf wird beim Stecken der Karte über eine Mechanik eine Feder gespannt, die der Terminalcomputer über einen Magneten freigeben kann. Dies führt zum teilweisen Herausschieben, nicht zum Auswurf, der Chipkarte, so dass der Benutzer sie ergreifen und ganz herausziehen kann.

Die Auswurfleser haben gegenüber den anderen Bauarten einen großen Vorteil. Dem Benutzer wird durch den Auswurf der Chipkarte auf sehr deutliche Weise das Ende der Sitzung signalisiert; gleichzeitig wird er auch daran erinnert, dass er seine Karte im Terminal nicht vergessen soll, was man oft noch durch einen Piepser unterstützt. Dieses aus der Praxis stammende Argument ist der wesentliche Vorteil der Auswurfleser.

Gerade Geldausgabeautomaten haben üblicherweise die Fähigkeit, Chipkarten bei Bedarf einzubehalten. Da diese Geräte standardmäßig mit einem Einzugleser ausgestattet sind, der die Chipkarte erst zum Kontaktkopf befördert, ist es natürlich technisch ohne weiteres machbar, die Karte bei Bedarf nicht mehr zum Ausgabeschlitz zu befördern, sondern zu einem speziellen Auffangbehälter im Automaten. Technisch gesehen bereitet also das Einziehen einer Chipkarte keine großen Probleme, sofern das Terminal nur groß genug für zusätzliche Mechanik und Auffangbehälter ist. Unter Umständen kann der Einzug aber zu rechtlichen Problemen führen, wenn der Kartenbenutzer der Eigentümer der Chipkarte ist.

Ziehbarkeit

Die Zuverlässigkeit eines auf Chipkarten aufbauenden Systems kann sehr darunter leiden, wenn die Benutzer die Möglichkeit haben, ihre Karten zu jedem beliebigen Zeitpunkt während einer Sitzung aus dem Terminal zu ziehen. Dies führt zum einen dazu, dass die Chipkarte von der Stromversorgung ohne die vorgeschriebene Abschaltsequenz getrennt wird. Zum anderen unterbricht man unter Umständen Schreib- oder Löschvorgänge im EEPROM, wobei der Inhalt der Datei anschließend undefiniert ist, was zum Totalausfall der Chipkarte führen kann. Dies begründet, warum es vorteilhaft ist, Auswurfleser zu benutzen, bei denen das manuelle Ziehen der Karte durch eine entsprechende Gestaltung des Terminals nicht möglich ist.

Für den Fall des Stromausfalls ist ein verdeckter mechanischer Notauswurf vorhanden, um die Chipkarte dem Terminal entnehmen zu können. Normalerweise kann jedoch das Terminal bestimmen, wann die Karte dem Benutzer zurückgegeben wird, ohne dass dieser eine Möglichkeit hat, störend in die ablaufenden Prozesse einzugreifen.

11.2 Elektrische Eigenschaften

Mit Ausnahme der Kontaktiereinheit besteht ein Terminal vor allem aus Elektronikkomponenten. Man benötigt diese für die Schnittstellen zum Benutzer, zum Hintergrundsystem und auch zur elektrischen Ansteuerung der Kontakte. Es sind einerseits die elektromechanischen Teile des Lesers und andererseits die Chipkarte mit elektrischen Signalen zu versorgen. Die einzige Information, die direkt von der Kontaktiereinheit kommt, sagt aus, ob die Chipkarte gesteckt worden ist oder nicht. Das einzige Signal zur Kontaktiereinheit hingegen ist die Betätigung des elektrischen Auswurfs, falls ein solcher überhaupt vorhanden ist.

Die Schnittstelle zur Chipkarte besteht aus den fünf Kontakten für Masse, Spannungsversorgung, Takt, Reset und die Datenübertragung. Es ist sehr wichtig und von großer Bedeutung für die Lebensdauer der Chipkarte, dass nach der Herstellung der elektrischen Verbindung durch die Kontaktelemente die Anschaltsequenz nach ISO/IEC 7816-3 genau eingehalten wird. Trifft dies nicht zu, kann es durch elektrische Überlastungen im Halbleiter des Chips zu erhöhten Ausfallraten kommen. Analog dazu ist aber auch die Einhaltung der Abschaltsequenz wichtig, da es sonst zu den gleichen Problemen im Chip kommt wie bei einer nicht eingehaltenen Anschaltsequenz.

In diesem Zusammenhang ist gerade bei einfachen Lesern, bei denen es möglich ist, die Chipkarte von Hand zu ziehen, ein wichtiges Detail von Bedeutung: Wenn die Kontaktiereinheit ein Herausziehen der Chipkarte feststellt, muss die Elektronik des Terminals sofort eine Abschaltsequenz ausführen. Nur so kann verhindert werden, dass eventuell noch spannungsführende Schleifkontakte über die Kontaktflächen der Chipkarte gleiten. Dies hat dann mit einer genormten Abschaltsequenz nicht mehr viel zu tun. Doch die Auswirkungen dieses unzulässigen Ziehens der Karte können noch viel größer sein, da es bei verschlissenen oder leicht verbogenen Kontaktelementen auch zu Kurzschlüssen zwischen zwei Leitungen kommen kann. Die dabei auftretende schwache Funkenbildung durch Entladung von Kondensatoren schadet den Kontaktelementen sowie den Kontaktflächen der Chipkarte.

In Hinblick auf die elektrische Beschaltung haben mittlerweile fast alle Terminal-Hersteller herausgefunden, dass Kurzschlussfestigkeit ein absolutes Muss ist. Wird dies missachtet, kann eine einzige Chipkarte mit kurzgeschlossenen Kontakten zum elektrischen Tod vieler Terminals führen. Im Übrigen treten kurzgeschlossene Karten immer wieder auf, zum Teil aus Vandalismus oder wegen technischer Defekte.

Die Kurzschlussfestigkeit sollte dabei so weit reichen, dass jeder Kontakt gegen jeden anderen und auch alle zueinander ohne Auswirkungen miteinander verbunden werden können. Ideal ist auch eine vollständige galvanische Trennung zwischen der elektrischen Ansteuerung der Chipkarte und der übrigen Elektronik im Terminal. Dies ist im Übrigen Standard bei den öffentlichen Kartentelefonen in Deutschland, denn damit lassen sich selbst Auswirkungen von extern angelegten Spannungen auf das Terminal weitgehend verhindern.

Die Spannung zum Schreiben und Löschen der Speicherseiten im EEPROM erzeugt der Mikrocontroller durch eine auf dem Chip befindliche Ladungspumpe. Dabei benötigt die Chipkarte, teilweise im Bereich von Nanosekunden, Ströme von bis zu 100 mA. Der gleiche Effekt, wenn auch nicht so stark, kann durch Umschaltvorgänge von Transistoren in den CMOS-Schaltkreisen auftreten. Durch diese kurzen Spikes sind selbst sehr schnelle Regelkreise in Stromversorgungen überfordert. Die Folge davon ist, dass die Versorgungsspannung der Chipkarte aufgrund des hohen Stromverbrauches einbricht und daraufhin der Schreib- oder Löschvorgang im EEPROM fehlschlägt. Im Extremfall kann der Spannungseinbruch so groß sein, dass der Prozessor außerhalb seines stabilen Funktionsbereiches gerät und ein Systemzusammenbruch eintritt.

Abhilfe kann geschaffen werden, indem ein Kondensator so nahe wie möglich an den Kontakten zur Chipkarte platziert wird. Gut geeignet sind Keramikkondensatoren mit einer Kapazität von ungefähr 100 nF, da diese sehr schnell ihre Ladung abgeben können. Die Leitungen zur Chipkarte müssen so kurz wie möglich sein, damit der erhöhte Strombedarf ohne großen Leitungswiderstand und Induktivität in der benötigten Zeit gedeckt werden kann. Bei kurzzeitig erhöhtem Stromverbrauch kann der Strom aus dem Kondensator so lange bezogen werden, bis die Spannungsregelung dies wieder ausgleicht. Damit vermeidet man auf einfache und kostengünstige Weise Probleme mit der Stromversorgung.

Gerade im Bereich des elektronischen Zahlungsverkehrs ist es aus Gründen der Nachvollziehbarkeit und des Verbraucherschutzes mittlerweile Stand der Technik, dass die entsprechenden Terminals eine eingebaute Echtzeituhr besitzen. Nach der EMV-Spezifikation für Kreditkarten-Terminals darf die Uhr keine größere Zeitabweichung als 1 Minute pro Monat haben. Dies stellt technisch keinen Aufwand dar, da entsprechend genaue Uhrenbausteine sogar als Einchip-Lösung erhältlich sind. Zusätzlich kann bei jeder Online-Verbindung mit den übergeordneten Systemen ein Abgleich der Uhr vorgenommen werden. Funkuhrenempfänger in Terminals haben bisher keine praktische Bedeutung erlangt, da ein Empfang der entsprechenden Signale zu stark vom Abschirmverhalten des Einsatzortes abhängt. So können beispielsweise die üblichen Zeitsignalsender im Inneren von Stahlbetonbauten meist gar nicht empfangen werden.

11.3 Sicherheitstechnik

Die in Terminals vorhandenen Sicherheitsmechanismen können sehr vielfältig sein. Die Palette reicht von mechanisch gesicherten Gehäusen bis zu Sicherheitsmodulen und Sensoren für diverse Kartenmerkmale. Bei reinen Online-Terminals, die also nur die elektrische Konvertierung der Signale eines Hintergrundrechners von und zur Chipkarte durchführen, besteht normalerweise keine Notwendigkeit für zusätzlich eingebaute Sicherheitstechnik. Die gesamte Sicherheit liegt dabei in dem das Terminal ansteuernden Rechner.

Sobald jedoch am Terminal Daten eingegeben werden müssen, oder dieses autark von einem übergeordneten System operieren kann, ist es notwendig, die entsprechenden Mechanismen für die zusätzliche Systemsicherheit einzubauen. Die Möglichkeiten dazu sind nahezu unbeschränkt, hängen aber sehr stark von der jeweiligen Chipkarte und deren Sicherheitsmerkmalen ab.

Bei typischen Chipkarten, deren Kartenkörper sehr einfach ausgeführt und nur Träger für den Mikrocontroller ist, sind üblicherweise keine Sicherheitsmerkmale auf dem Kartenkörper vorhanden. Damit entfällt auch die Prüfung dieser Merkmale durch das Terminal. Im Gegensatz dazu sind Chipkarten für den Zahlungsverkehr meist Hybridkarten, d. h. sie verfügen sowohl über einen Magnetstreifen als auch über einen zusätzlichen Chip, um dadurch kompatibel zu älteren Systemen zu sein. Hybridkarten besitzen aber auch die üblichen Kartenmerkmale, damit das Terminal unabhängig vom Chip die Echtheit überprüfen kann. Die entsprechenden Sensoren müssen dann auch in den Terminals vorhanden sein.

Terminals, die vollständig oder auch nur zeitweise im Offline-Betrieb arbeiten, müssen zwangsläufig über Hauptschlüssel für die verwendeten kryptografischen Algorithmen verfügen. Nur mit ihnen lassen sich dann jeweils die kartenindividuellen Schlüssel ableiten. Diese Hauptschlüssel sind sehr sicherheitssensibel, denn auf ihnen ist die Sicherheit des gesamten Systems aufgebaut. Um die Sicherheit und die Geheimhaltung der Hauptschlüssel in allen Fällen zu gewährleisten, sind sie nicht in der normalen Elektronikeinheit des Terminals gespeichert, sondern in einem mechanisch und elektrisch besonders abgesicherten Modul innerhalb des Terminals.

Dieses Sicherheitsmodul kann z. B. ein in Epoxidharz vergossener Einplatinencomputer sein, der nur über eine Schnittstelle mit dem eigentlichen Terminalcomputer Daten austauschen kann. Dabei dürfen aber die geheimen Hauptschlüssel das Sicherheitsmodul nie verlassen, sondern werden nur intern für Berechnungen genutzt. Im typischen Anwendungsfall erhält das Sicherheitsmodul von der Chipkarte über den Terminalcomputer eine individuelle Karten- oder Chipnummer, die es zur Ableitung eines kartenindividuellen Schlüssels benutzt. Dieser dient dann innerhalb des Sicherheitsmoduls wiederum zur Berechnung von Signaturen oder zur Authentisierung.

Moderne Ausführungen dieser meist nur Streichholzschachtel-großen Module besitzen eine umfangreiche Sensorik zur Detektion von Angriffen und sind auch elektrisch weitgehend autark, um Angriffe, selbst bei abgeklemmter Spannungsversorgung, aktiv abwehren zu können. Erkennen und Abwehr eines Angriffs führt üblicherweise zum Löschen aller Schlüssel, sodass ein Angreifer nur mehr eine in Harz vergossene Elektronikplatine im Metallgehäuse ohne irgendwelche Dateninhalte zur Analyse hat.

Aufgrund der hohen Kosten eines guten Sicherheitsmoduls geht man seit einigen Jahren dazu über, stattdessen Chipkarten zu verwenden. Dies führt zwar zu einigen Einschränkungen hinsichtlich Speicherplatz, Sensorik und Autarkie, doch reicht das Sicherheitsniveau im Allgemeinen selbst für Anwendungen im elektronischen Zahlungsverkehr aus. Um die physikalische Größe zu begrenzen, benutzt man Chipkarten im ID-000 Format („Plug-In“).

Da Sicherheitsmodule im Chipkartenformat nicht fest in Terminals eingebaut sind, sondern ausgewechselt werden können, eignen sie sich auch hervorragend für eine Erweiterung der Terminalhardware. An einem Beispiel sei dies verdeutlicht: Die statische einseitige RSA-Authentisierung wird in den nächsten Jahren sehr stark an Bedeutung gewinnen, da sie unter anderem auch in der internationalen EMV-Spezifikation für Kreditkarten mit Chip gefordert wird. Da eine RSA-Authentisierung so rechenintensiv ist, dass sie von den normalen Prozessoren in Terminals nicht mehr in akzeptabler Zeit ausgeführt werden kann, hat man bei fest eingebauten Sicherheitsmodulen ein Problem. Bei Terminals hingegen, die Chipkarten im Plug-In-Format als Sicherheitsmodule benutzen, kann man diese

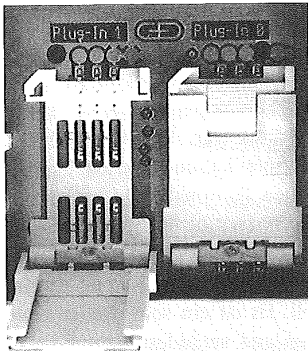


Bild 11.8 Beispiel für zwei in einem Chipkarten-Terminal nebeneinander angebrachte Kontaktierungseinheiten für Sicherheitsmodule im Plug-In-Format.

einfach austauschen. Man verwendet die etwas teuren Chipkarten mit einer zusätzlichen numerischen Recheneinheit, und nach einer Softwareänderung im Terminal ist Ersteres in der Lage, eine RSA-Berechnung in hoher Geschwindigkeit auszuführen.

In Zukunft werden von den verschiedensten Kartenherausgebern Debit- und Kreditkarten mit Chip auf den Markt gebracht. Alle diese Karten benutzen unterschiedliche Schlüssel, Schlüsselableitungs- und Authentisierungsverfahren. Zudem werden wohl nicht alle Kartenherausgeber bereit sein, diese geheimen Daten und Verfahren Herstellern von Sicherheitsmodulen zu offenbaren. Man wird aller Wahrscheinlichkeit nach auch so verfahren, dass einer oder auch mehrere Kartenherausgeber zusammen eine so genannte Terminalkarte herausgeben, die alle für ihr System sicherheitsrelevanten Prozesse beherrscht und diese innerhalb des Terminal ausführt. Die Terminalkarte wird mit einem der beiden genormten Übertragungsprotokolle $T = 0$ oder $T = 1$ angesprochen und verhält sich im Großen und Ganzen wie eine ganz normale Chipkarte. Der Unterschied dabei ist nur, dass sie über Funktionen für die geheimen Hauptschlüssel, die Ableitungsverfahren und das Sammeln von sicherheitsrelevanten Daten (z. B.: Saldo des Händlers) verfügt. Das Terminal übernimmt dann nur mehr die Benutzerführung und das Up- und Downloading von Daten mit dem Hintergrundsystem. Alle sicherheitstechnischen Belange führt die Terminalkarte aus. Damit besteht aber für die Terminals die Notwendigkeit, nicht nur mit einer Terminalkarte zusammenzuarbeiten, sondern mit mehreren. Je nach Kartenherausgeber und gewählter Funktion wird eine bestimmte Terminalkarte automatisch ausgewählt. Der Forderung nach mehreren unabhängigen Terminalkarten wird in den neueren Terminals Rechnung getragen. Sie haben zum Teil bis zu vier Kontaktiereinheiten für Karten im Plug-In-Format. Damit können sie Terminalkarten von mehreren unterschiedlichen Kartenherausgebern ohne Interferenzen zueinander parallel bedienen.

Neben der Absicherung der Terminals durch ein mechanisch stabiles und nur mit Spezialwerkzeug zu öffnendes Gehäuse und ein darin untergebrachtes Sicherheitsmodul existiert oft zusätzlich eine mechanische Sicherung gegenüber unbefugtes Abhören der Datenübertragung zur Chipkarte. Dies ist in der technischen Realisierung eine Art Schlagschere, die nach dem Stecken und Kontaktieren der Chipkarte alle eventuell von der Karte aus der Kontaktiereinheit führenden Drähte oder Kabel abschneidet. Zweck dieser Vorrichtung ist

es, Mithören und Manipulationen von Nachrichten zwischen Chipkarte und Terminal zu verhindern. Diese auch als Shutter bezeichnete Vorrichtung kann elektrisch oder durch Stecken der Chipkarte selber angetrieben sein. Ist ein Abschneiden aufgrund Materialdicke oder -beschaffenheit nicht möglich, kann auch der Schlitz der Kontaktiereinheit nicht ganz geschlossen werden. Dies stellt die Terminalelektronik fest, und die Chipkarte wird elektrisch nicht angeschaltet; es kommt also keine Kommunikation zustande.

Grundsätzlich muss die Kommunikation zwischen Terminal und Chipkarte so aufgebaut sein, dass durch Abhören oder Manipulation das System nicht in seiner Sicherheit beeinträchtigt werden kann. Shutter wären demnach eigentlich nicht notwendig. Allerdings lässt sich die Sicherheit auf diese Weise durchaus etwas verbessern, wenn es einem fiktiven Angreifer nicht allzu leicht gemacht werden soll. Es ist ein sehr großer Unterschied, ob man den Datenaustausch ohne Aufwand mithören und verändern kann oder ob man dazu vorher einige Hürden überwinden muss. Allerdings vergrößern und verteuern Shutter das Terminal, und es gibt nur sehr wenige Ausführungen, die nach einigen zehntausend Steckzyklen noch exakt schließen. Deshalb sollte man sich beim Systemdesign nicht vollständig auf diese mechanische Sicherung verlassen.

11.4 Anbindung von Terminals an übergeordnete Systeme

Um Chipkarten im Umfeld von PCs einsetzen zu können, bedarf es eines daran angeschlossenen Terminals und der Unterstützung durch die Software des PCs. Problematisch ist hier natürlich, dass in der Vergangenheit für jedes Terminal eigene Softwaretreiber auf dem PC installiert werden mussten. Diese hatten zudem ihre eigenen Softwareschnittstellen, sodass es praktisch nicht möglich war, eine terminalunabhängige Anwendungssoftware zu erstellen. Mitte der 90er begannen in verschiedenen Ländern und ausgehend von unterschiedlichsten Institutionen Spezifikationsarbeiten für Schnittstellen zur terminalunabhängigen Einbindung von Chipkarten in PC-Programme. Weltweit haben sich die beiden Industriestandards PC/SC SC (*personal computer/smart card*) und OCF (*open card framework*) etabliert. In Deutschland wie in einigen anderen Ländern gibt es seit langem die MKT-Spezifikation (Multifunktionales Kartenterminal – MKT), die im deutschsprachigen Raum eine überraschend hohe Verbreitung kennt. Alle drei Spezifikationen sind im Folgenden im Überblick dargestellt und können ansonsten auch via Internet kostenlos bezogen werden.

11.4.1 PC/SC

Ab Mai 1996 begann man, eine internationale Spezifikation für die Verbindung zwischen Karte und PC zu erstellen. Die Firmen Bull, Hewlett-Packard, Microsoft, Schlumberger, Siemens Nixdorf, Gemplus, IBM, Sun, Verifone und Toshiba arbeiteten an dieser Spezifikation mit.

Im Dezember 1997 wurde die Version 1.0 einer aus acht Teilen bestehenden „Interoperability Specification for ICCs and Personal Computer Systems“ veröffentlicht. Der Name für die Arbeitsgruppe war PC/SC (*personal computer/smart card*), welcher auch als Kurz-

form für die Spezifikation benutzt wird. Sie ist via Internet über den WWW-Server der Spezifikationsgruppe erhältlich [PC/SC].

Die PC/SC-Spezifikation ist zumindest ansatzweise plattformunabhängig, da sie auf allen von Windows unterstützten Rechnern funktioniert, welche den Großteil aller PCs ausmachen. Sie ermöglicht die Einbindung von Chipkarten in beliebige Anwendungen weitgehend unabhängig von einer Programmiersprache, da verbreitete Sprachen wie C, C++, Java und Basic von PC/SC unterstützt werden. Die Voraussetzung ist lediglich, dass für das jeweils benutzte Terminal ein passender Treiber vorhanden und die verwendete Chipkarte PC/SC-kompatibel ist. Diese geforderte Kompatibilität ist jedoch relativ unkritisch, da der Rahmen verhältnismäßig weit gehalten wurde.

Tabelle 11.2 Die acht Teile der PC/SC-Spezifikation im Überblick

PC/SC Spezifikation	Inhalt
Teil 1: Introduction and Architecture Overview	Dieser Teil ist die Grundlage für alle folgenden Spezifikations- teile. Er enthält die zum Thema relevanten Normen, einen Über- blick über die Systemarchitektur und die benutzten Hard- und Softwarekomponenten sowie Definitionen und Abkürzungen.
Teil 2: Interface Requirements for Compatible IC Cards and Readers	Festlegung der physikalischen Eigenschaften der kontaktbehafteten Chipkarte. Fixierung der grundlegenden elektrischen Eigen- schaften wie Spannungsversorgung und Resetverhalten. Ebenso sind die Datenelemente, der Aufbau und die möglichen Abläufe von ATR und PTS definiert. Überblicksartig werden die Grund- lagen der Datenübertragung auf physikalischer Ebene und die beiden Übertragungsprotokolle T = 0 und T = 1 beschrieben.
Teil 3: Requirements for PC- Connected Interface Devices	Aufstellung der Anforderungen an das Terminal sowie der unter- stützten Terminaleigenschaften (Display, Tastatur u. Ä.).
Teil 4: IFD Design Considerations and Reference Design Information	Informationen zum Design von Terminals in Hinblick auf PS/2-Tastaturschnittstellen und USB-Schnittstellen.
Teil 5: ICC Resource Manager Definition	Detaillierte softwaretechnische Beschreibung des ICC Ressource Managers inklusive der dazugehörigen Klassen.
Teil 6: ICC Service Provider Interface Definition	Detaillierte softwaretechnische Beschreibung des ICC Service Providers und Crypto Service Providers inklusive der dazugehö- rigen Klassen.
Teil 7: Application Domain and Developer Design Considerations	Beschreibung der Benutzung der PC/SC-Spezifikation von An- wendungen aus.
Teil 8: Recommendations for ICC Security and Privacy Devices	Zusammenstellung und Festlegung von empfohlenen Funktionen und Mechanismen, welche eine PC/SC Chipkarte unterstützen sollte. Dies schließt ein: das Dateisystem (MF, DF, EF), die da- zugehörigen Dateizugriffsbedingungen, notwendige System- dateien in der Chipkarte (Schlüssel, PIN u. Ä.), Kommandos, Returncodes und kryptografische Algorithmen.

Ausgehend von den definierten Hard- und Softwarekomponenten lässt sich am unkompliziertesten die Übersicht über die PC/SC-Spezifikation gewinnen. Es sind die folgenden sieben Komponenten in ihren Funktionen und Schnittstellen zueinander beschrieben: ICC-Aware Application, ICC Service Provider, Crypto Service Provider, ICC Resource Manager, IFD Handler, IFD und ICC. Im Folgenden wird top-down jede dieser Komponenten anhand ihrer Aufgaben und Funktionen kurz erläutert.

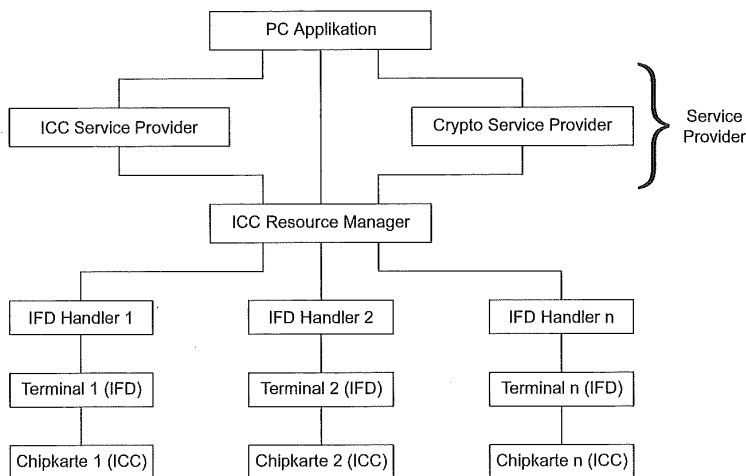


Bild 11.9 Überblick über die Softwarearchitektur der PC/SC-Spezifikation zur Anbindung von Chipkarten an PC-Betriebssysteme.

ICC-Aware Application

Diese Softwarekomponente ist die Anwendung, welche auf einem PC läuft und Funktionen und Daten einer oder mehrerer Chipkarten nutzen möchte. Es wird berücksichtigt, dass diese Anwendung auch unter einem Mehrbenutzer-Betriebssystem mit Multitasking und Multithreading ausgeführt werden kann.

Service Provider

Der Service Provider dient dazu, die einzelnen Funktionen einer Chipkarte, unabhängig vom jeweiligen Betriebssystem der Chipkarte, abzukapseln. So kann beispielsweise über das API (*application programming interface*) des Service Providers eine Datei selektiert werden, ohne dass man das Chipkarten-Kommando oder überhaupt die Codierung dazu kennen muss.

Die Komponente Service Provider ist in einen ICC Service Provider und einen Crypto Service Provider zweigeteilt. Der Grund dafür war, dass die in vielen Ländern bestehenden Exportbeschränkungen für kryptografische Algorithmen umgangen werden können, indem man den exportrechtlich problematischen Crypto Service Provider abtrennt und nicht mitliefert. Die PC/SC-Schnittstelle würde dann bis auf alle Kryptofunktionen nach wie vor benutzbar bleiben.

Der Service Provider muss nicht ein aus einem Stück bestehendes Programmteil sein, sondern kann aus über Netzwerke verbundenen Softwarekomponenten bestehen. Damit könnte zum Beispiel der Crypto Service Provider auf einem kryptografisch sicheren oder sehr leistungsfähigen Rechner weit entfernt von den restlichen PC/SC-Bestandteilen ausgeführt werden.

ICC Resource Manager

Der ICC Resource Manager ist die wichtigste Komponente der PC/SC-Architektur. Er ist der Verwalter von allen Ressourcen, die für die Einbindung von Chipkarten in das Be-

triebssystem notwendig sind. Drei fundamentale Aufgaben hat der ICC Resource Manager zu erfüllen:

Er ist verantwortlich für die Erkennung der angeschlossenen Terminals sowie der Chipkarten. Zudem muss er auch noch das Stecken und Ziehen einer Chipkarte im Terminal bemerken und daraufhin eine diesbezügliche Meldung zur Verfügung stellen.

Als zweite Aufgabe muss er die Belegung von Terminals durch eine oder mehrere Anwendungen verwalten. Dazu kann die Ressource „Terminal“ ausschließlich an eine bestimmte Anwendung vergeben werden. Sollten jedoch mehrere Anwendungen gleichzeitig auf ein einzelnes Terminal zugreifen, muss dieses vom ICC Resource Manager als aufgeteilt (*shared*) gekennzeichnet und verwaltet werden.

Die dritte Funktion ist die Bereitstellung von Transaktionsprimitiven (*transaction primitives*). Dabei werden die für eine bestimmte Funktionalität relevanten Kommandos zu einer Gruppe verbunden. Damit ist sichergestellt, dass sie unmittelbar nacheinander in Folge ausgeführt werden. Sonst könnte es passieren, dass zwei Anwendungen zeitgleich und zueinander unkoordiniert mit ihren jeweiligen Kommandos auf eine Chipkarte zugreifen.

Die dabei zum Tragen kommende Problematik kann man am einfachsten anhand des folgenden Beispiels veranschaulichen: In einer Chipkarte kann zu einer Zeit nur jeweils eine Datei selektiert sein. Versuchen zwei Anwendungen nun gleichzeitig mit Selektion (SELECT FILE) unterschiedlicher Dateien und anschließendem Lesekommando (z. B.: READ BINARY), Daten aus der Chipkarte abzurufen, dann ist völlig undefiniert, welche Datei nun wirklich ausgelesen wird. Dies hängt einzig und allein von der Reihenfolge des Eintreffens der Kommandos bei der Chipkarte ab. Weitaus komplizierter, doch nicht minder verwickelt wird die Situation, wenn komplexe Abläufe wie etwa beim Bezahlen mit einer elektronischen Geldbörse zwischen mehreren Anwendungen und einer Chipkarte durchgeführt werden müssen. Der ICC Resource Manager sorgt dafür, dass es nicht zu Überschneidungen zusammengehöriger Kommandosequenzen kommen kann, sondern die einzelnen Abläufe nacheinander ausgeführt werden.

IFD Handler

Der IFD Handler ist eine Art von Treiber, der für das jeweilige Terminal spezifisch ist. Er hat die Aufgabe, das Terminal an die festgelegten Schnittstellen des PCs anzubinden und die individuellen Eigenschaften des Terminals auf die PC/SC-Schnittstelle abzubilden. Der IFD Handler stellt damit sozusagen einen datentechnischen Kanal vom PC zu einem speziellen Terminal her.

IFD (*interface device*)

Die Komponente IFD der PC/SC Spezifikation ist ein Terminal, das über eine Schnittstelle an den PC angeschlossen ist. Diese kann beliebig sein, das Terminal kann also beispielsweise über RS-232, USB (*universal serial bus*) oder PC-Card (ex PCMCIA) mit dem Computer verbunden sein. Das Terminal muss die ISO/IEC 7816-1, 2, 3 Normen erfüllen, was unter anderem bedeutet, dass beide asynchrone Übertragungsprotokolle T = 0 und T = 1 unterstützt werden müssen. Optional können zusätzlich die synchronen Übertragungsprotokolle (2-Draht, 3-Draht, I²C-Bus) für Speicherkarten nach ISO/IEC 7816-10 implementiert sein.

Neben einem Display werden beim Terminal für die Benutzeridentifizierung auch numerische Tastatur, Fingerabdruckscanner und andere biometrische Identifikationssensoren durch PC/SC unterstützt.

ICC (*integrated chip card*)

Die von der PC/SC-Spezifikation obligatorisch unterstützten Chipkarten sind Prozessorkarten und kompatibel zu den Normen ISO/IEC 7816-1, 2, 3. Falls es das Terminal zulässt, können Speicherkarten nach ISO/IEC 7816-10 eingebunden werden.

11.4.2 OCF

Im Jahr 1997 wurde die Open Card Initiative [OCF] von mehr als zehn Firmen aus dem Umfeld von Chipkarten und PCs gegründet. Der Zweck war, eine Schnittstelle zu Chipkarten auf PCs zu schaffen, die unabhängig vom Betriebssystem des PCs (Windows, Unix, ...) und unabhängig von der jeweiligen Anwendung auf der Chipkarte ist. Das Ergebnis ist OCF (*open card framework*), eine Java-basierte Schnittstelle auf PCs, mit deren Hilfe von PC-Programmen aus auf Chipkarten-Anwendungen zugegriffen werden kann. OCF hat sich mittlerweile zu einem Industriestandard im Java-Umfeld etabliert.

11.4.3 MKT

In Deutschland wurde bereits frühzeitig damit begonnen, eine Spezifikation zur softwaretechnischen Anbindung von Terminals an PCs zu erstellen. Dies ist die seit 1994 von Teletrust Deutschland in mehreren Versionen veröffentlichte MKT-Spezifikation (Multifunktionale Karten Terminal – MKT). Sie ist vor allem auf die Belange des Gesundheitsbereiches ausgerichtet, wird jedoch mittlerweile auch von vielen anderen Terminals innerhalb Deutschlands als Grundlage benutzt.

Die MKT-Spezifikation setzt sich aus 7 Teilen zusammen. Teil 1 beinhaltet das MKT-Basiskonzept, das einen grundlegenden Überblick über die Softwarearchitektur und das MKT-Terminal enthält. In Teil 2, dem so genannten CT-ICC-Interface (*card terminal – integrated chip card*), ist die Schnittstelle für kontaktorientierte Chipkarten mit synchroner und asynchroner Übertragung spezifiziert.

Teil 3 enthält eine Beschreibung einer anwendungsunabhängigen Schnittstelle für Terminals namens CT-API (Anwendungsunabhängiges Card Terminal Applikation Programming Interface). Das CT-API ist unabhängig von einer Programmiersprache und prozedural aufgebaut. Dabei stehen die folgenden 3 Funktionen zur Verfügung: „CT_init“ für die Initialisierung einer Verbindung, „CT_data“ für den Datenaustausch auf einer bestehenden Verbindung und „CT_close“ zum Schließen einer Verbindung.

Ergänzend dazu findet sich in Teil 4 die Festlegung einiger grundlegender und anwendungsunabhängiger Kommandos zur Steuerung von Terminals, welche die Bezeichnung CT-BCS (Anwendungsunabhängiges Card Terminal Basic Command Set) hat.

In Teil 5 sind für Chipkarten mit synchroner Datenübertragung der ATR und allgemeine Datenbereiche beschrieben. Teil 6 enthält die dazugehörigen Übertragungsprotokolle sowie entsprechende generelle Kommandos an das Terminal. Darauf aufbauend spezifiziert

Teil 7 die Übersetzung von ISO/IEC 7816-4-Kommandos in Kommandos für Chipkarten mit synchroner Datenübertragung, d. h. Speicherkarten.

Die MKT-Spezifikation war weltweit eines der ersten Dokumente ihrer Art und wurde in Deutschland durch Tausende Terminals für 80 Millionen Krankenversichertenkarten auf eine enorm breite Basis gestellt. Obwohl technisch mittlerweile sicherlich nicht mehr auf dem neuesten Stand, wird diese Spezifikation noch für viele Jahre einen nationalen Industriestandard darstellen.

11.4.4 MUSCLE

Der Einsatz von Chipkarten unter Linux bedingt entsprechende Treiber, wie bei allen anderen PC-Betriebssystemen auch. Diese waren jedoch lange Zeit nicht verfügbar, sodass es aufwendig war, beispielsweise Chipkarten für einen Login unter Linux zu benutzen.

Im Jahr 2000 wurde die erste Version von MUSCLE (*Movement for the Use of Smart Cards in a Linux Environment*) veröffentlicht, die genau diese Lücke schloss. MUSCLE basiert seitens der Architektur stark auf PC/SC, liegt jedoch im Gegensatz zu PC/SC im Quellcode veröffentlicht vor [MUSCLE] und steht unter GPL-Lizenz, sodass es auch von Dritten adaptiert und weiterentwickelt werden kann. Mit MUSCLE wurde eine API unter Linux geschaffen, die einen unkomplizierten Zugriff auf Chipkarten in einem angeschlossenen Terminal ermöglicht.

12 Chipkarten im Zahlungsverkehr

12.1	Zahlungsverkehr mit Karten	682
12.1.1	Elektronischer Zahlungsverkehr mit Chipkarten	682
12.1.2	Elektronisches Geld	687
12.1.3	Grundsätzliche Möglichkeiten der Systemstruktur	689
12.2	Vorbezahlte Speicherkarten	692
12.3	Elektronische Geldbörsen	693
12.3.1	CEN-Norm EN 1546	694
12.3.2	CEPS (<i>common electronic purse specifications</i>)	709
12.3.3	Proton	710
12.3.4	Das Mondex-System	711
12.4	EMV-Anwendung	716
12.5	Das ec-System in Deutschland	722

Die ursprüngliche Hauptanwendung von Chipkarten mit Mikrocontrollern war die Benutzeridentifizierung im Telekommunikationsbereich. In den letzten Jahren haben sich Chipkarten allerdings in einem weiteren Marktsegment etabliert: dem Bereich des elektronischen Zahlungsverkehrs.

Das Marktpotenzial dieses Bereiches ist aufgrund der sehr hohen Stückzahlen enorm. Man denke nur an die weltweit mehr als 1 000 Millionen ausgegebenen Kreditkarten.¹ Die zukünftigen Anwendungen von elektronischen Börsen erstrecken sich vom Ersatz der konventionellen Zahlungsmittel Geldscheine und Münzen über Einkäufe in weltweiten Netzwerken bis hin zur individuellen Bezahlung von Fernsehsendungen.

Chipkarten bieten sich durch ihre Eigenschaften geradezu für den Bereich des Zahlungsverkehrs an. Im Chip können problemlos und manipulations sicher Daten aufbewahrt werden. Die Handhabung ist aufgrund der Größe und Robustheit der Karten ebenfalls für jedermann ohne Probleme möglich. Weil Chipkarten zusätzlich auch noch als aktive Teilnehmer unabhängig von äußeren Einflüssen komplizierte Berechnungen ausführen können, ist es möglich, völlig neue Wege im Zahlungsverkehr zu beschreiten. Dies sieht man sehr deutlich am Beispiel elektronischer Geldbörsen in Form von Chipkarten, die ausschließlich mit diesem Medium realisierbar sind.

Der elektronische Zahlungsverkehr und elektronische Geldbörsen bieten allen Beteiligten nennenswerte Vorteile. Für Banken und Händler reduzieren sich die Kosten für die Bargeldbearbeitung. Durch offline arbeitende elektronische Geldbörsen entfallen weitgehend die Datenübertragungskosten für die Transaktionen. Das Risiko durch Raub und Vandalismus sinkt, da bei elektronischen Zahlungsverkehrssystemen kein Bargeld mehr durch Diebe gestohlen werden kann.

Für die Händler ist der schnellere Bezahlvorgang ebenfalls ein stichhaltiges Argument, da auf diese Weise Optimierungen im Kassenbereich möglich werden. Auch kann man einfachere und billigere Automaten bauen, da die Münz- und Geldscheinprüfeinheit nicht mehr benötigt wird. Da elektronisches Geld über beliebige Telekommunikationswege trans-

¹ Wert vom Sommer 2002.

ferierbar ist, fällt das tägliche oder wöchentliche Einsammeln des Bargeldes an Automaten weg. Auch der Kunde hat durch die neue Bezahlungsart Vorteile, selbst wenn sie nicht allzu groß sind. Das Problem des nicht vorhandenen Kleingelds fällt weg, und ein schnelleres Bezahlen an Automaten ist ebenfalls möglich.

Letztendlich entscheidet aber der zukünftige Börsenbenutzer über Erfolg oder Misserfolg eines Zahlungsverkehrssystems. Wenn die Vorteile für ihn zu unerheblich sind, wird er es nicht nutzen und andere Varianten der Bezahlung wählen. Elektronische Börsen sind eben nur eine neue Art der Zahlung, die bisherige Varianten wie Kreditkarte, Barzahlung und Ähnliches nicht ersetzen, sondern ergänzen. Eine Ablösung dieser schon seit vielen Jahren zuverlässig betriebenen Zahlungsarten durch elektronische Geldbörsen in Chipkartenform ist nicht zu befürchten.

12.1 Zahlungsverkehr mit Karten

Die einfachste Lösung für den Zahlungsverkehr mit Karten ist die Verwendung von Magnetstreifenkarten, deren Daten für eine Online-Autorisierung benutzt werden. Nach Überprüfung der Sperrliste und der Zahlungsfähigkeit kann eine Überweisung direkt vom Bankkonto des Karteninhabers zum Händler ausgeführt werden. Für Chipkarten lässt sich das Schema etwas modifizieren, bleibt aber vom Prinzip her gleich. Die Chipkarte ist logisch mit einem Bankkonto verbunden, und nach einer ein- oder gegenseitigen Authentisierung von Hintergrundsystem und Chipkarte wird die Überweisung eines vorher eingegebenen Betrages ausgeführt. Im Zuge der Zahlungstransaktion wird natürlich noch eine PIN-Prüfung in der Chipkarte oder im Hintergrundsystem durchgeführt.

Die beiden oben beschriebenen Szenarien basieren auf einem alles bestimmenden Hintergrundsystem und nutzen die Fähigkeiten von Chipkarten in keiner Weise aus. Auf den folgenden Seiten sind jedoch verschiedene Varianten für Verfahren und Zahlungsarten aufgeführt, die durch den konsequenten Einsatz von Chipkarten möglich sind.

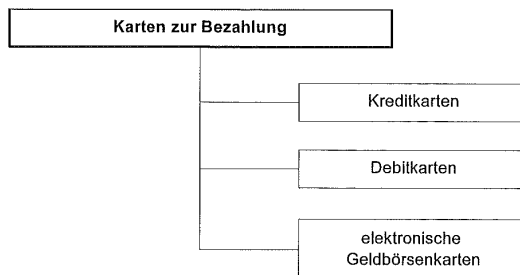


Bild 12.1 Klassifizierungsbaum für Zahlungsverkehrskarten

12.1.1 Elektronischer Zahlungsverkehr mit Chipkarten

Es gibt drei grundsätzliche Modelle für den elektronischen Zahlungsverkehr in Verbindung mit Chipkarten: Kreditkarten, bei denen nach Inanspruchnahme einer Leistung bezahlt wird (*pay later*), Debitkarten, bei denen bei Inanspruchnahme der Leistung bezahlt werden

muss (*pay now*), und elektronische Geldbörsen, welche vorbezahlt sind (*pay before*).² Im Folgenden werden die beiden Modelle und eine Variante detailliert beschrieben.

Kreditkarten (*credit card*)

Die ursprüngliche Idee, eine Plastikkarte zum Bezahlen von Waren oder Dienstleistungen zu verwenden, stammt aus dem Kreditkartenbereich. Man bezahlt, und der entsprechende Betrag wird nach einiger Zeit von einem Konto abgebucht. Die dabei entstehenden Kosten trägt der Händler. Sie sind in der Regel umsatzabhängig und bewegen sich in der Größenordnung von 2 bis 5 Prozent des Kaufpreises.

Bisher benutzt man für Kreditkarten weitgehend Karten ohne Chip. Der Nachteil der geringen Fälschungssicherheit führt bei den Kartenherausgebern zu einem nicht unerheblichen Verlust aufgrund von Fälschungen, da der Händler eine Zahlungsgarantie hat. Offensichtlich ist dieser Verlust bislang noch geringer, als die Einführung von Karten mit Chip kosten würde. Allerdings werden wohl in nicht allzu langer Zeit die Kreditkarten durch einen Chip ergänzt werden, um die steigenden Kosten durch Betrug zu senken.

Debitkarten (*debit card*)

Vor allem in Deutschland sind Debitkarten sehr verbreitet. Diese Magnetstreifenkarten oder Chipkarten ermöglichen es, unmittelbar bei einem Bezahlvorgang dem Händler oder Dienstleister den entsprechenden Betrag zu überweisen. In der Regel wird sowohl bei Debit- als auch bei Kreditkarten der eigentliche Bezahlvorgang über eine Bonitätsanfrage bei einem Hintergrundsystem autorisiert. Damit für Kleinbetragszahlungen nicht immer eine Verbindung zum Hintergrundsystem hergestellt werden muss, wird diese Anfrage erst beim Überschreiten eines Schwellenwerts in der Größenordnung von 200 Euro angestoßen.

Elektronische Geldbörsen (*electronic purse*)

Bei den elektronischen Geldbörsen wird vor der Bezahlung elektronisches Geld auf die Karte geladen. Dieses wird vom Kartenbenutzer beim Laden in bar oder mit einem bargeldlosen Verfahren erhoben. Beim eigentlichen Bezahlvorgang wird der Saldo in der Karte des Kartenbenutzers erniedrigt und parallel dazu in der elektronischen Börse des zweiten Beteiligten (d. h. normalerweise des Händlers) erhöht. Dieser reicht dann die erhaltene elektronische Geldsumme beim Börsenbetreiber ein und erhält daraufhin von diesem den entsprechenden Geldbetrag. Der Benutzer einer elektronischen Geldbörse tauscht also sein echtes Geld in eine elektronische Form von Geld um, das dann in seine Chipkarte geladen wird. Beim Bezahlen tauscht der Kartenbesitzer dann die elektronischen Werteinheiten wieder in Waren oder Dienstleistungen um.

Das Verfahren hat für den Kartenbenutzer drei erhebliche Nachteile: Beim Aufladen der Börse erhält er elektronisches Geld für echtes Geld. Wirtschaftlich betrachtet gibt er dem Börsenbetreiber also einen zinslosen Kredit, da er sein elektronisches Geld unter Umständen erst in einigen Wochen verbraucht, das echte Geld aber sofort in den Besitz des Börsenbetreibers geht. Für einen einzelnen Kartenbesitzer ist der durch diesen Kredit entgan-

² Diese Systematik lässt sich noch mit der Variante *pay never* ergänzen, die beim Betrug zum Tragen kommt.

gene Zinsgewinn gering, doch in der Summe ist es für einen Börsenbetreiber eine erhebliche Zusatzeinnahme.

In bislang mehreren Feldversuchen wurde herausgefunden, dass der durchschnittliche Geldbetrag in einer elektronischen Geldbörse in Industrieländern ungefähr 75 Euro beträgt. Dieser durchschnittliche Betrag in einer elektronischen Geldbörse wird als Bodensatz (*float*) bezeichnet. Unter der Annahme von 10 Millionen ausgegebenen Karten und einem Zinssatz von 5 % beträgt der Zinsgewinn über ein Jahr hinweg 37,5 Millionen Euro. Diesem Betrag stehen keine zusätzlichen Ausgaben gegenüber. Da der entgangene Zins für den Kartenbesitzer im vorergehenden Beispiel lediglich 3,75 Euro beträgt, ist dies für ihn kein allzu großer Nachteil. Zusätzlich zu dem Gewinn aus dem Bodensatz nimmt der Systembetreiber Geld ein und zwar durch elektronisches Geld enthaltene Sammlerkarten und nicht reklamierte defekte Karten.

Ein echter Nachteil wäre dann gegeben, wenn der Börsenbetreiber in Konkurs geht. Der Grund dafür ist Folgender: Der Kartenbenutzer hat sein echtes Geld, dessen Wert vom Staat innerhalb gewisser Grenzen garantiert wird, in elektronische Währungseinheiten auf einer Chipkarte gewechselt. Beim Konkurs des Börsenbetreibers kann es nun passieren, dass das elektronische Geld plötzlich wertlos wird. Der Kartenbenutzer hat also sein Geld verloren. Dies ist der Grund, warum in einigen Staaten mittlerweile Bestrebungen in Gange sind, diese Art von Geldbörsen nur von Banken oder ähnlichen Institutionen herausgeben zu lassen. Zumindest fordert man eine bei einer staatlichen Stelle hinterlegte Sicherheitsleistung, sodass im Falle eines Konkurses des Kartenherausgebers der in die Chipkarten geladene Betrag gedeckt ist.

Nun gibt es aber noch einen dritten nennenswerten Nachteil für den Benutzer: Was macht der Besitzer einer elektronischen Geldbörse, wenn diese nicht mehr funktioniert? Falls Letztere anonym ist, besteht auch für den Börsenbetreiber keine Möglichkeit, den letzten gespeicherten Geldbetrag herauszufinden. Für den Börsenbesitzer ist es ebenfalls praktisch unmöglich, den letzten Stand seiner Börse unwiderlegbar glaubhaft zu machen. Ist der Chip zerstört, dann ist das elektronische Geld unwiderruflich verloren. Die Robustheit einer Chipkarte im Vergleich mit Papier- oder gar Münzgeld ist aber aus verständlichen Gründen viel schlechter.

In der Praxis kennt man zurzeit einen Kompromiss: Da die letzte Online-Aufladung bekannt ist und somit auch der Börsenstand bei dieser Transaktion, kann man den ungefähren Betrag in der Börse befindlichen Betrag errechnen. Dieser wird dann dem Kunden ausbezahlt. Erhebt allerdings derselbe Kunde aufgrund von kaputten Chipkarten öfter Forderungen an den Börsenbetreiber, schränkt dieser die Kulanz ein. Somit erhält der das Risiko tragende Kunde vom Börsenbetreiber keinen Schadenersatz mehr in der Hoffnung dass er in Zukunft mit seiner Chipkarte behutsamer umgeht.

Offene und geschlossene Systemarchitekturen

Man muss bei elektronischen Zahlungsverkehrssystemen zwischen offenen und geschlossenen Architekturen unterscheiden. Ein offenes System steht grundsätzlich mehreren Anwendungsbetreibern zur Verfügung und kann für den allgemeinen Zahlungsverkehr zwischen verschiedenen Instanzen verwendet werden. Im Gegensatz dazu lässt sich ein geschlossenes System nur für Bezahlungen bei einem einzigen Systembetreiber benutzen.

Am Beispiel einer Telefonkarte mit Speicherchip sei die technische Seite hier kurz verdeutlicht. Bei Speicherkarten werden bei der Bezahlung lediglich irreversible Zähler

dekrementiert. Es ist dazu nicht notwendig, in den Terminals eine genaue Buchführung über die Anzahl der herabgezählten Einheiten zu betreiben. Es muss lediglich sichergestellt sein, dass der Zähler in der Karte bei einer Inanspruchnahme der Leistung (d. h. beim Telefonieren) auf jeden Fall herabgezählt wird. Das Terminal ist damit also eine Art „Maschine zur Vernichtung von elektronischen Geldeinheiten“. Natürlich wird man in der Praxis für jedes Terminal einen Saldo führen, doch werden die abgebuchten Einheiten nur innerhalb des Börsenbetreibers verrechnet. Ein Betrug bei der Verrechnung der Einheiten zwischen Terminalbesitzer und Börsenbetreiber ist damit vom Grundsatz her nicht möglich, da beide der gleichen Institution (im Beispiel: Telefongesellschaft) angehören.

Im Gegensatz dazu können bei offenen Systemen Terminalbesitzer und Börsenbetreiber völlig unterschiedliche Firmen sein. Der Börsenbetreiber muss deshalb bei der Abrechnung der Terminalumsätze prüfen können, ob diese korrekt und nicht manipuliert sind. Dies muss im Systemkonzept von Anfang an berücksichtigt werden, da sonst eine Abrechnung des Terminalbetreibers mit dem Börsenbetreiber sehr schwierig oder gar unmöglich ist.

Im obigen Beispiel mit der Speicherkarte ist es für einen Terminalbetreiber von der Systemphilosophie her nicht möglich, dem Börsenbetreiber glaubhaft zu versichern, dass der eingereichte Betrag korrekt ist. Dies liegt daran, dass der Terminalbetreiber nur eine bestimmte Anzahl von Einheiten zur Abrechnung vorlegen kann, jedoch keine fälschungssicheren Signaturen über die bezahlten Beträge, die von einer echten elektronischen Geldbörse stammen.

Systemaufbau und Terminalanbindung

Der Systemaufbau von elektronischen Zahlungsverkehrssystemen mit Chipkarten kann sowohl zentralistisch als auch dezentral sein. Gerade im Zahlungsverkehr ist die Systemicherheit der wichtigste Aspekt. Deshalb tendiert man oft zu einem zentralistisch aufgebauten System, weil damit der Systembetreiber alle Fäden in der Hand hat.

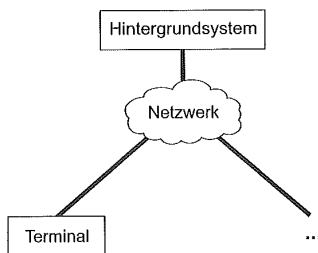


Bild 12.2 Der grundlegende Aufbau eines zentralistisch orientierten Systems für den elektronischen Zahlungsverkehr. Alle eingezeichneten Verbindungen sind permanent.

Konkret bedeutet dies ein Online-System, bei dem jede Bezahlung direkt und online mit dem Hintergrundsystem ausgeführt wird. Kommt diese Kommunikationsverbindung nicht zustande, so ist auch die Bezahlung nicht möglich. Ein zentralistisch betriebenes System hat aber trotzdem einige Vorteile: Die eingehenden Transaktionen können beispielsweise unmittelbar und in Echtzeit mit der aktuellen Sperrliste verglichen werden. Schlüsselwech-

sel lassen sich direkt am Hintergrundsystem ohne Zeitverzögerung durchführen. Ein Ändern von Software in den Terminals oder von Rahmenparametern in den Chipkarten ist ohne Umwege und ohne großen Aufwand möglich, da bei jeder Transaktion eine direkte Verbindung zum Hintergrundsystem hergestellt werden muss.

Diesen Vorteilen stehen aber einige größere Nachteile gegenüber. In vielen Ländern sind die Telekommunikationsgebühren so hoch, dass es sich für einen Händler nicht lohnt, eine Standleitung zum Hintergrundsystem einzurichten oder für jede Transaktion über eine Wählverbindung Kontakt aufzunehmen. Auch ist die Ausfallsicherheit des Telefonnetzes in manchen Gegenden nicht auf einem Niveau, das zu jedem beliebigen Zeitpunkt eine Online-Verbindung zu einem übergeordneten Computer zulässt. Aufgrund ihrer aktiven Natur eignen sich Chipkarten hervorragend für dezentrale Systeme, da sich in ihnen ein Teil der Systemsicherheit vor Ort befindet. Dies ist auch der große Vorteil gegenüber den passiven Magnetstreifenkarten, die keinerlei Abläufe im System erzwingen können.

Gerade der Einsatz von elektronischen Geldbörsen im Automatenbereich erfordert zwangsläufig ein dezentrales System, da Erstere teilweise über Wochen und Monate völlig autark arbeiten und keinerlei Möglichkeit bieten, zu einem vorhandenen Kommunikationssystem Kontakt aufzunehmen. Dies alles sind Gründe, warum ein dezentrales System oft favorisiert wird. Hinzu kommt das wesentlich günstigere Verhalten in Bezug auf Ausfallsicherheit. Fällt das Hintergrundsystem aus, dann ist bei einem zentralistischen Aufbau der gesamte elektronische Zahlungsverkehr blockiert. Bei dezentralen Systemen hingegen dringen die Auswirkungen eines zeitweiligen Ausfalls meist überhaupt nicht bis zu den Händlerterminals vor.

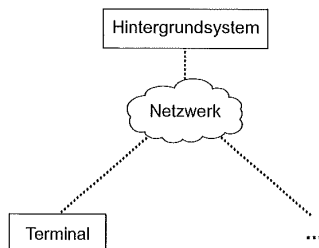


Bild 12.3 Der grundlegende Aufbau eines dezentralen Systems für den elektronischen Zahlungsverkehr. Alle eingezeichneten Verbindungen können bei Bedarf aufgebaut werden.

Auch dezentrale Systeme haben jedoch Nachteile, die vor allem im Bereich der Systemverwaltung liegen. Der Grund dafür ist die Tatsache, dass Online-Verbindungen nur in gewissen Zeiträumen hergestellt werden können und in der Regel auch nur ausgehend von den Terminals. Allerdings ist es für die Systemsicherheit unabdingbar, dass die Terminals immer mit der aktuellen Sperrliste arbeiten. Dies ist einer der Gründe, warum es üblich ist, dass Terminals in vielen Systemen mindestens einmal am Tag eine Online-Verbindung zum Hintergrundsystem aufbauen müssen. Dabei werden die entstandenen Transaktionsdaten zum Hintergrundsystem übertragen und im Gegenzug dazu verschiedene Verwaltungsdaten in der entgegengesetzten Richtung zum Terminal transferiert. Zu den Verwaltungsdaten für das Terminal zählen beispielsweise: neue Software für das Terminal, ein neuer Schlüsselsatz, die aktuelle Sperrliste und in die Karte des Kunden zu ladende Daten.

Oft wählt man in der Praxis einen Mittelweg zwischen einem reinen dezentralen und einem zentralistischen System. Man versucht damit die Vorteile, die beide Systemstrukturen bieten, zu vereinen, ohne dafür die Nachteile in Kauf nehmen zu müssen. Dabei geht man folgendermaßen vor: Sowohl das Terminal als auch die Chipkarte können aufgrund von bestimmten Bedingungen eine Online-Verbindung erzwingen. Kann diese nicht hergestellt werden, kommt der Zahlungsvorgang nicht zustande.

Einige typische Bedingungen sind im Folgenden genannt: Ab einem bestimmten Zahlungsbetrag muss eine Online-Autorisierung durchgeführt werden. Dieser Betrag kann meist individuell vom Systembetreiber in der jeweiligen Chipkarte eingestellt werden. Zusätzlich können noch die Anzahl von Offline-Transaktionen und die Zeitdauer seit der letzten Online-Transaktion für die Entscheidung, „online zu gehen“, genutzt werden. Außerdem ist manchmal noch ein Zufallsgenerator vorhanden, dessen Ergebnis dazu führt, dass ein bestimmter Prozentsatz aller Transaktionen online erfolgt. Es gibt in verschiedenen Systemen auch noch eine spezielle Taste am Terminal, die ebenfalls eine Online-Transaktion erzwingt. Diese kann vom Verkaufspersonal gedrückt werden, wenn der Verdacht besteht, dass der Kunde eine manipulierte Karte verwendet.

Tabelle 12.1 Typische Auslöser zum Aufbau einer Online-Verbindung zwischen Chipkarte und Hintergrundsystem

Auslöser	üblicher Wert
Art der Transaktion (z. B.: Bargeldauszahlung)	---
Bedingungen am POS (z. B.: PIN-Pad vorhanden o. Ä.)	---
parametrisierbarer Zufall	10 %
manuelle Anforderung am Händlerterminal	1 %
Erstbenutzung der Chipkarte	---
Anzahl getätigter Offline-Transaktionen seit der letzten Online-Transaktion	10
kumulierter Offline-Betrag seit der letzten Online-Transaktion	500 Euro
Zeitdauer seit der letzten Online-Transaktion	7 Tage
Überschreitung eines einstellbaren Schwellwertes des Zahlungsbetrags	200 Euro

Alle diese Entscheidungskriterien führen dazu, dass im Durchschnitt jede Karte innerhalb eines bestimmten und mit statistischen Methoden errechenbaren Zeitraums eine direkte Verbindung zum Hintergrundsystem aufnimmt. Der Systembetreiber gewinnt auf diese Weise wieder eine direkte Kontrolle, die er durch ein dezentrales System ursprünglich verloren hatte. Terminal und Automaten, an denen nur geringe Beträge umgesetzt werden, sind von den oben genannten Online-Zwängen auszunehmen, weil selbst im Betrugsfall nur geringer Schaden entstehen kann. Dafür spart man sich den Anschluss an ein Kommunikationsnetz, da der Datenaustausch vom Servicepersonal manuell durchgeführt wird.

12.1.2 Elektronisches Geld

Damit elektronisches Geld ähnlich flexibel wie normales Geld verwendet werden kann, muss es bestimmte Eigenschaften haben. Sind diese nicht oder nur teilweise vorhanden, so sind zwangsläufig die Möglichkeiten des elektronischen Geldes mehr oder minder stark eingeschränkt. Es folgen alle wesentlichen Eigenschaften, die notwendig sind, damit der Unterschied zwischen normalem Geld und elektronischem Geld möglichst gering ist.

Verarbeitbarkeit

Eine wichtige, doch im Prinzip triviale Eigenschaft von elektronischem Geld ist die vollständige automatische und maschinelle Verarbeitbarkeit. Nur so lassen sich große Systeme wirtschaftlich betreiben.

Übertragbarkeit

Das elektronische Geld darf nicht an ein bestimmtes Medium, wie z. B. an Chipkarten, gebunden sein. Es muss möglich sein, es durch beliebige Medien wie Netzwerke oder Computer sicher zu transferieren.

Teilbarkeit

Elektronisches Geld muss teilbar sein, damit beliebige Beträge ohne Zuhilfenahme von normalem Geld bezahlt werden können. Diese Eigenschaft ist wiederum analog dem normalen Geld, das zwar nicht beliebig teilbar ist, doch eine Stückelung aufweist, die es ermöglicht, übliche Beträge mit einer minimalen Anzahl von Münzen und Scheinen zu bezahlen.

Dezentral

Zentralistisch aufgebaute Zahlungssysteme sind für den Börsenbetreiber einfach zu überwachen, die Betrugsmöglichkeiten sind sehr eingeschränkt. Als bestes Beispiel kann man hier die Online-Autorisierung bei Kreditkarten anführen. Allerdings haben zentralistisch aufgebaute Systeme sehr viele Nachteile: Sie sind teuer, gegenüber Störungen technisch anfällig, unflexibel und schwer änder- und erweiterbar. Dezentral aufgebaute Systeme minimieren diese Nachteile. Man sieht das sehr deutlich bei Zahlungen im privaten Bereich, bei denen direkt Geld von einer Person zur nächsten wandert, ohne dass dabei eine zentrale Instanz eingeschaltet wird. Elektronisches Geld sollte diese Eigenschaft ebenfalls haben, da es nur so mit normalem Geld konkurrenzfähig ist. Auf Zahlungsverkehrssysteme angewandt, bedeutet dies konkret, dass sowohl Zahlungen offline möglich sein müssen als auch Zahlungen von einer Börse direkt zur anderen. Die Eigenschaft der direkten Zahlung zwischen zwei Börsen (*purse-to-purse transaction*) wird oft noch als Übertragbarkeit bezeichnet.

Systemüberwachung (*monitoring*)

Trotz der Forderung nach Anonymität muss elektronisches Geld dem Börsenbetreiber die Möglichkeit bieten, sein System zu überwachen. Nur so können Manipulationen oder Sicherheitslücken entdeckt und beseitigt werden. Es verhält sich hierbei völlig analog zum bisherigen Geld, bei dem jeder Bürger verpflichtet ist, ein Auftreten von Falschgeld bei den entsprechenden Behörden sofort zu melden. Im Falle von elektronischem Geld ist dafür der Börsenbetreiber zuständig, der Zahlungsströme auf Konsistenz überwachen kann und muss.

Sicherheit

Die grundlegende Eigenschaft von elektronischem Geld muss natürlich die Fälschungssicherheit sein. Jedes System würde innerhalb kurzer Zeit zusammenbrechen, wenn es möglich ist, Geld in irgendeiner Form zu fälschen, zu duplizieren oder Zahlungsströme zu manipulieren. Dies ist auch der Grund des sehr umfangreichen Einsatzes von kryptografischen Funktionen im Bereich des elektronischen Zahlungsverkehrs, da nur auf diesem Weg die geforderte Sicherheit erreicht werden kann.

Anonymität

Anonymität bedeutet, dass es niemandem möglich ist, eine Zuordnung zwischen Zahlungen und Personen zu treffen. Diese Forderung ist eine Eigenschaft, die je nach Betrachtungsweise sehr unterschiedlich bewertet wird. Der Börsenherausgeber wünscht sich vom technischen Standpunkt aus ein möglichst nicht anonymes System, damit er die Systemüberwachung optimal durchführen kann. Die Betrugsmöglichkeiten sind bei nicht anonymen Verfahren stark begrenzt, da der Betrüger sehr schnell festgestellt werden kann. Ähnlich verhält es sich mit staatlichen Institutionen, wie beispielsweise Polizei und Finanzamt, denen bei nicht anonymem elektronischen Geld wesentlich mehr Überwachungsmöglichkeiten offen stehen als bisher.

Die Position des Börsenbenutzers ist genau entgegengesetzt. Hier wird die gegenwärtige Zahlungsvariante mit normalem Geld als ein sehr guter Zustand angenommen. Eine vollständige Anonymität und Nichtverfolgbarkeit von Zahlungsströmen wird als das Optimum betrachtet. Gerade bei dieser Eigenschaft werden aber vom Börsenbetreiber aus Gründen der Systemsicherheit oft Kompromisse gemacht. So ist das Bezahlen meist anonym, das Aufladen der elektronischen Börse jedoch nicht. Auf diese Weise lässt sich einfach und ohne größeren Aufwand eine relativ gute Systemüberwachung durchführen.

Die obigen Eigenschaften sind dem vorangegangenen auf den ersten Blick teilweise genau entgegengesetzt. So schließt beispielsweise vollständige Anonymität in vielen Fällen eine ideale Systemüberwachung aus. Allerdings steht man hier erst am Anfang der Entwicklung, und es sind bereits Systeme in Planung, bei denen diese beiden Eigenschaften durchaus gleichzeitig erfüllt werden können.

Zwei Eigenschaften, die echtes Geld auszeichnen, wurden in der vorangegangenen Aufstellung nicht genannt, obwohl sie von großer Bedeutung sind. Echtes Geld ist ein gesetzliches Zahlungsmittel, das in dem jeweiligen Land von allen Personen akzeptiert werden muss. In fast allen Ländern sind die Verkäufer einer Ware oder einer Dienstleistung verpflichtet, die jeweilige gesetzliche Währung als Zahlungsmittel zu akzeptieren. Der zweite Punkt betrifft die Wertbeständigkeit des Zahlungsmittels. Bis auf einige hochinflationäre Länder sind die im Umlauf befindlichen gesetzlichen Zahlungsmittel in ihrem Wert beständig. Wäre dies nicht der Fall, käme es zu einer Flucht in Sachwerte oder Fremdwährungen.

12.1.3 Grundsätzliche Möglichkeiten der Systemstruktur

Ein elektronisches Zahlungsverkehrssystem mit Chipkarten kann auf sehr vielfältige Weise aufgebaut werden. Oft benutzt man aus wirtschaftlichen Gründen bestehende Systeme als Grundlage. Diese bauen meist auf Magnetstreifenkarten auf. Es gibt aber kein grundsätzliches Modell für den Aufbau von elektronischen Zahlungsverkehrssystemen, da die jeweiligen Anforderungen viel zu unterschiedlich sind. Deshalb werden hier, aufbauend auf den für ein solches System notwendigen Komponenten und einem Beispiel, nur die grundlegenden Prinzipien dargestellt.

Große Zahlungsverkehrssysteme mit Chipkarten bestehen grundsätzlich aus vier unterschiedlichen Komponenten. Diese sind: das Hintergrundsystem, die Terminals, das Netzwerk und die Chipkarten.

Hintergrundsystem

Das Hintergrundsystem besteht aus den beiden Teilen Clearing und Verwaltung. Im Clearing werden alle eingehenden Transaktionsdaten mit den am System beteiligten Banken, Händlern und Kartenbesitzern verrechnet. Gleichzeitig übernimmt das Clearing auch die Aufgaben der Systemüberwachung. Eine einfache Aufgabe ist beispielsweise die laufende Berechnung, dass die insgesamt zum Clearing eingereichten Beträge niemals größer sind als der Gesamtbetrag, der sich in den elektronischen Geldbörsen befindet. Wäre dies der Fall, hat ein Angreifer ohne Kenntnis des Hintergrundsystems Geld in die Chipkarten geladen.

Im Verwaltungsteil des Hintergrundsystems werden alle administrativen Abläufe geregelt. So zum Beispiel die Verteilung von neuen Sperrlisten, Umschalten auf neue Schlüsselversionen, Softwareänderungen in den Terminals und Ähnliches. Auch die Erstellung der Datensätze für die Personalisierung der Chipkarten geschieht hier. Im Hintergrundsystem laufen alle Fäden des elektronischen Zahlungsverkehrssystems zusammen. Dies ist völlig unabhängig von der Systemstruktur. Selbst bei vollständig offline arbeitenden Systemen werden hier die globalen Systemparameter gesetzt und findet die Überwachung auf Sicherheit und Funktionsweise statt.

Netzwerk

Das Netzwerk verbindet das Hintergrundsystem mit den Terminals. Es kann eine leitungsorientierte (z. B. ISDN) oder paketorientierte (z. B. X.25) Anbindung sein. Das Netzwerk ist im Regelfall völlig transparent zu den Nachrichten, welche ohne Veränderung vom Sender zum Empfänger weitergeleitet werden.

Terminals

Die verschiedenen Terminals kann man nach ihrer Funktion im Zahlungsverkehr in Lade- und Bezahlterminals einteilen. Man kann sie aber auch in Automaten und Terminals mit Bedienpersonal einteilen. Das klassische Beispiel für Automaten sind Geldausgabeautomaten. Sie werden bei Börsensystemen meist nur zum Aufladen der Karten genutzt. Vorstellbar wäre natürlich auch eine Funktion zum Entladen der elektronischen Geldbörse auf den Chipkarten und somit zur Ausgabe von Bargeld. Terminals mit Bedienpersonal stehen typischerweise an Kassen von Supermärkten oder in Einzelhandelsgeschäften. Die Verwendung ist in diesen Fällen immer die Bezahlung einer Ware. Terminals in Banken können aber je nach System auch zum Aufladen von Chipkarten gegen Bargeld benutzt werden.

Chipkarten

Als Komponente mit dem höchsten Verbreitungsgrad dienen Chipkarten, beispielsweise als elektronische Geldbörse. Sie können jedoch auch als Sicherheitsmodul in den verschiedenen Terminalarten eingesetzt werden. Ein weiteres Einsatzgebiet ist der Transport von Daten zwischen verschiedenen Komponenten im System. Diese Karten werden als Transferkarten bezeichnet und dienen zur manuellen Übertragung von Transaktionsdaten zwischen einem vollständig offline arbeitenden und einem online arbeitenden Terminal (z. B.: Geldausgabeautomat).

Anhand eines Beispiels (Bild 12.4) sollen die obig erläuterten Komponenten und die logischen Verbindungen im System nochmals verdeutlicht werden. Das Hintergrundsystem ist über ein transparentes Netzwerk mit den verschiedenen Komponenten verbunden. Dies kann ein Hintergrundsystem eines weiteren Betreibers sein oder die Komponenten des eigenen Systems.

Geldausgabeautomaten werden meist zum Aufladen von elektronischen Geldbörsen benutzt und arbeiten in vielen Fällen online, obwohl sie bei einem Netzzusammenbruch eine bestimmte Zeit lang auch offline arbeiten können. Aus diesem Grund verfügen sie über ein eigenes Sicherheitsmodul, in dem alle für den Betrieb relevanten Schlüssel sowie die Ableitungsverfahren aufbewahrt sind.

Nun gibt es auch noch vollständig offline arbeitende Systeme zum Bezahlen mit der elektronischen Geldbörse. Als Beispiel könnte man hier Parkuhren oder Terminals in Taxis nennen. Mithilfe einer Transferkarte können die Transaktionsdaten vom Sicherheitsmodul zu einem Geldausgabeautomaten gebracht werden, von dem sie dann über das Netzwerk zum Hintergrundsystem gelangen. In anderer Richtung erhält dann das Terminal aktuelle Verwaltungsdaten, wie z. B. Sperrlisten, Softwareänderungen und Ähnliches.

Der zweite Terminaltyp zum Bezahlen ist mit einer bei Bedarf aufbaubaren Online-Verbindung an das Netzwerk angebunden. Dieses Terminal wird normalerweise offline arbeiten und periodisch eine Verbindung zum Hintergrundsystem aufnehmen, mit dem es die anfallenden Abrechnungs- und Verwaltungsdaten austauscht. Der dritte Terminaltyp im Verkaufsbereich hat keine direkte Verbindung zum Netzwerk. Er ist beispielsweise an eine Kasse in einem Supermarkt angeschlossen, die wiederum mit einem Konzentrator, der sich in der Supermarktfiliale befindet, verbunden ist. Dieser Konzentrator, der üblicherweise ein als Server fungierender PC ist, nimmt beispielsweise einmal pro Tag Verbindung via Netzwerk zum Hintergrundsystem auf. Dabei findet der notwendige Datenaustausch statt.

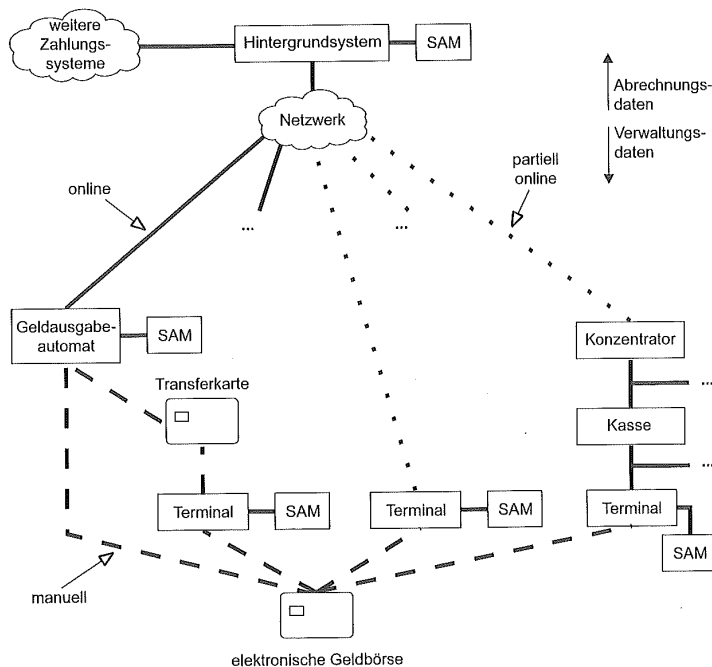


Bild 12.4 Beispielhafte Systemarchitektur eines elektronischen Zahlungsverkehrssystems für eine elektronische Geldbörse (SAM – Sicherheitsmodul).

Das beschriebene System existiert in ähnlicher Form als elektronische Geldbörse „Quick“ in Österreich, als „Geldkarte“ in Deutschland, und auch die elektronische Geldbörse Visa Cash entspricht in weiten Teilen obiger Darstellung. Bei größeren Anwendungen ist es durchaus üblich, dass eine auf mehrere verschiedene und parallel arbeitende Hintergrundsysteme verteilte Systemstruktur besteht, mit der dann auch unterschiedliche Börsensysteme mit mehr als einem Börsenbetreiber kompatibel zueinander betrieben werden.

12.2 Vorbezahlte Speicherkarten

Gerade im Zusammenhang mit elektronischem Zahlungsverkehr muss man auch Speicherkarten berücksichtigen. In der Form vorbezahlter elektronischer Börsen werden sie in sehr hohen Stückzahlen gefertigt und in vielen Anwendungen eingesetzt. Dies wird sich in den nächsten Jahren mit Sicherheit noch nicht ändern. Eine Ablösung durch Mikrocontroller-Chipkarten wird zwar langsam, aber stetig erfolgen, doch liegt die Stärke der Speicherkarten in ihrem konkurrenzlos günstigen Preis. Die typische Anwendung von vorbezahlten elektronischen

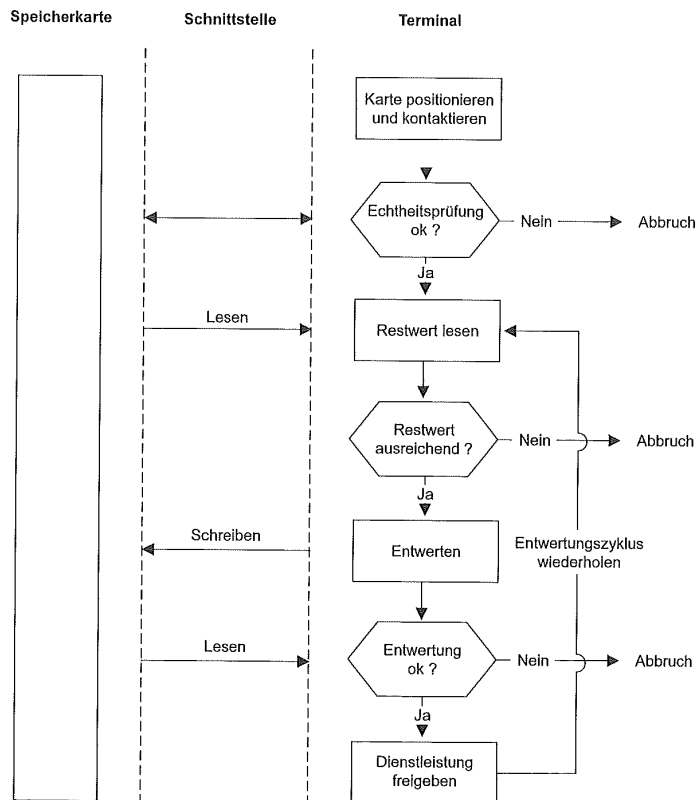


Bild 12.5 Der Entwertungszyklus einer vorbezahlten Speicherkarte aus Sicht des Terminals.

Geldbörsen mit Speicherchip sind die in vielen Ländern verbreiteten Telefonkarten, die weg-
geworfen werden, nachdem sie abtelefoniert worden sind.³

Speicherkarten⁴ besitzen für ihre Funktionen lediglich eine Logikeinheit sowie einen ir-
reversiblen Abwärtszähler. Neuere Ausführungen ermöglichen zusätzlich noch eine einsei-
tige Authentisierung der Karte durch das Terminal. Dazu ist die Logikeinheit durch eine
einfache Verschlüsselungsfunktion ergänzt worden, deren Aufgabe es ist, eine übergebene
Zufallszahl mit einem auf der Speicherkarte befindlichen geheimen Schlüssel zu ver-
schlüsseln und an das Terminal anschließend zu retournieren. Nur so kann das Terminal
sicherstellen, dass die verwendete Speicherkarte echt ist.

Neben der schwierigen Sicherstellung der Echtheit haben Speicherkarten noch einen
weiteren Nachteil, der ihren Einsatz als generelles elektronisches Zahlungsmittel beein-
trächtigt. Aufgrund der einfachen Manipulationsmöglichkeiten ist es schwierig, ein Zah-
lungsverkehrssystem aufzubauen, das verschiedene unabhängige Leistungsanbieter (z. B.:
Kiosk, Taxi) unterstützt, weil keine gesicherte Kommunikation zwischen Terminal und
Speicherkarte möglich ist und sich aufgrund dessen eine Zuordnung der bezahlten Beträge
und somit eine Systemüberwachung vom Ansatz her nur auf Umwegen durchführen lässt.

Speicherkarten haben in bestimmten und eng begrenzten Einsatzgebieten gegenüber
Mikroprozessorkarten aufgrund ihres geringen Preises Vorteile, doch eignen sie sich wenig
für offene Anwendungen im Zahlungsverkehr. Der Zahlungsverkehr der Zukunft wird
sicherlich vor allem mit Mikroprozessorkarten ablaufen, da diese wesentlich flexibler sind.

12.3 Elektronische Geldbörsen

Die Idee, eine elektronische Geldbörse auf einer Chipkarte zu verwirklichen, geht bis in
die Anfänge der Chipkartentechnik zurück. Doch erst seit Mitte der 90er-Jahre ist dieses
Thema wirklich aktuell. Damals hat man begonnen, große Systeme zu realisieren.

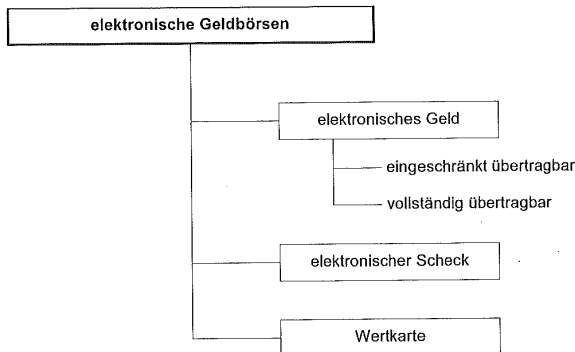


Bild 12.6 Klassifizierungsbaum für elektronische Geldbörsen auf der Grundlage von Chipkarten. Elekt-
ronisches Geld ist nicht kartegebunden und kann nach Bedarf gestückelt werden. Hingegen
weisen elektronische Schecks eine feste Stückelung auf. Wertkarten (elektronische Geldbör-
sen) enthalten kartegebundenes elektronisches Geld in freier Stückelung.

³ Siehe auch Abschnitt 13.6 „Öffentliches Kartentelefon in Deutschland“.

⁴ Aufbau von Chips für Speicherkarten, siehe auch Abschnitt 2.3.1 „Speicherkarten“.

Nimmt man eine normale Börse mit Münz- und Papiergeld als Grundlage, dann erkennt man die Eigenschaften, die eine elektronische Geldbörse für den Benutzer hat. Sie muss vorbezahlt sein und lässt sich also nicht wie eine Debit- oder Kreditkarte benutzen, mit der bei oder erst nach Inanspruchnahme der Leistung bezahlt wird, sondern wie eine Telefonkarte, die im Voraus bezahlt werden muss. Der Vorgang des Bezahlens selber muss schnell und unkompliziert ablaufen, da sonst die Akzeptanz sinkt. Außerdem finden bei einer Geldbörse alle Bezahlungen anonym statt, es kann also nicht mehr zurückverfolgt werden, welche Person zu welchem Zeitpunkt was gekauft hat. Die ärgerlichste Eigenschaft wird beim Verlust deutlich. Das in ihr enthaltene Geld ist unwiderruflich verloren. Bei einer elektronischen Geldbörse muss dies jedoch nicht unbedingt der Fall sein.

Der größte Vorteil einer Börse bzw. des in ihr enthaltenen Bargeldes besteht jedoch darin, dass sie bzw. es überall innerhalb der Grenzen eines Landes akzeptiert wird. Genau dies fehlt aber bei den meisten bestehenden elektronischen Systemen. Mit einer Telefonkarte kann man eben nur telefonieren und sonst nichts; was typisch für geschlossene Anwendungen ist. Die ideale elektronische Geldbörse ist jedoch branchenübergreifend und ermöglicht es dem Benutzer, mit seiner Karte in einer Vielzahl verschiedener Geschäfte zu bezahlen.

12.3.1 CEN-Norm EN 1546

Von der Europäischen Kommission wurde im Jahr 1990 beschlossen, eine europäische Norm für eine branchenübergreifende Geldbörse durch CEN (*Comité Européen de Normalisation*) erstellen zu lassen. Mit den Arbeiten an der Norm wurde 1991 begonnen. Bis 1998 wurde von den einzelnen Projektteams ein Arbeitsaufwand von etwa 6 Personenjahren in die Normungsarbeiten gesteckt. Da an diesen Arbeiten unterschiedliche und voneinander unabhängige Experten beteiligt waren, hat diese Norm mit hoher Wahrscheinlichkeit keine Sicherheitslücken und ist so quasi evaluiert. Die wesentlichen Teile sind mittlerweile in ihrem Inhalt stabil und werden nach der Schlussabstimmung als europäische Norm veröffentlicht.

Die EN 1546 Norm ist öffentlich, und die Abläufe bei den einzelnen Funktionen sind sehr detailliert beschrieben. Deshalb eignet sie sich sehr gut zur Demonstration von Bezahl- und Aufladevorgängen mit einer elektronischen Geldbörse. Bei vielen existierenden Systemen kann dies nicht so detailliert gezeigt werden, da die betreffenden Kommandos, Abläufe und internen Funktionen vertraulich sind. Deshalb eignet sich diese Norm sehr gut, um grundlegende externe und interne Abläufe einer elektronischen Geldbörse zu veranschaulichen.

Wo das typische Einsatzgebiet liegt, zeigt sich bei den Systemen, deren Grundlage diese Norm darstellt. Der Systembetreiber Danmønt in Dänemark hat eine dieser Norm entsprechende Geldbörse innerhalb seines bestehenden Systems eingeführt. In Österreich befindet sich auf der landesweit ausgegebenen ec-Karte neben anderen Anwendungen eine elektronische Geldbörse, die ebenfalls die EN 1546 Norm als Grundlage verwendet (Geldbörse „Quick“). Die größte internationale Anwendung von EN 1546 ist jedoch Visa Cash. Dabei handelt es sich um eine von mehreren elektronischen Geldbörsen, die Visa anbietet. Die internationale Spezifikation für elektronische Geldbörsen CEPS (*common electronic purse specifications*) basiert ebenfalls in ihren wesentlichen Grundzügen auf EN 1546.⁵

⁵ Siehe auch Abschnitt 12.3.1 „CEN-Norm EN 1546“.

Die CEN-Norm für elektronische Geldbörsen EN 1546 hat den Titel „Branchenübergreifende elektronische Geldbörse“ und ist in vier Teile gegliedert: Im ersten Abschnitt „Konzepte und Strukturen“ wird das Gesamtsystem beschrieben. In dem Grundlagenpapier sind in abstrakter Form alle logischen Komponenten und die Verbindungen untereinander definiert und erklärt.

Aufbauend auf diesen Grundlagen findet man im zweiten Teil die Sicherheitsarchitektur sowohl für das Gesamtsystem als auch für die einzelnen Komponenten. Hier sind jedoch nicht nur Mechanismen zur Aufrechterhaltung der Sicherheit aufgeführt, sondern auch mögliche Angriffe und dazu notwendige Gegenmaßnahmen.

In Teil drei mit der Bezeichnung „Datenelemente und Protokolle“ befinden sich die Beschreibungen und Definitionen der für die elektronische Geldbörse benötigten Datenelemente. Auch enthält er die Kommandos und dazugehörigen Antworten in Verbindung mit der Chipkarte und den verwendeten Sicherheitsmodulen.

Im abschließenden Teil vier der Norm sind die Zustandsautomaten und Zustände der verwendeten Geräte beschrieben. Dazu benutzte man eine den bekannten Ablaufdiagrammen ähnliche, symbolische Darstellungsweise. Dieser als SDL-Notation bekannte Formalismus stammt aus der CCITT Z.100-Empfehlung.⁶

Die etwa 300 Seiten umfassende Norm enthält damit die vollständige Beschreibung einer elektronischen Geldbörse von der Chipkarte über die Terminals mit den Sicherheitsmodulen bis zum Hintergrund- und Clearingsystem. Das Ziel ist die Standardisierung von großen Systemen für elektronische Geldbörsen mit einer sehr hohen Anzahl von Chipkarten und einer großen Verbreitung.

Der Vorteil einer allgemeinen Norm für elektronische Geldbörsen liegt vor allem darin, dass einzeln und unabhängig voneinander betriebene Systeme zueinander kompatibel werden. Ähnlich wie bei GSM besteht dann in Zukunft für den Benutzer die Möglichkeit, auch in Systemen anderer Börsenanbieter mit seiner Chipkarte zu bezahlen. Dies ist eine wesentliche Voraussetzung für den Erfolg eines Zahlungsverkehrssystems dieser Art.

Allerdings muss zu diesem Punkt eine kleine Anmerkung gemacht werden. Die Norm EN 1546 lässt sehr großen Spielraum für die letztendliche Implementierung und sieht sich mehr als Rahmenwerk denn als genaue Angabe der verwendeten Bits und Bytes. Deshalb ist es ohne weiteres möglich, dass zwei unterschiedliche Systeme dieser Norm vollständig entsprechen, doch zueinander nicht kompatibel sind, da beispielsweise ein anderer Kryptalgorithmus gewählt wurde.

Der Börsenanbieter (*purse provider*) trägt die Gesamtverantwortung und ist auch der Verwalter des Systems. Er ist vergleichbar mit den Netzbetreibern bei GSM. Für den Benutzer der elektronischen Geldbörse wurde der Begriff „Börseninhaber“ in der Norm festgelegt. Er ist derjenige, der mit der Börsenanwendung auf der Chipkarte bezahlt und dafür Waren oder Dienstleistungen erhält.

Daneben existieren noch drei weitere Teilnehmer, die Aufgaben im System übernehmen. Der Leistungsanbieter (*service provider*) ist derjenige, der die Leistungen anbietet, die der Benutzer in Anspruch nimmt und dann mit seiner elektronischen Geldbörse bezahlt.

⁶ Siehe auch Abschnitt 4.3 „SDL-Symbolik“.

Der Acquirer⁷ (*acquirer*) übernimmt Errichtung und Verwaltung der datentechnischen Verbindungen zwischen Börsenherausgeber und den Leistungsanbietern. Zusätzlich kann er die von der Zahlungseinrichtung kommenden einzelnen Transaktionen zusammenfassen, sodass der Börsenanbieter nur mehr gesammelte Zertifikate erhält. Der Aufladebevollmächtigte (*load agent*) ist das Gegenstück zum Leistungsanbieter, da er die Geldbörse gegen Bezahlung wieder aufladen kann.

Die oben vorgestellten fünf Teilnehmer müssen nicht real existieren, sie sind nur virtueller Natur. Ihnen sind aber jeweils real existierende technische Komponenten zugeordnet, die nach ihrer Sicherheit aufgeteilt sind. In den sicheren Komponenten können Informationen verarbeitet und gespeichert werden, ohne dass die Möglichkeit besteht, sie von außen zu manipulieren. Bei den nicht sicheren ist dies – zumindest theoretisch – möglich. Das Gesamtsystem ist dabei aber so ausgelegt, dass Manipulationen an den im Bild 12.8 als nicht sicher gekennzeichneten Komponenten zu keinen Auswirkungen hinsichtlich der Gesamtsicherheit führen.

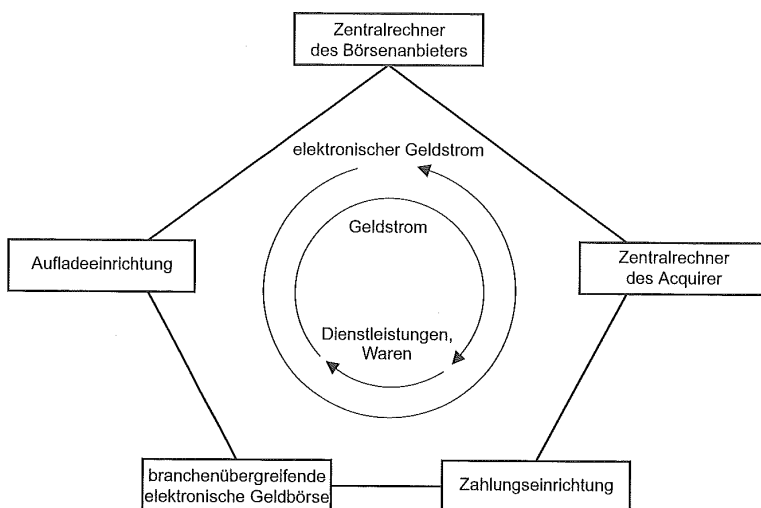


Bild 12.7 Der grundlegende Aufbau der branchenübergreifenden elektronischen Geldbörse und die dazugehörigen Zahlungsströme nach EN 1546.

„IEP“ ist die englische Abkürzung für „inter-sector electronic purse“ und steht für die Anwendung der branchenübergreifenden elektronischen Geldbörse in einer Chipkarte. Die Zahlungseinrichtung benutzt man zum Bezahlen von erhaltenen Waren oder Dienstleistungen. Es ist ein Terminal mit Tastatur und Display und muss zusätzlich über ein Sicherheitsmodul verfügen. Diese werden in der Norm einheitlich als SAM (*secure application module*) bezeichnet. In dieser sicheren Komponente befinden sich alle geheimen Schlüssel, die für die Aktionen zwischen IEP und Zentralrechner des Börsenanbieters notwendig sind.

⁷ Der Acquirer wird manchmal auch als Sammelbeauftragter bezeichnet.

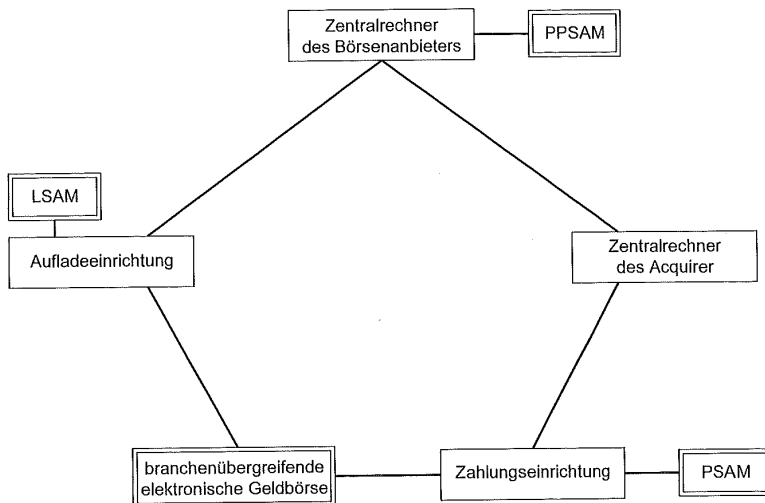


Bild 12.8 Die Komponenten und dazugehörigen Verbindungen bei elektronischen Geldbörsen nach EN 1546. Die einfach umrahmten Komponenten der Zeichnung sind im Gegensatz zu den doppelt umrahmten Komponenten nicht sicher.

Die Schlüssel verlassen natürlich nie das Sicherheitsmodul, sondern werden nur intern von den Kryptoalgorithmen benutzt.⁸ Es bestehen damit in vielen Fällen nur direkte Verbindungen zwischen den sicheren Komponenten des Systems. Die nicht sicheren Komponenten benutzt man nur, um die sicherheitsrelevanten Daten transparent weiterzuleiten.

Erst durch Chipkarten ist es möglich geworden, die Idee einer elektronischen Geldbörse auch in die Tat umzusetzen. Deshalb sind Chipkarten auch der zentrale Bestandteil der Systembeschreibung in der CEN-Norm EN 1546. Dort sind alle dazugehörigen Dateien, Kommandos, Zustände und Abläufe definiert und beschrieben. Damit nicht nur die Chipkarte, sondern das gesamte System definiert ist, gibt es auch analoge Teile für die Sicherheitsmodule in den anderen Komponenten.

Da es sich aber um eine Norm handelt und nicht um eine Spezifikation, wurden dem Börsenanbieter natürlich viele Freiräume und Optionen gelassen. Die Börse kann mit den unterschiedlichsten Funktionalitäten aufgebaut werden. So ist es ohne weiteres möglich, in einem einfachen System nur Laden und Bezahlen mit der elektronischen Geldbörse zu realisieren. In weiteren Ausbaustufen können dann zusätzlich Storno von Bezahlungen, Änderung von Börsenparametern und Währungsumrechnung eingefügt werden. Die genaue Wahl der sehr vielschichtigen Optionen ist weitgehend dem Börsenanbieter überlassen. Er muss für seine Bedürfnisse das Optimale aus den Möglichkeiten herausuchen, die die Norm anbietet.

Für die Chipkarte, also die IEP, sind hier im Folgenden die wesentlichen Aspekte der Anwendung „branchenübergreifende elektronische Geldbörse“ aufgeführt.

⁸ Ein mögliches Schlüsselmanagement für eine elektronische Geldbörse nach EN 1546 ist in Abschnitt 4.8 „Schlüsselmanagement“ erläutert.

Datenelemente bei EN 1546**Tabelle 12.2** Überblick über die wichtigsten standardisierten Datenelemente von EN 1546. Für die deutsche Übersetzung wurden die Bezeichnungen der deutschen Ausgabe der Norm verwendet.

Datenelement	Beschreibung
ALG _{IEP}	Kryptografischer Algorithmus für eine IEP (<i>cryptographic algorithm used by an IEP</i>)
AM _{IEP}	Authentisierungsmodus für eine IEP (<i>authentication mode required by an IEP</i>)
AP _{IEP}	Anwendungsprofil einer IEP (<i>application profile of an IEP</i>)
BAL _{IEP} , BAL _{PSAM} , BAL _{PPSAM}	Saldo einer/eines IEP/PSAM/PPSAM (<i>balance of an IEP/PSAM/PPSAM</i>)
BALmax _{IEP}	Maximaler Saldo einer IEP (<i>maximum balance of an IEP</i>)
CC _{IEP} , CC _{PSAM} , CC _{PPSAM}	Beendigungscode einer/eines IEP/PSAM/PPSAM (<i>completion code from the IEP/PSAM/PPSAM</i>)
CT	Beendigungsstatus für eine Endsumme (<i>collection status</i>)
CURR _{IEP} , CURR _{LDA} , CURR _{PDA}	Währung einer IEP/PSAM/PPSAM (<i>actual currency for an IEP/LDA/PDA</i>)
DACT _{IEP}	Aktivierungsdatum einer IEP (<i>activation date of an IEP</i>)
DD	Benutzerbestimmte Daten (<i>discretionary data</i>)
DDEA _{IEP}	Deaktivierungsdatum einer IEP (<i>deactivation data of an IEP</i>)
DEXP _{IEP}	Ablaufdatum einer IEP (<i>expiry date of an IEP</i>)
ID _{IEP} , ID _{PSAM} , ID _{PPSAM}	Identifikation für eine(n) IEP/PSAM/PPSAM (<i>identifier for an IEP/PSAM/PPSAM</i>)
IEP	branchenübergreifende elektronische Geldbörse (<i>inter-sector electronic purse</i>)
IKIEP, IKPSAM, IKPPSAM	Schlüsselinformation für eine(n) IEP/PSAM/PPSAM (<i>key information for an IEP/PSAM/PPSAM</i>)
LDA	Aufladeeinrichtungsanwendung (<i>load device application</i>)
LSAM	Auflade-SAM (<i>load SAM</i>)
MLDA, MPDA	Transaktionsbetrag für Laden/Bezahlen (<i>transaction amount for load/purchase</i>)
MTOT IEP, MTOTPSAM	Endsummenbetrag der Transaktion für eine Zahlung (<i>total transaction amount for a purchase</i>)
NC	Bezeichner für eine Endsumme (<i>number of collection</i>)
NI	Anzahl einzelner Transaktionen für eine Endsumme (<i>number of individual transactions</i>)
NTIEP, NTLSAM, NTPSAM	Transaktionsnummer für eine(n) IEP/PSAM/PPSAM (<i>transaction number for an IEP/LSAM/PSAM</i>)
PDA	Zahlungseinrichtungsanwendung (<i>purchase device application</i>)
PIIEP, PPPSAM, PPPPSAM	Bezeichner eines Börsenanbieters für eine(n) IEP/PSAM/PPSAM (<i>purse provider identifier for an IEP/PSAM/PPSAM</i>)
PPSAM	Börsenanbieter-SAM (<i>purse provider SAM</i>)
PSAM	Zahlungs-SAM (<i>purchase SAM</i>)
R	Zufallszahl (<i>random number</i>)

Tabelle 12.2 (Fortsetzung)

Datenelement	Beschreibung
S ₁	Signatur einer IEP (<i>IEP signature</i>)
S ₂	Signatur eines PSAM/PPSAM (<i>PSAM/PPSAM signature</i>)
S ₃	Signatur einer IEP (<i>IEP signature</i>)
S ₄	Signatur eines PSAM (<i>PSAM signature</i>)
SAM	Anwendungssicherheitsmodul (<i>secure application module</i>)
TM	Endsummenbetrag für eine Endsumme (<i>total amount</i>)
TRT	Transaktionstyp und Zustand (<i>transaction type and status</i>)

Zur eindeutigen Angabe aller für die elektronische Geldbörse systemweit verwendeten Daten wurde eine Kennzeichnung für alle Datenelemente eingeführt. Anhand der sehr kurzen Bezeichnungen können der Datenfluss und die Verarbeitung einfach und eindeutig in mathematisch korrekter Schreibweise dargestellt werden. In der Norm existiert ebenfalls ein einfaches Data Dictionary, in dem die entsprechenden Dateninhalte und dazugehörigen Formate für die standardisierten Datenelemente beschrieben sind.

Dateien

Die vollständige Anwendung „elektronische Geldbörse“ ist in der Chipkarte in einem eigenen DF untergebracht. Alle Dateien, die für das einwandfreie Funktionieren notwendig sind, befinden sich darin. Zusätzlich sind noch Informationen über die Karte, den Chip, weitere Anwendungen und Ähnliches in mehreren Dateien direkt unter dem MF gespeichert.

Die für den Betrieb der Börse notwendigen Datenelemente befinden sich in sechs Dateien des Typs EF innerhalb eines Verzeichnisses für die Börse. Tabelle 12.4 zeigt die Dateien und die in ihnen enthaltenen Datenelemente im Überblick.

Tabelle 12.3 Die notwendigen Dateien mit den zugeordneten Datenelementen für die elektronische Geldbörse nach EN 1546. Die Protokolldateien für Währungsumrechnung und Parameteränderung sind nicht dargestellt.

Datei	Funktion	Enthaltene Datenelemente und Beschreibung	
EF _{IEP}	Feste Daten und Parameter über die Börse	PP _{IEP}	Bezeichner des Börsenanbieters für die IEP
		ID _{IEP}	Identifikation der IEP
		DEXP _{IEP}	Ablaufdatum der IEP
		DACT _{IEP}	Aktivierungsdatum der IEP
		DDEA _{IEP}	Deaktivierungsdatum der IEP
		AM _{IEP}	Authentisierungsmodus der IEP
		AP _{IEP}	Anwendungsprofil der IEP
		DD	Benutzerbestimmte Daten
EF _{IK}	Informationen über alle Schlüssel	ALG _{IEP}	Kryptografischer Algorithmus der IEP
		IK _{IEP}	Schlüsselinformation über die IEP
		DD	Benutzerbestimmte Daten
EF _{BAL}	Guthaben der Börse	BAL _{IEP}	Saldo der IEP
		CURR _{IEP}	Währung der IEP
		BAL _{maxIEP}	Maximaler Saldo der IEP
		DD	Benutzerbestimmte Daten
EF _{TFIELD}	Transaktionsfeld	NT _{IEP}	Transaktionsnummer der IEP

Tabelle 12.3 (Fortsetzung)

Datei	Funktion	Enthaltene Datenelemente und Beschreibung	
EF _{LOG}	Protokolldatei für Aufladevorgänge	TRT	Transaktionstyp und Zustand
		NT _{IEP}	Transaktionsnummer der IEP
		BAL _{IEP}	Saldo der IEP (der neue Saldo)
		M _{LDA}	Transaktionsbetrag für das Laden
		CURR _{LDA}	Währung für das Laden
		ID _{PPSAM}	Bezeichner des Börsenanbieters für den PPSAM
		CC _{IEP}	Beendigungscode der IEP
		DD	Benutzerbestimmte Daten
EF _{PLOG}	Protokolldatei für Bezahlvorgänge	TRT	Transaktionstyp und Zustand
		NT _{IEP}	Transaktionsnummer der IEP
		BAL _{IEP}	Saldo der IEP (der neue Saldo)
		MP _{DA}	Transaktionsbetrag für das Bezahlen
		CURR _{PDA}	Währung für das Bezahlen
		ID _{PSAM}	Bezeichner des Börsenanbieters für den PSAM
		NT _{PSAM}	Transaktionsnummer des PSAM
		CC _{IEP}	Beendigungscode der IEP
		DD	Benutzerbestimmte Daten

Im EF_{IEP} sind die Rahmenparameter der Börse festgelegt, welche die Grundlage für alle stattfindenden Transaktionen bilden. Die Datei EF_{IK} enthält für jeden vorhandenen Schlüssel spezifische Informationen. In EF_{BAL} steht der aktuelle Betrag, der sich noch in der Börse befindet und über den der Benutzer noch frei verfügen kann. Die drei Protokolldateien sind ausschließlich dazu da, alle Transaktionen getrennt nach Funktion aufzuzeichnen. Erst dadurch wird ein Storno von Transaktionen oder eine Fehlerbehandlung überhaupt möglich. Es gibt sowohl für das Aufladen, das Bezahlen, das Ändern von Börsenparametern und die Währungsumrechnung eigene Dateien. Sie haben alle die Struktur cyclic, um so die letzten Transaktionen zu protokollieren.

Kommandos⁹

Die Dateien sind das Fundament der Börse, auf ihnen bauen die Kommandos auf. Acht davon sind zum Betrieb notwendig. Drei Kommandos gehören zur Norm ISO/IEC 7816-4: SELECT FILE, READ BINARY und READ RECORD. Sie benutzt man nur, um die Anwendung „elektronische Börse“ mit einem AID auszuwählen und danach bei Bedarf diverse Börsendaten aus den Dateien zu lesen.

Die anderen fünf Kommandos sind ganz speziell für die Verwendung in einer elektronischen Geldbörse entwickelt worden. Sie werden für die einzelnen Transaktionen immer paarweise verwendet, da sie dem Prinzip nach wie eine gegenseitige Authentisierung funktionieren. Parallel zur Authentisierung werden noch die für die Börsentransaktionen notwendigen Daten ausgetauscht. Die entsprechenden Kommandos und Antworten sind natürlich so aufgebaut, dass Manipulationen auf der Schnittstelle zwischen Terminal und Chip-

⁹ Eine detaillierte Befehlserklärung findet sich in Abschnitt 7.14 „Befehle für elektronische Geldbörsen“.

karte sofort erkannt würden und einen unmittelbaren Abbruch der Transaktion mit Protokollierung zur Folge hätten.

Alle Börsenkommandos greifen sowohl lesend als auch schreibend direkt auf Datenelemente in den Börsendateien zu, die vorher automatisch vom Betriebssystem selektiert werden. Beispielsweise benötigt man während der Kommandoabarbeitung bisweilen grundlegende Informationen über die Börse. Dazu selektiert das Betriebssystem die Datei EFIEP, und das gewünschte Datenelement wird dann dem Kommando zur Verfügung gestellt. Alle Transaktionen und die wichtigsten Daten werden im Rahmen der Kommando-Antwort-Zyklen in entsprechende Protokolldateien geschrieben. Die folgenden Börsenkommandos sind nach EN 1546 definiert und auch in ihrer karteninternen Funktion genau festgelegt:

Tabelle 12.4 Die spezifischen Kommandos der elektronischen Geldbörse nach EN 1546.

Kommando	Funktion
INITIALIZE IEP	– Initialisierung für ein nachfolgendes Börsenkommando
CREDIT IEP	– Aufladen der Börse – Storno einer getätigten Zahlung – Fehlerbehebung
DEBIT IEP	– Bezahlen mit der Börse – Bestätigung der Bezahlung
CONVERT IEP CURRENCY	– Währungsumrechnung
UPDATE IEP PARAMETER	– Ändern der Rahmenparameter der Börse

Innerhalb der Norm sind keine Kommandos zur Überprüfung oder Änderung einer PIN vorgesehen, da dies für die korrekte Funktionsweise einer Börse nicht notwendig ist. Allerdings könnte man bei Bedarf zusätzliche Kommandos zur Überprüfung und Verwaltung einer PIN in die Anwendung einbinden, ohne dass es zu Überschneidungen oder Problemen mit den vorhandenen Börsenkommandos käme.

Zustände

Wie schon aus dem Überblick der Kommandos zu erkennen ist, setzt sich jede Transaktion aus einem einleitenden Initialisierungskommando und einem anschließenden Kommando, das dann die Transaktion abschließt, zusammen. Um die Reihenfolge der Kommandos festzulegen, sind in Zustandsdiagrammen die dafür notwendigen Zustände und Zustandsübergänge der Börsenanwendung festgelegt. Dies erfordert natürlich einen Zustandsautomaten für die Chipkarte. Je nach aktuellem Zustand akzeptiert dann die Chipkarte verschiedene Kommandos oder blockt sie ab.

Kryptografische Algorithmen¹⁰

Die gesamte Sicherheit des Systems ist auf einem kryptografischen Algorithmus begründet. Die Nachrichten, die zwischen den Komponenten ausgetauscht werden, sind alle mit

¹⁰ Ein Beispiel für eine mögliche Schlüsselhierarchie mit Schlüsselableitung findet sich in Abschnitt 4.8.6 „Beispiel für Schlüsselmanagement“.

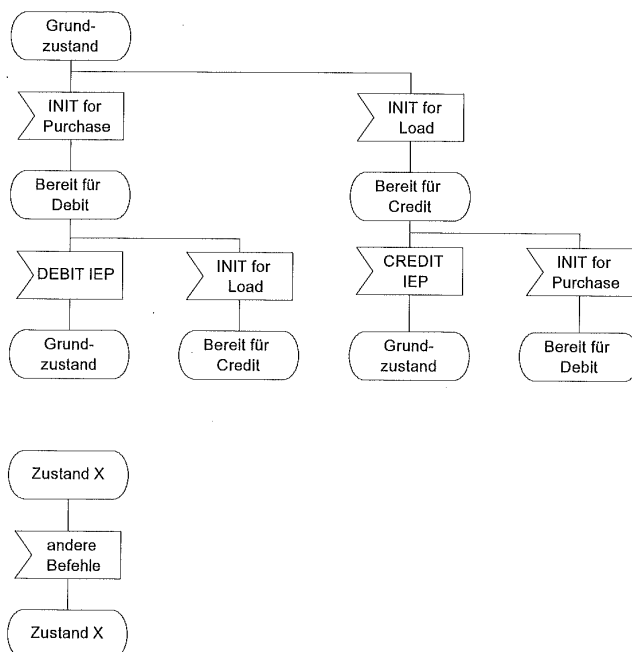


Bild 12.9 Vereinfachtes Zustandsdiagramm für Laden und Bezahlen mit der elektronischen Geldbörse nach EN 1546.

einer nachgestellten Signatur versehen, sodass Manipulationen zu erkennen sind. Für die immer im Klartext ausgetauschten Nachrichten ist dies der einzige Schutz.¹¹ Der Nachrichtenaustausch ist auch so beschaffen, dass jeder kryptografische Algorithmus zur Erzeugung der Signaturen benutzt werden kann. Zurzeit verwendet man meist den symmetrischen DES-Algorithmus, doch sieht die Norm auch den Einsatz asymmetrischer Algorithmen wie RSA oder DSS vor. Diese Unabhängigkeit ist ein großer Vorteil, da damit die Lebenserwartung und Flexibilität dieser Norm ganz wesentlich gesteigert wird.

Abläufe

Die Norm legt jedoch nicht nur Dateien, Kommandos und Zustände fest, sondern beschreibt und erklärt auch die dazugehörigen Abläufe. Diese sind in einer BASIC ähnlichen Pseudosprache auf der Grundlage der Datenelemente detailliert festgelegt. Dies war notwendig, da die Sicherheit mancher Vorgänge in erheblichem Maß von der Reihenfolge der internen Kommandoabarbeitung abhängt. Beispielsweise muss bei Transaktionen zuerst immer die jeweilige Protokolldatei auf den neuesten Stand gebracht werden, bevor das Terminal eine Antwort erhält. Im Übrigen sind die Abläufe und Transaktionen für alle

¹¹ Im Falle des DES-Algorithmus versteht man die Nachrichten am Ende mit einem 4 Byte langen MAC als Signatur.

Komponenten genau festgelegt. Für die Chipkarte mit der Börsenanwendung gibt es Beschreibungen für die folgenden Abläufe:

- Aufladen
- Bezahlen
- Storno einer Bezahlung
- Fehlerbehebung
- Währungsumrechnung
- Ändern der Börsenparameter

Zusätzlich können für Überwachungszwecke natürlich auch noch die Lesekommandos für Dateien benutzt werden. Die dazu notwendigen Aktionen können aber je nach Börsenanbieter und Überwachungsziel variieren.

Ein wesentliches Paradigma bestimmt den Aufbau aller festgelegten Abläufe. Es darf in keinem Fall möglich sein, durch Manipulation der Datenübertragung zwischen den Komponenten oder den Abbruch von Transaktionen elektronisches Geld zu erzeugen. Das ungünstigste Ergebnis solcher Eingriffe ist die Vernichtung von elektronischem Geld. Dies unterbindet gewisse Angriffe bereits im Vorfeld. Im Übrigen fließt dieses Paradigma bei beinahe allen elektronischen Geldbörsensystemen in das Systemdesign ein.

Alle Abläufe sind grundsätzlich in drei Phasen aufgeteilt: In Phase 1 findet die grundsätzliche Initialisierung der beteiligten Komponenten statt. Phase 2 dient für die eigentliche Funktionsausführung. Die daran anschließende Phase 3 ist optional und dient zur Bestätigung der vorangegangenen Abläufe. Der erfolgreiche Durchlauf von Phase 1 und 2 beinhaltet die einseitige oder optional gegenseitige Authentisierung beider Komponenten.

Bei allen Abläufen sind die eigentliche Börsenfunktion (Bezahlen, Aufladen, ...) und die ein- oder gegenseitige Authentisierung miteinander verschachtelt. Der Grund liegt in der

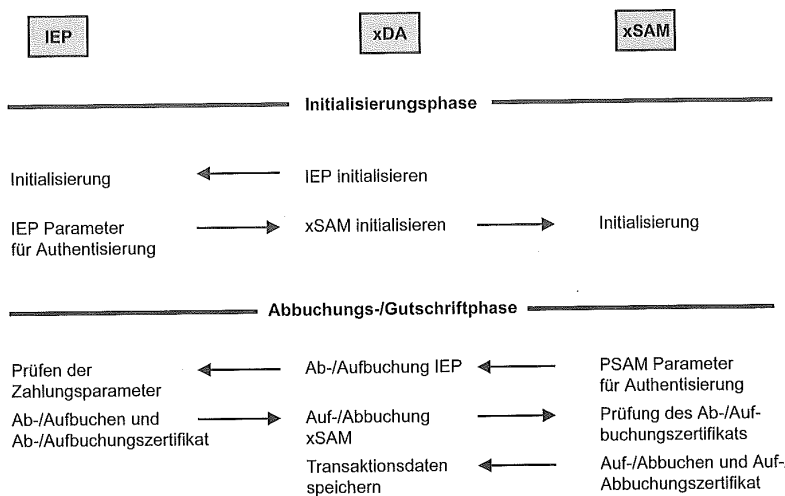


Bild 12.10 Der grundsätzliche Ablauf einer Transaktion bei der elektronischen Geldbörse nach EN 1546 (Phase 1 und Phase 2).

Minimierung des Zeitbedarfs für die Börsentransaktion und der Erhöhung der Sicherheit, da wesentlich weniger Kommandos notwendig sind. Vergleichbar ist dies mit den nach ISO genormten Kommandos INTERNAL AUTHENTICATE und EXTERNAL AUTHENTICATE, die durch das noch nicht genormte Kommando MUTUAL AUTHENTICATE ohne Einbuße an Funktionalität und Sicherheit ersetzt werden können.

Tabelle 12.5 Abkürzungen der Funktionen und Prozesse für die Darstellung von Zahlungstransaktionen nach EN 1546

Abkürzung	Bedeutung
Ax	Eindeutige Bezeichnung für eine Bearbeitung (<i>action</i>)
Cx	Eindeutige Bezeichnung für ein Kommando (<i>command</i>)
Parameters (...)	Anfrage an einen Teilnehmer mit den angegebenen Datenelementen
Response (...)	Antwort auf eine vorhergehende Anfrage mit den angegebenen Datenelementen
Rx	Eindeutige Bezeichnung für eine Antwort (<i>response</i>)
Sign (...)	Erstellung einer Unterschrift über die angegebenen Datenelemente
Verify (...)	Prüfung der angegebenen Datenelemente oder einer Funktion
Write (...)	Schreiben der angegebenen Datenelemente in eine Datei

Ablauf beim Aufladen

Um mit einer elektronischen Geldbörse bezahlen zu können, muss man diese zunächst aufladen. Dies wird im folgenden Ablauf gezeigt. Es handelt sich um die Variante, bei der die Geldbörse IEP mit einem Terminal direkt online von einem Hintergrundsystem mit dem dazugehörigen Sicherheitsmodul PPSAM aufgeladen wird. Es gäbe in der Norm auch noch weitere Varianten, so beispielsweise die Aufladung der Geldbörse über ein Sicherheitsmodul LSAM in einem Terminal. Der dargestellte Ablauf ist aber in heutigen Systemen üblich, da mit ihm die Kontrolle über das Aufladen vollständig beim Systembetreiber liegt.

Im Beispiel (Bild 12.11) wird eine elektronische Geldbörse IEP über ein Terminal LDA durch das Hintergrundsystem mit dem Sicherheitsmodul PPSAM aufgeladen. Dazu steckt zuerst der Kartenbenutzer seine Karte in das Terminal, welches als Erstes einen Reset ausführt. Im ATR teilt die IEP dann dem Terminal diverse Rahmenparameter zum nachfolgenden Kommunikationsablauf mit.

Anschließend selektiert das Terminal das DF für die elektronische Börse auf der IEP-Karte. Nachdem dies erfolgreich geschehen ist, wird vom Kartenbenutzer der zu ladende Betrag in der verfügbaren Währung im Terminal eingegeben. Diese Daten werden nun mit dem ersten Börsenkommando zum PPSAM gesendet. Das PPSAM prüft die angegebene Währungseinheit und den noch möglichen zu ladenden Betrag. Als Antwort sendet es drei Datenelemente zum Terminal zurück.

Das Terminal erweitert die vom PPSAM erhaltenen Daten durch den Ladebetrag M_{LDA} sowie die dazugehörige Währung $CURR_{LDA}$ und sendet sie mit dem Kommando INITIALIZE IEP for Load zur IEP-Karte. Diese prüft nun unter anderem, ob das durch den Ladebetrag resultierende neue Börsensaldo den in der Karte gespeicherten Maximalbetrag BAL_{maxIEP} für die Börse überschreiten würde. Ist dies nicht der Fall, erhöht die IEP einen Transaktionszähler NT_{IEP} , berechnet den Sitzungsschlüssel $KSES_{IEP}$ und errechnet eine Signatur S_I . Diese sendet es mit einigen weiteren Datenelementen zum Terminal zurück.

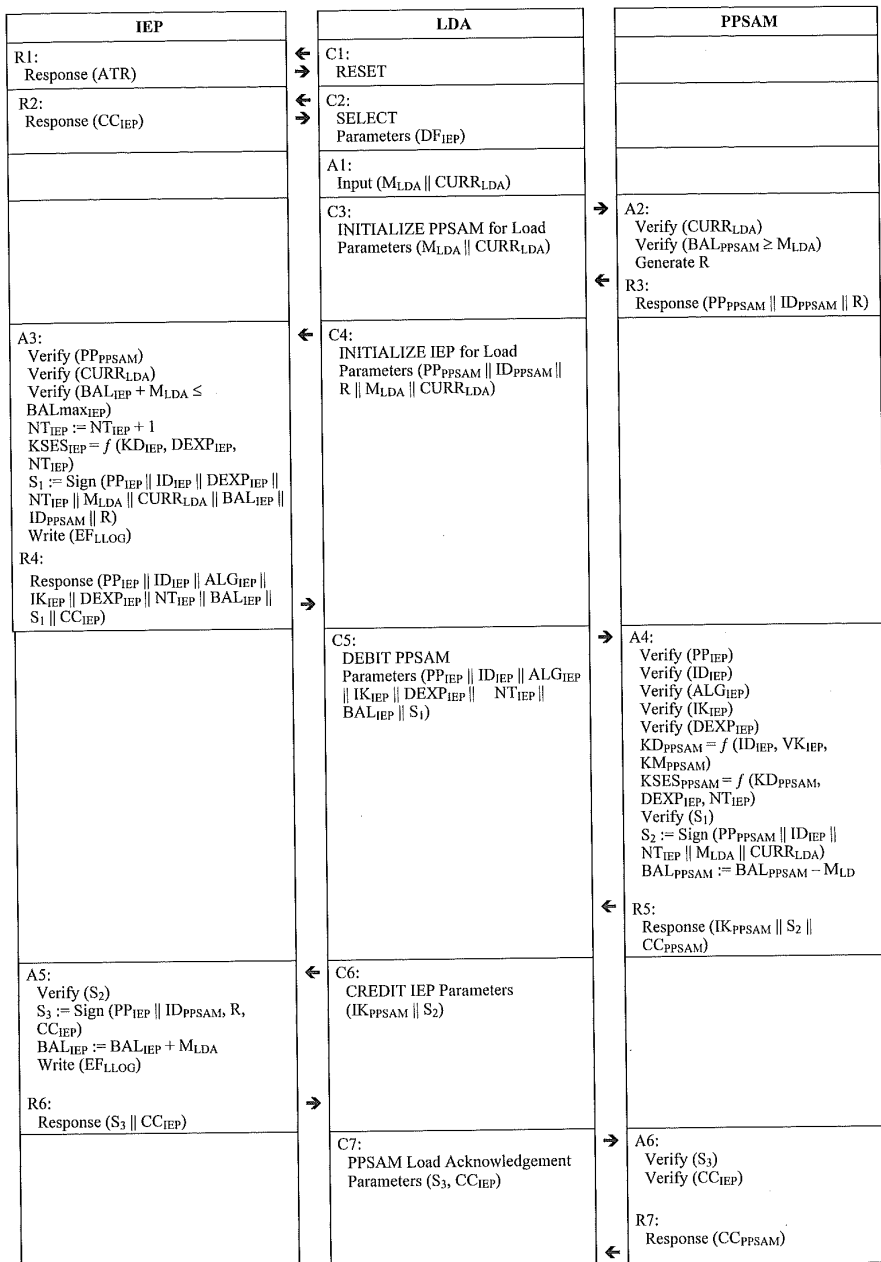


Bild 12.11 Transaktionsablauf beim Aufladen einer elektronischen Geldbörse IEP an einem Terminal LDA im Online-Fall mit einem Sicherheitsmodul PPSAM.

Die Aufgabe des Terminals nach diesem Kommando besteht nur darin, die erhaltenen Datenelemente an das PPSAM weiterzureichen. Dort werden nun die übergebenen Daten auf ihren erlaubten Wertebereich geprüft und sowohl der kartenindividuelle Schlüssel KD_{PPSAM} als auch der Sitzungsschlüssel $KSES_{PPSAM}$ erzeugt.

Verläuft die darauf folgende S_1 -Prüfung positiv, so ist die IEP-Karte authentisiert, da sie den geheimen Schlüssel für die S_1 -Berechnung kennen muss. Das PPSAM erstellt nun eine Signatur S_2 und sendet diese mit der Schlüsselinformation IK_{PPSAM} zum Terminal. Dort findet wiederum nur eine Durchreichung der Daten zur IEP-Karte statt. Diesmal mit dem Kommando CREDIT IEP.

Die Karte mit der elektronischen Geldbörse prüft nun S_2 . Konnte dies positiv abgeschlossen werden, ist nun auch das PPSAM durch die IEP authentisiert. Daraufhin wird das Saldo BAL_{IEP} in der elektronischen Geldbörse erhöht. Anschließend erstellt die IEP die dritte Signatur S_3 , welche zur anschließenden Bestätigung der erfolgreichen Erhöhung des Saldos in der IEP zum Terminal gesendet wird. Im abschließenden Kommando an das PPSAM wird diese Signatur übergeben, woraufhin die gesamte Aufladetransaktion abgeschlossen ist.

Der beschriebene Ablauf ist eine von mehreren Varianten. Er wird in der Praxis häufig angewandt, da es durchaus üblich ist, Aufladetransaktionen auf eine elektronische Geldbörse online durchzuführen. Es gibt auch Varianten in der EN 1546, die ein Aufladen an einem speziellen Ladesicherheitsmodul, dem LSAM, beschreiben. Dieses kann dezentral in speziellen Ladeterminals, wie z. B. in Geldausgabeautomaten, eingebaut sein.

Ablauf beim Bezahlen

Im folgenden Beispiel wird das Bezahlen mit den dazu notwendigen Komponenten elektronische Geldbörse IEP, Terminal PDA und Sicherheitsmodul im Terminal PSAM gezeigt. Nachdem die Börsenkarte in das Terminal gesteckt wurde, fordert dieses über einen Reset sowohl vom PSAM als auch von der IEP einen ATR an. Stimmt der ATR mit dem erwarteten Wert nicht überein, bricht das Terminal den Bezahlvorgang ab. Entspricht der ATR dem erwarteten Wert, wird vom Terminal das Börsen-DF in der IEP selektiert. Kann diese Dateiauswahl nicht durchgeführt werden, wird der Vorgang abgebrochen. Diese Vorgänge und die allgemeine Fehlerbehandlung sind aber aus Gründen der Übersichtlichkeit grundsätzlich nicht eingezeichnet.

Nach Auswahl des Börsen-DFs in der IEP-Karte sendet das Terminal das Initialisierungskommando für das Bezahlen INITIALIZE IEP for Purchase. Die IEP-Karte empfängt dieses Kommando, erhöht den Transaktionszähler und erstellt schließlich eine Signatur S_1 über verschiedene Datenelemente. Anschließend sendet sie diese und die Signatur zum Terminal.

Das Terminal sendet nun das Initialisierungskommando für das Bezahlen INITIALIZE PSAM for Purchase zum PSAM. Dabei werden lediglich alle Daten, die im vorherigen Kommando von der IEP-Karte empfangen wurden, zum PSAM weitergereicht. Das PSAM prüft nun diese Daten, d. h. es werden Ablaufdatum $DEXP_{IEP}$, Währung $CURR_{IEP}$, verwendeter kryptografischer Algorithmus ALG_{IEP} und andere übergebene Datenelemente mit im PSAM gespeicherten Werten verglichen. Fallen alle diese Vergleiche positiv aus, wird der Transaktionszähler NT_{IEP} erhöht. Würde nur ein Vergleich negativ ausfallen – z. B. wenn

das Verfallsdatum der IEP erreicht wäre –, würde die weitere Kommandoabarbeitung sofort abgebrochen und zum Terminal PDA ein entsprechender Returncode gesandt.

Das PSAM führt nun mit den von der IEP übergebenen Datenelementen eine Schlüsselableitung sowie die Sitzungsschlüsselerzeugung durch und prüft dann die Signatur S_1 . Bei Korrektheit von S_1 sind alle übergebenen Datenelemente authentisch. Zugleich ist die IEP durch das PSAM authentisiert. Das PSAM weiß also, dass die Karte mit der elektronischen Geldbörse eine echte und keine gefälschte Karte ist.

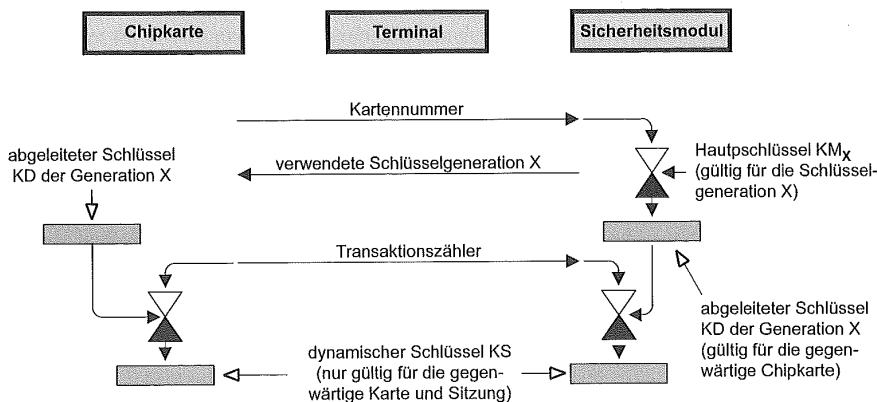


Bild 12.12 Eine mögliche Schlüsselableitung für eine elektronische Geldbörse nach EN 1546. Der Schlüssel ist abhängig von einer an die Chipkarte zu übergebenden Generation eines kartenindividuellen Schlüssels und von einem sitzungsindividuellen Transaktionszähler. Der erzeugte Schlüssel kann beispielsweise für das Bezahlen oder das Aufbuchen von Wertseinheiten benutzt werden.

Im nächsten Schritt erstellt das PSAM ebenfalls eine Signatur S_2 , die zusammen mit einigen weiteren Datenelementen zum Terminal gesandt wird. Am Terminal wird nun der zu bezahlende Betrag M_{PDA} sowie die dazugehörige Währung $CURR_{PDA}$ eingegeben. Als Nächstes sendet das Terminal den eingegebenen Betrag M_{PDA} und die vorherigen vom PSAM erhaltenen Daten mit dem Kommando DEBIT IEP zur Chipkarte mit der elektronischen Geldbörse. Diese prüft nun, ob überhaupt noch genügend Geld auf der Chipkarte für die Bezahlung vorhanden ist. Ist dies der Fall, wird die übergebene Signatur S_2 geprüft. Ist sie korrekt, dann sind die übergebenen Daten während der Übertragung nicht verfälscht worden, und auch das PSAM ist durch die IEP authentisiert. Das PSAM ist deshalb authentisiert, weil nur ein echtes PSAM über den geheimen Schlüssel zur Erstellung der Signatur S_2 verfügen kann. Nun wird der entsprechende Betrag vom Börsensaldo subtrahiert, eine Protokolldatei aktualisiert und eine Signatur S_3 zur Bestätigung der gerade getätigten Abbuchung erstellt.

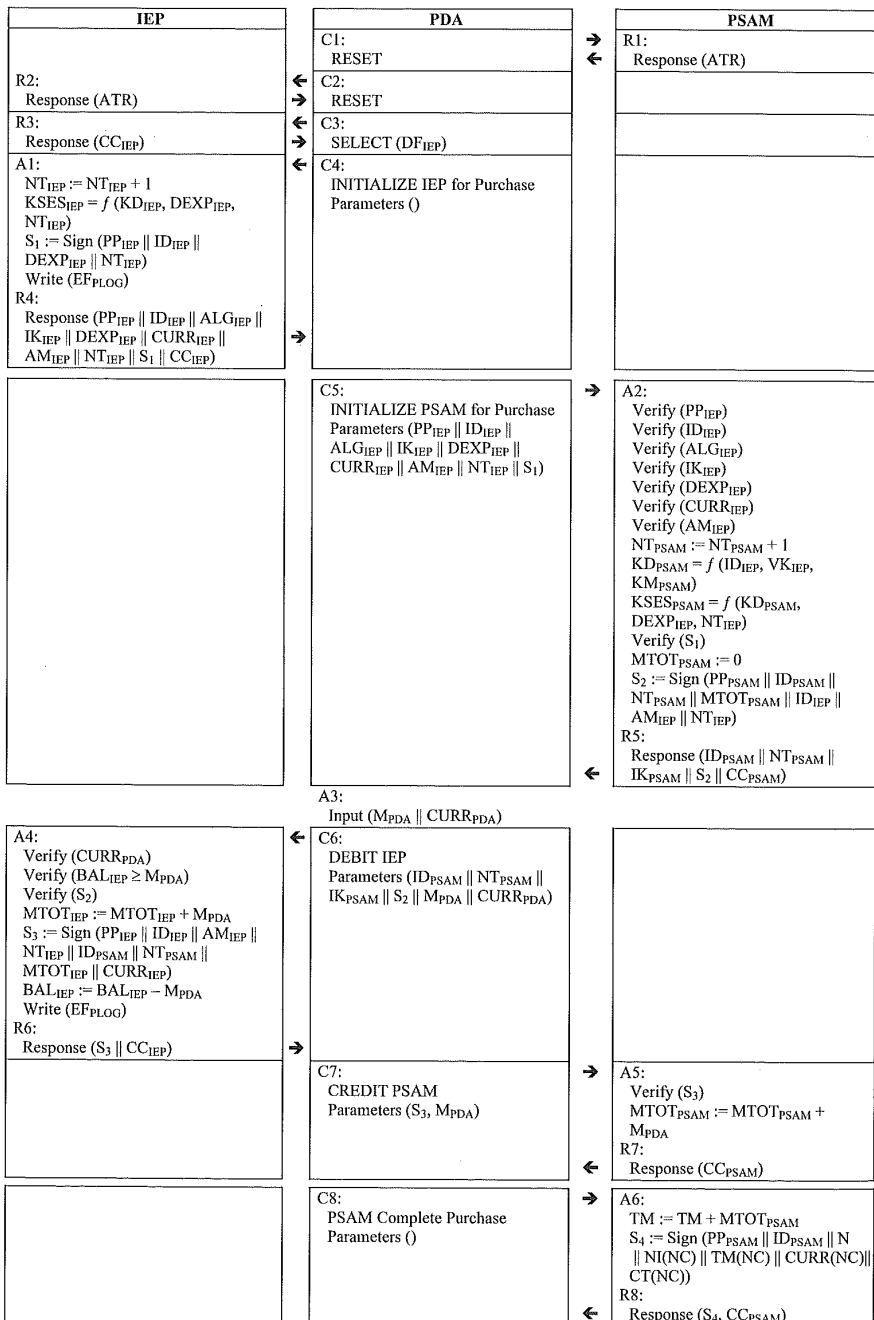


Bild 12.13 Transaktionsablauf beim Bezahlen mit der elektronischen Geldbörse IEP an einem Terminal PDA und einem Sicherheitsmodul PSAM nach EN 1546.

Die Signatur S_3 sowie der abgebuchte Betrag gelangen nun durch das Terminal zum PSAM, welches S_3 prüft. Ist diese Signatur korrekt, wird der in der IEP abgebuchte Betrag auf ein internes Datenelement $MTOT_{PSAM}$ addiert. Ein weiteres Kommando „PSAM Complete Purchase“ bringt nun das Saldo des PSAM auf den aktuellen Stand, indem $MTOT_{PSAM}$ auf das Börsensaldo TM addiert wird. Anschließend erhält das PSAM noch eine Signatur S_4 zur Bestätigung der korrekt verlaufenen Zahlung.

Der dargestellte Ablauf ist eine sehr einfache Variante der diversen in der EN 1546 Norm beschriebenen Bezahlungsabläufe. Es existieren parallel dazu noch zusätzliche Variationen mit speziellem schnelleren Abbuchen für Kartentelefone sowie auch eine Variante mit einer zusätzlichen Quittungsmöglichkeit am Ende der Transaktion.

Die hier für die Chipkarte beschriebenen Dateien, Kommandos und Abläufe sind auch für alle wichtigen Komponenten des Systems festgelegt. Dies betrifft vor allem die Sicherheitsmodule, da die Systemsicherheit nur von ihnen abhängt. Um das Gesamtsystem zu überwachen, das bei größeren Anwendungen aus Zehntausenden von Terminals und mehreren hunderttausend Chipkarten bestehen kann, können statistische Methoden benutzt werden. Für jede einzelne Chipkarte immer eine komplette Buchführung zu erstellen widerspräche der geforderten Anonymität und wäre im Übrigen viel zu rechenintensiv. Mit einer nur auf Stichproben basierenden Überwachung der Zahlungsströme kann jedoch, wie Versuche beweisen, die Sicherheit des Gesamtsystems laufend mit einem vertretbarem Aufwand geprüft werden.

Mit der europäischen Norm EN 1546 wurde ein Grundstein für branchenübergreifende elektronische Börsensysteme mit Chipkarten gelegt. Es wurden fast alle zurzeit üblichen und für sinnvoll erachteten Verfahren und Funktionen in die Norm integriert. Lediglich eine Funktion wurde bislang nicht beschrieben, die aber seitens der Kartenbenutzer von großer Bedeutung ist. Es ist die so genannte Purse-to-Purse-Transaktion, also die Übergabe von elektronischem Geld von einer Geldbörse zu einer anderen. Zurzeit gibt es im Rahmen von EN 1546 keine Beschreibung eines solchen Geldtransfers.

12.3.2 CEPS (*common electronic purse specifications*)

Mitte der 90er-Jahre entstanden in vielen europäischen Staaten unabhängig voneinander viele elektronische Geldbörsen auf der Basis von Chipkarten. Typische Beispiele dafür sind Quick in Österreich, Geldkarte in Deutschland, Proton in Belgien und in den Niederlanden. Alle diese Börsensysteme haben eine ähnliche Funktionalität, doch sind sie alle zueinander inkompatibel. Die Notwendigkeit, diese Börsensysteme zueinander kompatibel zu machen, wurde unter anderem mit dem Hintergrund der gemeinsamen europäischen Währung ab 2002 immer unerlässlicher. Da alle diese Börsenkarten in der Regel eine Gültigkeitsdauer von drei Jahren haben, ist es grundsätzlich möglich, über einen festzulegenden Migrationspfad schleichende Veränderungen an den Börsensystemen über einige Jahre hinweg vorzunehmen.

Die fundamentale Voraussetzung für mehrere zueinander kompatible elektronische Börsensysteme ist jedoch ein Dokument, das die kompatibilitätskritischen Merkmale für alle Börsensysteme spezifiziert. Dieses Dokument trägt den Namen CEPS (*common electronic purse specifications*) und wurde 1999 in seiner ersten Version von CEPSCO [CEPSCO] veröffentlicht. Der Fokus von CEPS wurde jedoch bereits in einer frühen Spezifikations-

phase auf ein weltweit interoperables elektronisches Börsensystem gesetzt und nicht nur auf europäische Belange begrenzt.

CEPS hat die für moderne elektronische Geldbörsensysteme üblichen Funktionen wie offline Bezahlen, online Aufladen, online Entladen und online Wechsel von Währungen. Die Grundlage von CEPS stellt die europäische Norm für elektronischen Geldbörsen EN 1546 dar¹², wobei jedoch einige Erweiterungen und Änderungen vorgenommen wurden. Gegenüber EN 1546 werden beispielsweise zur Authentisierung von Terminal und Chipkarte RSA-basierte hierarchische Zertifikate verwendet. Als symmetrischer Kryptoalgorithmus wird der Triple-DES empfohlen.

CEPS ist wie viele elektronische Geldbörsensysteme für einfache Chipkarten-Mikrocontroller optimiert. Eine typische Realisierung von CEPS in Assembler oder hardwarenahem C benötigt 8 kByte ROM, 4 kByte EEPROM, 1 kByte RAM und einen numerischen Prozessor für den asymmetrischen Kryptoalgorithmus. Die meisten europäischen Börsensysteme werden in Zukunft kompatibel mit CEPS sein und somit sollte es mittelfristig auch möglich sein, in Europa grenzüberschreitend mit einer Börsenkarte zu bezahlen.

12.3.3 Proton

Proton ist eine weltweit verbreitete elektronische Geldbörse, die bis heute nahezu ausschließlich von der Firma Bull entwickelt wird und deren Anfänge bis 1995 zurückreichen. Ihr Ursprung liegt in den Ländern Belgien und Niederlande (dort unter dem Markennamen „Chipknip“ bekannt), in denen sie bis heute auch die größte Verbreitung und mittlerweile den Status eines nationalen elektronischen Geldbörsensystems hat. Größere Börsensysteme auf der Basis von Proton gibt es auch in der Schweiz und in Schweden (Markenname „Cash“). Mit Stand Frühjahr 2002 gab es weltweit ca. 40 Millionen ausgegebene Karten und etwa 360 000 Terminals.

Die ursprüngliche Bezeichnung für die elektronische Geldbörse war CC 60, ein Name, der von der Firma Bull stammt. Die aktuelle Version der Börse ist mit R3 betitelt und die bereits in ersten Ausführungen verfügbare und stark erweiterte Generation hat die Bezeichnung R4.

R3 ist eine auf kostengünstige Chipkarten-Mikrocontroller optimierte Geldbörse, die durchaus auch auf Chips mit 16 kByte ROM, 2 kByte EEPROM und 256 Byte RAM realisiert werden kann. R4 ist eine erweiterte Version von R3 und auch kompatibel zu CEPS¹³. Die Spezifikationen von R4 definieren neben der eigentlichen Börse auch ein Betriebssystem für Multiapplication-Chipkarten, das neben der elektronischen Geldbörse auch Kredit- und Debitfunktionalität nach EMV,¹⁴ eine digitale Signaturanwendung und Java Card Funktionalität beinhaltet. Des Weiteren existiert auch eine kontaktlose Variante von Proton, die vor allem für den ÖPNV-Sektor gedacht ist. Herausgeber der vertraulichen Spezifikationen ist Proton World [Proton].

¹² Siehe auch Abschnitt 12.3.1 „CEN-Norm EN 1546“.

¹³ Siehe auch Abschnitt 12.3.2 „CEPS“.

¹⁴ Siehe auch Abschnitt 12.4 „EMV-Anwendung“.

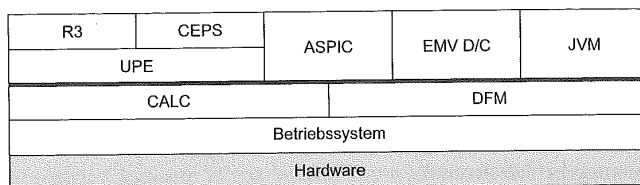


Bild 12.14 Schematische Darstellung der grundlegenden Architektur des Chipkarten-Betriebssystems Proton R4. Es wurden die folgenden Abkürzungen verwendet: CALC – *card application life cycle*, DFM – *data file management*, UPE – *universal purse engine*, ASPIC – *Application for Secure Personal Identification* = PKI-Funktionalität, EMV D/C – EMV Debit-/Kreditanwendung, JVM – *Java virtual machine*.

Die elektronische Börse beinhaltet die üblichen Funktionen wie Laden (*loading*), Zahlen (*single payment*), inkrementelles Zahlen (*sliced payment*) sowie Storno einer Bezahlung (*refund*). Die Hauptanwendung für inkrementelles Zahlen sind öffentliche Kartentelefone, bei denen kurz hintereinander innerhalb einer Sitzung Kleinbeträge vom Börsensaldo abgebucht werden müssen. Die Transaktionen werden in Record-orientierten Protokolldateien gespeichert. Auch die Parameter für die Börse sind in Dateien abgelegt. Zur Realisierung auf sehr kostengünstigen und somit speicherarmen Mikrocontrollern dürfen sich die Dateien sogar direkt unter dem MF ohne ein DF befinden. Als Kryptoalgorithmen kommen bei R3 sowohl DES und Triple-DES zum Einsatz. Die verwendeten Chipkarten-Kommandos sind vielfach an ISO/IEC 7816-4 und EN 1546 angelehnt bzw. dazu kompatibel und zusätzlich noch um einige anwendungsspezifische Kommandos ergänzt. Die eigentliche Börsenfunktion ist an vielen Stellen mit den üblichen EN 1546¹⁵ Abläufen verwandt. Man sieht jedoch deutlich, dass Proton einige Jahre älter als EN 1546 ist.

12.3.4 Das Mondex-System

Es gibt weltweit mittlerweile einige große Zahlungsverkehrssysteme, die als zentralen Bestandteil Chipkarten verwenden. Nur sehr wenige dieser Systeme benutzen dabei als Grundlage eine elektronische Geldbörse, d. h. die Geldeinheiten sind bei solch einer Architektur direkt in der Karte und nicht in einem Hintergrundsystem gespeichert. Nur ein System nimmt für sich in Anspruch, dass es einen elektronischen Zahlungsverkehr zulässt, der dem von normalem Bargeld entspricht. Es ist dies das Mondex-System [Mondex].

Die Idee für dieses zurzeit weltweit einmalige Systemkonzept wurde im Jahr 1990 geboren. Nach fünf Jahren Entwicklungszeit wurde im Juli 1995 ein erster Feldversuch in der südenglischen Stadt Swindon durchgeführt. Dabei waren vom Zeitungstand, Imbissbude über Supermarkt, Reisebüro bis zu Tankstellen und Telefonen die unterschiedlichsten Kategorien von Händlern abgedeckt. Als Maximalbetrag der Börse wurde für den Feldversuch 500 £ (≈ 550 €) festgelegt, ein Wert, der prinzipiell auf jede beliebige Höhe eingestellt werden kann. Seit diesem in der Presse ausführlich dargestellten Test hat es noch weitere Feldversuche in den unterschiedlichsten Regionen gegeben, wobei es aber bislang noch in keinem Staat zu einer landesweiten Einführung des Systems gekommen ist.

¹⁵ Siehe auch Abschnitt 12.3.1 „CEN-Norm EN 1546“.

Mondex war ein Konsortium aus drei Firmen: British Telekom, National Westminster Bank und Midland Bank. Der Zweck der Firma war es, ein Zahlungsmittel zu schaffen, das wie Bargeld benutzt werden kann, aber nicht die Nachteile von Bargeld hat. Das Ergebnis dieser technischen Entwicklung sollte dann in einem Franchise-Verfahren an Banken und Firmen verkauft werden. Nach einigen Zwischenständen gehört Mondex mittlerweile dem Kreditkartenherausgeber Mastercard.

Mondex ist eines der wenigen Systeme von elektronischen Geldbörsen, die als komplettes System von der Karte bis zum Hintergrundsystem angeboten werden. Die hohen ursprünglichen Erwartungen der Verbreitung haben sich jedoch trotz immenser Marketingaufwendungen nicht erfüllt. Mondex ist mittlerweile eine elektronische Börsenanwendung auf dem Multos Chipkarten-Betriebssystem¹⁶ und wird weltweit an einigen Orten verbreitet.

Das Chipkarten-Betriebssystem von Mondex ist nicht nur für elektronische Geldbörsen geeignet, sondern allgemein verwendbar, multifunktional und auch offen für mehrere Anwendungen auf einer Chipkarte. Es hat den Namen Multos und wird vor allem im Sektor des kartengestützten Zahlungsverkehrs von der Firma Maosco [Maosco] international vermarktet. Als Besonderheit unterstützt das Mondex Chipkarten-Betriebssystem im Feld nachladbaren Programmcode einer C ähnlichen Sprache namens MEL (*Multos executable language*), die in der Chipkarte von einem Interpreter abgearbeitet wird.

Das System

Da sich die Mondex-Geldbörse wie echtes Geld verhalten soll, ist es natürlich auch möglich, so genannte Purse-to-Purse-Transaktionen vorzunehmen. Damit können Kartenbesitzer auch untereinander zahlen, ohne dass eine Bank oder eine ähnliche Instanz dazwischen ist oder von dieser Buchung erfährt. Das System ist vollständig offen und anonym, und es können beliebig viele Parteien daran beteiligt sein. Die möglichen Geldströme und die beteiligten Instanzen verdeutlicht Bild 12.15.

Die elektronische Geldbörse befindet sich im Chip einer üblichen ID-1 Karte mit Kontakten. Um den Saldo der Börse einzusehen, gibt es einen Schlüsselanhänger in Streichholzschachtelgröße mit Display. Steckt man die Chipkarte in dieses Miniterminal, kann man

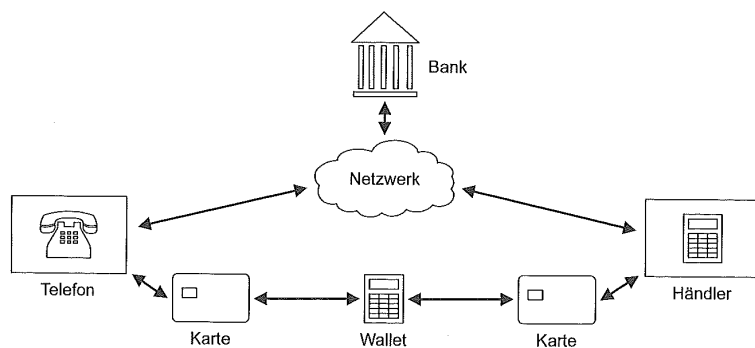


Bild 12.15 Mögliche Zahlungsströme und beteiligte Parteien im elektronischen Zahlungsverkehr des Mondex-Systems.

¹⁶ Siehe auch Abschnitt 5.14.2 „Multos“.

den aktuellen Börsensaldo und die letzten zehn Transaktionen ansehen. Möchte man einem anderen Kartenbesitzer Geld von seiner elektronischen Börse geben, benötigt man das sogenannte Wallet. Dies ist ein Taschenrechner-ähnliches Gerät mit kleiner Tastatur und Display. Auch besitzt es ein eingebautes Sicherheitsmodul und ein Terminal für die elektronische Geldbörsenkarte. Um nun eine Purse-to-Purse-Transaktion durchzuführen, steckt man als Erstes seine Chipkarte in das Wallet. Dann gibt man den zu transferierenden Betrag ein, der anschließend von der elektronischen Börse in das Sicherheitsmodul des Wallets übertragen wird. Anschließend wird die zweite Karte gesteckt und der Betrag vom Sicherheitsmodul in diese übertragen. Damit ist die gesamte Transaktion abgeschlossen.

Als weiteres Gerät zu diesem Zahlungsverkehrssystem gibt es ein Telefon mit eingebautem Terminal. Dies ermöglicht während des Telefonierens einen Geldtransfer über die Telefonleitung. Der typische Anwendungsfall ist beispielsweise die telefonische Bestellung von Artikeln aus einem Versandhauskatalog. Damit kann man zeitgleich mit der Bestellung auch bezahlen. Ebenso lässt diese Technik natürlich das Aufladen der elektronischen Börse über Telefon oder eine Transaktion zwischen zwei Karteninhabern zu. Beim Aufladen der Karte gegenüber einem Bankkonto muss natürlich aus Sicherheitsgründen eine 4-stellige PIN eingegeben werden, um so einen Schutz des Kontoinhabers gegenüber ungeberechtigten Abbuchungen zu realisieren.

Jede elektronische Geldbörse kann bis zu fünf unterschiedliche Währungen aufnehmen. Sobald der Börsensaldo einer Währung null erreicht hat, ist es möglich, dafür eine neue Währung in die Karte zu laden. Die Börse lässt sich mit einem einfachen Kommando sperren und nach Eingabe einer 4-stelligen PIN wieder entsperren, sodass keine unbefugte Benutzung möglich ist.

In den Terminals der Händler befinden sich als Sicherheitsmodule die gleiche Art von Chipkarten, die der Kunde hat. Es wäre also auch möglich, mit dem Sicherheitsmodul des Händlers wiederum andere Güter zu bezahlen. Interessanterweise könnte sich dann natürlich auch der Diebstahl einer solchen Karte rentieren, da sie wie eine ganz normale Börsenkarte gehandhabt werden kann. Allerdings hat man das Problem erkannt und hier einen Riegel vorgeschoben. Die Händlerkarten sind dahingehend konfigurierbar, dass sie nur den Empfang von elektronischem Geld zulassen. Ein Abbuchen ist nur im Zuge einer Online-Transaktion mit der Bank des Händlers möglich. Man sieht also, dass elektronisches Geld nicht zwangsläufig immun gegenüber Diebstahl ist. Dies hängt lediglich davon ab, ob es sich entsprechend weiterverwenden lässt. Sofern das Händlerterminal über die Möglichkeit einer (aufbaubaren) Online-Verbindung zur Bank verfügt, kann ab einem bestimmten Betrag auch ein automatisches Umladen von der Händlerkarte zum Konto des Händlers eingestellt werden.

Sicherheitsmechanismen und Zahlungsablauf

Alle Spezifikationen, die den Zahlungsablauf beschreiben, sowie das Sicherheitsmodell des Mondex-Systems sind vertraulich. Damit ist es sehr schwierig, genauere technische Details über das System und die einzelnen Komponenten zu erhalten. Deshalb kann im Folgenden nur ein grober technischer Überblick dargestellt werden, der nur einige der Mechanismen und Verfahren des Systems aufzeigt.

Der benutzte Mikrocontroller ist ein Hitachi H8/3102. Für die Serienproduktion ist ein eigens für Mondex entwickelter Mikrocontroller mit numerischer Recheneinheit und entsprechender Speichergröße geplant, da die Anwendung ca. 5 kByte Platz im EEPROM be-

nötigt. Es wird wahrscheinlich ein symmetrischer kryptografischer Algorithmus verwendet, wie z. B. der DES. Aufgrund der Tatsache, dass in Zukunft ein spezieller Mikrocontroller mit Recheneinheit eingesetzt wird, lässt sich vermuten, dass zur Erhöhung der Sicherheit auf einen asymmetrischen Algorithmus übergegangen wird. Es könnte sich dabei beispielsweise um den RSA-Algorithmus handeln. Prinzipiell ist das System aber unabhängig vom verwendeten Kryptoalgorithmus, da es nicht auf speziellen Eigenschaften von bestimmten Algorithmen beruht, sondern zur Sicherung der Datenübertragung nur (digitale) Signaturen verwendet. Es unterscheidet sich dabei wenig von der branchenübergreifenden europäischen Geldbörse nach EN 1546.¹⁷

Da das Mondex-System vollständig dezentral betrieben wird, muss ein besonderes Verfahren zum Wechsel von Schlüsselversionen und Algorithmen vorhanden sein. Jede ausgegebene Karte besitzt mindestens zwei völlig unterschiedliche kryptografische Algorithmen mit mehreren dazugehörigen Schlüsseln. Soll auf eine neue Schlüsselversion weitergeschaltet werden oder es ist gar notwendig, einen anderen kryptografischen Algorithmus zu verwenden, dann wird in allen Chipkarten, die eine Online-Verbindung zum Hintergrundsystem aufbauen, ein entsprechender Parameter gesetzt. Die Karten haben aber wiederum die Fähigkeit, auf allen Karten, mit denen sie eine Zahlungstransaktion durchführen, ebenfalls die Parameter für die neue Schlüsselversion oder den neuen Algorithmus zu setzen. Dieses Schneeballsystem führt aufgrund der exponentiell wachsenden Informationsweitergabe innerhalb kürzester Zeit zu einem systemweiten Umschalten auf die neuen Rahmenparameter. Dies wäre selbst dann der Fall, wenn man seitens des Hintergrundsystems nur auf einer einzigen Karte den Parameter ändern würde. Dies ist eine sehr effektive, schnelle und einfache Methode, um globale Daten in einem dezentral arbeitenden Zahlungsverkehrssystem zu ändern.

Natürlich muss es auch eine Möglichkeit geben, um bestimmte Chipkarten im Zahlungsverkehrssystem zu isolieren. Dazu gibt es drei verschiedene Vorgehensweisen. Über eine Sperrliste können verdächtige Chipkarten an Automaten erkannt und daraufhin eingezogen werden – was aber meist nur an Geldausgabeautomaten möglich ist, da nur sie über die technischen Voraussetzungen verfügen, Karten einzubehalten. Die Sperrlisten werden in alle Terminals geladen, die die Fähigkeit haben, Chipkarten zu sperren, sodass sie nicht weiter für Transaktionen genutzt werden können. Alle ausgegebenen elektronischen Geldbörsen lassen grundsätzlich nur eine bestimmte Anzahl von Transaktionen zu, danach sind sie automatisch gesperrt. Diese Sperre kann durch eine Online-Anfrage nach Überprüfung der Sperrlisten wieder aufgehoben werden, sodass die Chipkarte nicht ausgetauscht werden muss. Damit wird sichergestellt, dass eine Chipkarte mit elektronischer Geldbörse nicht unbegrenzt lange ohne Kontrolle durch das Hintergrundsystem betrieben werden kann.

Eine typische Zahlungstransaktion zwischen zwei Chipkarten im Mondex-System gliedert sich in zwei Phasen, welche in Bild 12.16 grafisch dargestellt sind. In der ersten Phase wird eine Registrierung der aktuellen Transaktion vorgenommen. Dabei werden alle für den darauf folgenden Geldtransfer notwendigen Daten ausgetauscht. In der anschließenden zweiten Phase sendet die Chipkarte 2 den gewünschten Betrag zur Chipkarte 1. Der komplette Datensatz ist mit einer digitalen Signatur unterschrieben, sodass er während der Datenübertragung nicht manipuliert werden kann. Nach Erhalt prüft die Chipkarte 1 die Sig-

¹⁷ Siehe auch Abschnitt 12.3.1 „CEN-Norm EN 1546“.

natur und kann damit zusätzlich zur authentischen Datenübertragung die Authentizität von Chipkarte 2 prüfen. Laufen alle Prüfungen zufrieden stellend ab, wird in Chipkarte 1 der gewünschte Betrag abgebucht und mit einer digitalen Signatur versehen zur Chipkarte 2 gesendet. Diese prüft nun ihrerseits die digitale Signatur und schließt dadurch die Manipulation der Datenübertragung aus. Gleichzeitig authentisiert sich dadurch Chipkarte 1. Laufen alle Prüfungen erfolgreich ab, wird der übertragene Betrag abgebucht. Anschließend erstellt Chipkarte 2 eine Bestätigung der korrekten Abbuchung, versieht sie ebenfalls mit einer digitalen Signatur und sendet sie zur Chipkarte 1. Mit dem Empfang dieser Zahlungsbestätigung und einer erfolgreichen Prüfung ist die Transaktion abgeschlossen.

Beide Karten enthalten Protokolldateien und haben entsprechende Verfahren, damit bei einer abgebrochenen Transaktion wieder korrekt an der entsprechenden Stelle aufgesetzt werden kann. Diese Fehlerbehebungsverfahren sind von großer Bedeutung, da es sonst vorkommen könnte, dass elektronisches Geld durch eine abgebrochene Transaktion vernichtet wird. Beide beteiligten Karten haben jeweils drei verschiedene Protokolldateien, in denen transaktionsrelevante Daten gespeichert werden. In der Transaktionsprotokolldatei (*transaction log*) sind diverse Daten über die letzten zehn erfolgreichen Zahlungstransaktionen gespeichert. Die Protokolldatei für schwebende Transaktionen (*pending log*) enthält alle Daten, die während einer Transaktion anfallen und für ein eventuelles Fehlerbehebungsverfahren notwendig sind. Die dritte Protokolldatei (*exception log*) beinhaltet alle nicht erfolgreich abgeschlossenen Transaktionen. Sind alle Records in dieser Datei beschrieben, wird die Chipkarte automatisch gesperrt. Der Karteninhaber muss dann seine Karte durch eine Online-Transaktion wieder entsperren lassen. Dabei werden die Dateieinträge in das Hintergrundsystem geladen, analysiert und daraufhin die Einträge in der Protokolldatei gelöscht.

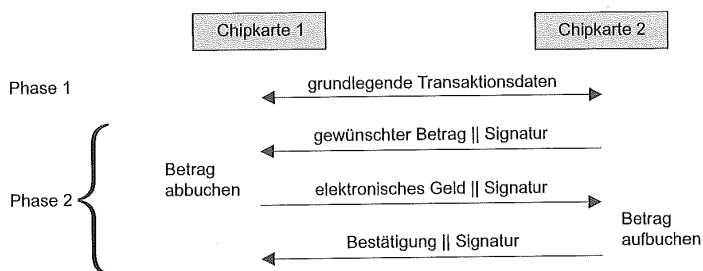


Bild 12.16 Informationsfluss bei einer Transaktion zwischen zwei Chipkarten im Mondex-System.

Rekapitulation

Das Mondex-System ist zurzeit das einzige vollständig offene elektronische Zahlungsverkehrssystem mit einer elektronischen Geldbörse. Es lässt alle Transaktionen zu, die auch mit normalem Bargeld möglich sind. Darüber hinaus können auch noch Zahlungstransaktionen über diverse Telekommunikationsmedien, wie z. B. Telefon, abgewickelt werden. Falls die Chipkarte mit der elektronischen Börse verloren geht, ist bei diesem System natürlich auch das in der Börse befindliche Geld verloren, ganz wie bei einer echten Geldbörse mit Bargeld. Allerdings ist damit das System vollständig anonym, was die Benutzerakzeptanz sicherlich noch weiter erhöht. Das Mondex-System ist gewissermaßen die Simulation

eines echten Geldkreislaufs. Da bei vielen Zentralbanken und Behörden starke Vorbehalte gegenüber direkten Geldtransfer von Karte zu Karte bestehen, wurde auch eine Variante von Mondex geschaffen, bei dem die Purso-to-Purse Transaktion gesperrt ist. Damit besteht dann der für elektronische Geldbörsensysteme übliche Geldkreislauf analog EN 1546.

Da es bei einem solchen System aufgrund des prinzipiellen Aufbaus natürlich nicht möglich ist, für jede einzelne Transaktion eine Gebühr zu verlangen, kommt sehr schnell die Frage auf, wie sich der Systembetreiber finanziert. Die notwendigen Investitionen für Aufbau und Betrieb sind nicht gerade gering. Im Feldversuch in Swindon wurde von jedem Kartenbenutzer eine nicht allzu hohe monatliche Gebühr (1,5 £ $\approx$ 1,70 €) für seine elektronische Geldbörse erhoben. Das Gleiche trifft natürlich auch für die Händler zu. Gebühren für das Clearing der Händlerumsätze können zwar erhoben werden, doch steht es aufgrund des vollständig offenen Systems den Händlern natürlich frei, mit dem erhaltenen elektronischen Geld ihrerseits wieder bei anderen Händlern einzukaufen. Der Systembetreiber könnte sich aber auch durch verschiedene Dienstleistungen an den Börseninhabern ebenso wie an den Händlern finanzieren. Bei Mondex genießt er den großen Vorteil, dass die Clearingkosten beinahe null sind, da ein Clearing im ursprünglichen Sinn nicht mehr notwendig ist. Gerade bei Zahlungen im Kleinstbereich (*micropayments*) können die Clearingkosten im Verhältnis zum getätigten Umsatz sehr groß werden. So bewegen sich bei einigen elektronischen Geldbörsensystemen die Kosten für das vollständige Clearing inklusive Transaktionsaufzeichnung über Schattenkonten von einer einzigen Transaktion im Bereich von 2,5 Euro Cent. Somit betragen die Clearingkosten für den Kauf eines Kaugummis am Automaten (10 Euro Cent) immerhin 25 Prozent vom Kaufpreis. Dies ist für einen Händler völlig unakzeptabel. Im Mondex-System würden diese Kosten jedoch nicht entstehen.

Das Mondex-System wird in den nächsten Jahren den Markt für elektronischen Zahlungsverkehr in vielen Punkten beeinflussen. Es gibt auf internationaler Ebene mehrere große Banken, die über eine Einführung eines solchen Zahlungsverkehrssystems nachdenken. Die von vielen Nationalbanken als sicherheitskritisch angesehene Möglichkeit von Purso-to-Purse-Transaktionen kann in den neuen Versionen abgeschaltet werden, wodurch die Akzeptanz stark gestiegen ist. Man darf gespannt die weitere Entwicklung abwarten.

12.4 EMV-Anwendung

Spezifikationen und Normen für Chipkarten existieren schon seit vielen Jahren für die unterschiedlichsten Anwendungsgebiete. Allerdings war bisher traditionell eine starke Konzentration auf den Bereich der Telekommunikation, also Telefon- und GSM-Karten, vorhanden. Seit Mitte der 90er-Jahre hat sich diese Situation aber sehr geändert. Die europäische Geldbörsennorm EN 1546 ist dafür ein gutes Beispiel. Der wichtigste Vertreter auf dem Gebiet des Zahlungsverkehrs ist aber die nach ihren drei Initiatoren Europay, Mastercard und Visa benannte EMV-Spezifikation, die Kreditkarten mit einem Mikroprozessorchip detailliert und in allen Aspekten beschreibt. Mittlerweile existiert auch eine entsprechende Spezifikation für den dazu passenden Terminaltyp.

Im Herbst 1993 begannen die drei international tätigen Kreditkartenfirmen Europay, Mastercard und Visa mit den ersten Arbeiten an der Spezifikation mit dem Namen „IC Card Specifications for Payment Systems“. Innerhalb verhältnismäßig kurzer Zeit stand eine erste

Ausgabe zur Verfügung, die im Oktober 1994 als Version 1 veröffentlicht wurde. Mitte des Jahres 1995 wurde dann die revidierte Spezifikation, nunmehr mit der Versionsnummer 2, fertig gestellt. Ende Juli 1996 wurde die endgültige Spezifikation namens EMV '96 freigegeben, die mit der Version 2 abwärts kompatibel ist. Nach Bereinigung von Unklarheiten und kleineren Fehlern wurde dann im Sommer 1998 eine vollständig überarbeitete und kompatible EMV-Version mit der Nummer 3 veröffentlicht. Ende 2000 wurde die Versionsnummer mit einigen Änderungen auf den Stand 4 gehoben, welche die derzeit gültige EMV-Spezifikation mit dem offiziellen Namen EMV 2000 ist. Sie ist auch via Internet erhältlich [EMV], und man kann jedem Interessierten uneingeschränkt empfehlen, sie zu studieren.

Die Motivation für die Herausgeber von Kreditkarten, eine Spezifikation für Kreditkarten mit Chip in so kurzer Zeit zu erstellen, ist vielschichtig. Zum einen können bisherige Kreditkarten mit Magnetstreifen sehr leicht gefälscht werden. Das einzige Hindernis stellt mittlerweile nur noch das noch einigermaßen fälschungssichere Hologramm dar. Alle anderen Kartenelemente lassen sich mit relativ geringem Aufwand duplizieren. Der zweite nennenswerte Stichpunkt sind zusätzliche Funktionen, die eine Mikroprozessorkarte bietet. Elektronische Geldbörse, Bonuspunkte und Telefonfunktionen seien hier nur als Beispiele genannt.

Die Spezifikation für die Kreditkarte mit Chip gliedert sich in vier Teile, Bücher genannt: Buch 1 (*book 1 – application independent ICC to terminal interface requirements*) lehnt sich stark an die Normenreihe ISO/IEC 7816-1/2/3 an und beschreibt die elektromechanischen Eigenschaften, die logische Schnittstelle und die Übertragungsprotokolle, also alle anwendungsunabhängigen Parameter. Die Chipkarte hat nach diesem Teil der Spezifikation das ID-1 Format¹⁸ mit ISO-Kontaktposition. Sie muss, neben anderen Werten, eine Versorgungsspannung von $5\text{ V} \pm 0,5\text{ V}$, einen maximalen Strom von 50 mA und eine Taktfrequenz von 1 bis 5 MHz haben. Die Kräfte pro Kontakt dürfen 0,6 N nicht überschreiten. Die Datenübertragung auf physikalischer Ebene ist im Grunde genommen identisch mit ISO/IEC 7816-3. Dies betrifft sowohl die Zeitdauer für die einzelnen Bits,¹⁹ den ATR²⁰ und die beiden Übertragungsprotokolle $T = 0$ und $T = 1$.²¹ Die APDU Festlegung²² ist identisch mit der in der ISO/IEC 7816-4 beschriebenen.

Im zweiten Teil der Spezifikation (*book 2 – security and key management*) sind die Sicherheitsmechanismen und das Schlüsselmanagement beschrieben. Das dritte Buch (*book 3 – application specification*) enthält schließlich die Datenelemente, Kommandos und Abläufe, die für eine Debit- oder Kredit-Zahlungstransaktion gemäß EMV notwendig sind. Der vierte Teil (*book 4 – cardholder, attendant, and acquirer interface requirements*) enthält schließlich viele allgemeine benutzer- und händlerseitige Festlegungen für ein EMV-konformes Zahlungsverkehrssystem.

Da es sich bei der typischen Kreditkarte um einen Massenartikel handelt, dürfen die Herstellungskosten natürlich nicht hoch sein. Da diese aber zu einem überwiegenden Teil vom implantierten Mikrocontroller abhängen, wurde von Anfang an die Anwendung

¹⁸ Siehe auch Abschnitt 3.1.1 „Formate“.

¹⁹ Siehe auch Abschnitt 6.1 „Physikalische Übertragungsschicht“.

²⁰ Siehe auch Abschnitt 6.2 „Answer to Reset – ATR“.

²¹ Siehe auch Abschnitt 6.4 „Übertragungsprotokolle“.

²² Siehe auch Abschnitt 6.5 „Struktur der Nachrichten – APDU“.

„Kreditkarte“ auf wenig Speicherverbrauch ausgelegt. Deshalb passt eine übliche EMV-Anwendung ohne Zusatzfunktionen auch auf einen Mikrocontroller mit 6 kByte ROM, 1 kByte EEPROM und 128 Byte RAM. Selbst im Chipkartenbereich sind diese Werte am unteren Ende der Bandbreite möglicher Chips, doch ermöglichen sie eine kostengünstige Herstellung.

Die EMV-Spezifikationen sind zum Teil auch Grundlagenpapiere, die den Minimalstandard für die verschiedenen Kartenherausgeber festlegen. Die zurzeit aktuelle Version 2 lässt auch noch einige Punkte offen, so ist beispielsweise das Risikomanagement von Terminal und Chipkarte bei einer Transaktion nicht exakt festgelegt. Deshalb wird die EMV-Anwendung hier nur im Überblick beschrieben, da viele Details vom jeweiligen Kartenherausgeber für sein System festgelegt werden müssen.

Dateien und Datenelemente

Die Spezifikation für die Kreditkarte mit Chip legt lediglich fest, dass für Dateien eine Baumstruktur verwendet werden muss.²³ Die Anwendung selbst befindet sich in einem eigenen DF, das mittels eines AID (*application identifier*) selektiert wird und alle Datenelemente der Anwendung „Kreditkarte“ enthält. Diese sind in EFs mit den üblichen Dateistrukturen nach ISO/IEC 7816-4 untergebracht. Üblicherweise lassen sich die EFs mittels Short-FID implizit selektieren. Der große Unterschied zu ähnlichen Spezifikationen besteht darin, dass keine bestimmten EFs und ihre FIDs festgelegt sind. Dies ist für die Funktionalität im Zahlungsablauf auch nicht notwendig, da alle dafür benötigten Daten mit bereitgestellten Kommandos bearbeitet werden können. Direkt unter dem MF befindet sich lediglich eine Datei (EF_{DIR} nach ISO/IEC 7816-4/5), die alle Informationen über vorhandene Anwendungen der Karte enthält.

Alle Datenelemente im System Terminal-Karte sind mit eindeutigen Templates und Kennzeichen (*tags*) festgelegt. Mit den festgelegten Kommandos können sie innerhalb der Anwendung angesprochen werden, ohne dass ihre genaue Lage im Dateibaum oder in einer Datei bekannt sein muss. Somit ist es möglich, die Einteilung in Dateien dem Kartenherausgeber zu überlassen, da dies den Ablauf einer Zahlungstransaktion nicht beeinflusst.

Kommandos

Für den eigentlichen Ablauf der Zahlungsfunktionen sind vom technischen Prinzip her nur drei Kommandos erforderlich. Allerdings benötigen Personalisierung, Verwaltung, Sonderfunktionen und Zusatzanwendungen noch weitere Kommandos. Dies fällt jedoch nicht in den Bereich der EMV-Spezifikation. Die folgenden Kommandos müssen aufgrund der Vorgaben der EMV-Spezifikation zur Verfügung stehen, wobei alle verwendeten Return-codes analog ISO/IEC 7816-4 sind:²⁴

- APPLICATION BLOCK (EMV-spezifisch)
- APPLICATION UNBLOCK (EMV-spezifisch)
- CARD BLOCK (EMV-spezifisch)
- EXTERNAL AUTHENTICATE (als Teilmenge von ISO/IEC 7816-4)
- GENERATE APPLICATION CRYPTOGRAM (EMV-spezifisch)

²³ Siehe auch Abschnitt 6.5 „Struktur der Nachrichten – APDU“.

²⁴ Siehe auch Abschnitt 7.15 „Kommandos für Kredit- und Debitkarten“.

- GET CHALLENGE (ISO/IEC 7816-4)
- GET DATA (EMV-spezifisch)
- GET PROCESSING OPTIONS (EMV-spezifisch)
- INTERNAL AUTHENTICATE (als Teilmenge von ISO/IEC 7816-4)
- PIN CHANGE/UNBLOCK (EMV-spezifisch)
- READ RECORD (als Teilmenge von ISO/IEC 7816-4)
- SELECT (als Teilmenge von ISO/IEC 7816-4)
- VERIFY (als Teilmenge von ISO/IEC 7816-4)

Kryptografie

Die in einer Anwendung verwendeten kryptografischen Mechanismen sind naturgemäß sehr stark von den dazugehörigen Rahmenbedingungen abhängig. Dies ist besonders gut bei der EMV-Anwendung zu sehen. Eine Grundvoraussetzung für die Gestaltung des Systems war von Anfang an, dass die verwendeten Terminals je nach Betreiber kein Sicherheitsmodul besitzen. Damit können symmetrische Kryptoalgorithmen nicht verwendet werden, da eine Geheimhaltung der Schlüssel nicht möglich ist. Der Grund für die fehlenden Sicherheitsmodule war, dass sie die Terminals erheblich verteuert hätten und ein weltweites Schlüsselmanagement für teilweise offline betriebene Terminals sehr aufwendig und teuer ist.

Um im Rahmen dieser Voraussetzungen eine Authentisierung der Chipkarte durch das Terminal zu ermöglichen, muss man asymmetrische Kryptoalgorithmen verwenden. Dazu benutzt man bei EMV eine einseitige statische Authentisierung mit kartenindividuellen Schlüsseln.²⁵ Die Chipkarte verfügt zwar nicht über die Möglichkeit, die Authentizität des Terminals zu prüfen, doch besteht bei der EMV-Anwendung dazu auch keine zwingende Notwendigkeit, und zwar deshalb, weil von der Karte kein Betrag abgebucht, sondern lediglich ein Transaktionszertifikat für das Terminal erstellt wird. Dieses Transaktionszertifikat ist seitens des Karteninhabers nicht anonym und kann auch nur von einem zugelassenen, d.h. bekannten Händler beim jeweiligen Kartenherausgeber eingereicht werden. Damit entfallen weitgehend Betrugsmöglichkeiten, da ein gültiges Transaktionszertifikat nur von einem autorisierten und dem Kartenherausgeber bekannten Händler in Geld „umgetauscht“ werden kann.

Die verwendeten kryptografischen Algorithmen können im Prinzip beliebig sein, da sowohl die dazugehörigen Datenelemente als auch die Algorithmen selber über TLV-codierte Datenstrukturen eindeutig identifiziert werden. In der Version 2 der EMV-Spezifikation ist als Hash-Algorithmus der SHA-1 nach FIPS 180-1 bzw. ANSI X 9.30-2 vorgesehen.²⁶ Als kryptografischer Algorithmus wird aber nicht, wie aus der Benutzung von SHA-1 erwartet, der DSS, sondern der RSA-Algorithmus (ANSI X 9.31-1) verwendet.²⁷ Die Schlüssellänge kann je nach Kartenherausgeber variiert werden und wird durch eine entsprechende Codierung (Kennzeichen der Signatur) angezeigt. Als öffentlicher Schlüssel werden aus Gründen der Berechnungsgeschwindigkeit kleine Zahlenwerte vorgeschlagen, wie z. B. 3 oder $2^{16} + 1$.

²⁵ Siehe auch Abschnitt 4.11.3 „Statische asymmetrische Authentisierung“.

²⁶ Siehe auch Abschnitt 4.9 „Hash-Funktionen“.

²⁷ Siehe auch Abschnitt 4.7.2 „Asymmetrische Kryptoalgorithmen“.

Für den Fall, dass die Chipkarte eine Online-Verbindung mit dem Hintergrundsystem aufgebaut hat, ist es möglich, die Datenübertragung mit Secure Messaging nach ISO/IEC 7816-4 abzusichern.²⁸ Im Falle dieser End-to-end-Kommunikation wird ein symmetrischer kryptografischer Algorithmus benutzt, und zwar der Triple-DES. Was in diesem Falle ohne Einschränkung der Systemsicherheit möglich ist, da sowohl Hintergrundsystem als auch Chipkarte den geheimen Schlüssel sicher aufbewahren können.

System und Transaktionsabläufe

Der Systemaufbau für Zahlungsverkehrssysteme im Kreditkartenbereich ist traditionell sehr zentralistisch ausgelegt. Es gibt üblicherweise mehrere Hintergrundsysteme, die alleine oder mit weiteren Systemen für einen bestimmten Raum (z. B. Deutschland) zuständig sind. Diese mit leistungsfähigen Computern ausgestatteten Rechenzentren sind über das eigenständige Netzwerk des jeweiligen Kreditkartenherausgebers miteinander verbunden. Dies dient zum Datenaustausch für das Clearing und erhöht auch die Betriebssicherheit, da beim Ausfall eines Rechenzentrums andere Hintergrundsysteme die Aufgaben mit übernehmen können. Die öffentlichen Telefon- und Datennetze (z. B. ISDN, X.25) verbinden die einzelnen Terminals mit dem Hintergrundsystem. Dazwischen kann sich noch ein Acquirer befinden, der das Routing und die Zusammenfassung von Transaktionsdaten übernimmt. Dies ist jedoch stark abhängig vom jeweiligen Kartenherausgeber bzw. vom jeweiligen Land. Auf Terminalebene gibt es folgende beide Varianten: direkter Datenaustausch mit dem Acquirer oder Einbindung des Konzentrators des Händlers bzw. der Handelskette. Beide Varianten sind möglich und werden auch eingesetzt.

Der Ablauf einer Zahlung mit der Kreditkarte mit Chip ändert sich mit dem neuen Chipkartensystem nicht wesentlich. Der Kunde legt an der Kasse seine Kreditkarte vor, die in ein Chipkarten-Terminal gesteckt wird. Ist ein solches nicht vorhanden, kann die Zahlung durch Benutzung des nach wie vor vorhandenen Magnetstreifens oder der Hochprägung auf konventionelle Weise getätigt werden. Ist aber ein Chipkarten-Terminal vorhanden, besteht nach wie vor die Möglichkeit, durch Leistung einer Unterschrift die Identität des

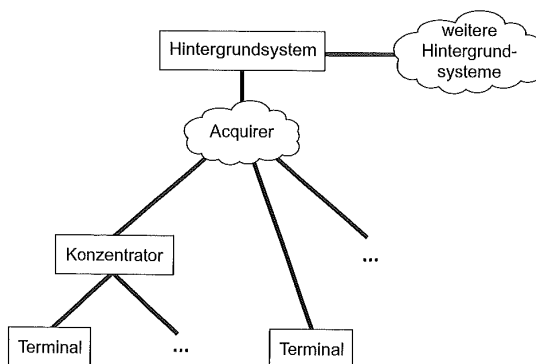


Bild 12.17 Grundlegender Aufbau des Zahlungsverkehrssystems für EMV-Karten.

²⁸ Siehe auch Abschnitt 6.6 „Sicherung der Datenübertragung“.

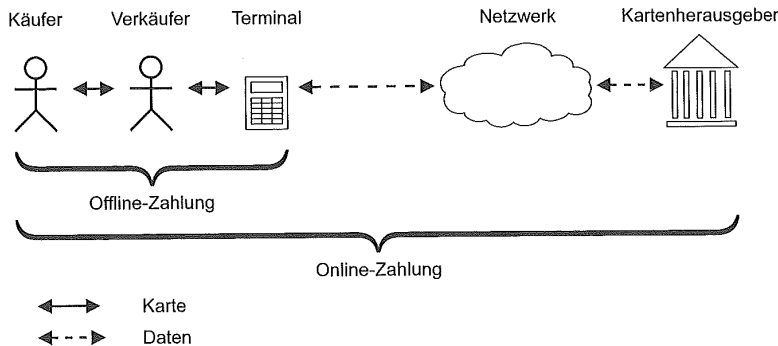


Bild 12.18 Grundlegende Infrastruktur beim Zahlungsablauf mit einer Chipkarte nach der EMV-Spezifikation.

Kartenbesitzers zu prüfen. Der dazugehörige Transaktionsbeleg enthält aber ein Kennzeichen, wonach der Kartenbesitzer mittels Unterschrift identifiziert wurde. Die zweite Variante sieht die Eingabe einer vierstelligen PIN vor. Diese kann online vom Hintergrundsystem, doch auch offline von der Chipkarte geprüft werden. Das dazu generierte Transaktionszertifikat enthält dann einen Hinweis zur Art der PIN-Prüfung. Der Ablauf einer Zahlung ist in Bild 12.18 im Überblick dargestellt.

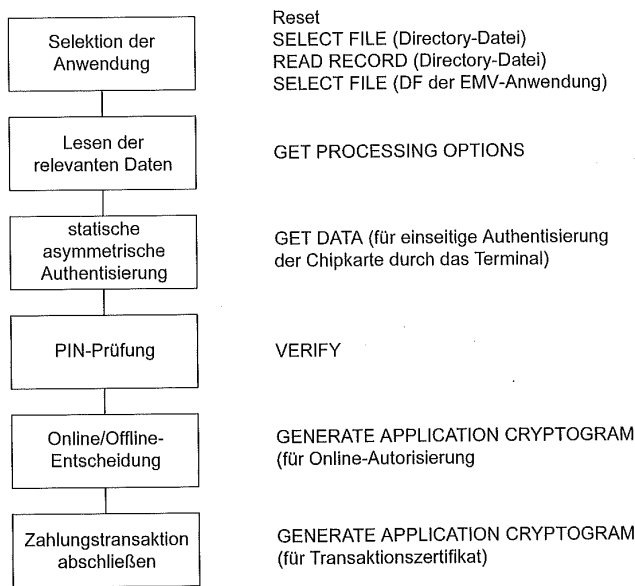


Bild 12.19 Stark schematisierter Zahlungsablauf mit einer Chipkarte nach der EMV-Spezifikation aus Sicht des Terminals.

Im Detail sind die einzelnen Funktionen und der Ablauf für eine erfolgreiche Zahlung mit Chipkarte natürlich etwas aufwendiger. Ein Beispiel dafür gibt Bild 12.19, welches die grundlegenden Mechanismen dazu zeigt. Sowohl Chipkarte als auch Terminal berechnen anhand verschiedener Transaktionsdaten (z. B. Höhe des Betrags) den genauen Ablauf der Transaktion. Überschreitet beispielsweise der zu zahlende Betrag eine bestimmte Summe, fordert die Chipkarte eine Online-Autorisierung der Zahlung durch das Hintergrundsystem. Das Terminal muss dann eine Verbindung zu diesem aufbauen und sich die Zahlung bestätigen lassen. Erst mit einer zustimmenden Antwort vom Hintergrundsystem erzeugt die Chipkarte ein gültiges Transaktionszertifikat, das der Händler dann zum Clearing einreichen kann. Der Zweck der Online-Autorisierung ist die Minimierung des finanziellen Risikos für den Kartenherausgeber. Da sich eine Kreditkarte mit Chip aufgrund mehrerer Kriterien immer wieder online beim Hintergrundsystem „melden“ muss, kann man den maximalen finanziellen Verlust bei gestohlenen oder manipulierten Karten innerhalb genau berechenbarer Grenzen halten.

Zukünftige Entwicklung

Die EMV-Spezifikation lässt in ihrem Aufbau und ihren Inhalten den Kartenherausgebern sehr viel Spielraum für Eigeninitiative, sodass Weiterentwicklungen und eigene Varianten möglich sind. Diese Flexibilität wird sicherlich von den verschiedenen Firmen ausgiebig wahrgenommen. So hat jeder der an der EMV-Spezifikation beteiligten Kreditkartenherausgeber eine oder mehrere elektronische Geldbörsen²⁹ spezifiziert, die bei Bedarf zusätzlich auf der Chipkarte Platz finden können. Vor allem im Bereich von Automaten oder bei geringen Beträgen hat eine vorbezahlte elektronische Geldbörse durchaus Vorteile, da die bei Kreditkarten sonst obligatorische Transaktionsgebühr nicht anfällt.

Die fortschreitende kommerzielle Nutzung von weltweiten Netzwerken, wie beispielsweise dem Internet, erfordert schon in naher Zukunft sichere, weltweit verbreitete und akzeptierte Zahlungsmöglichkeiten. Kreditkarten mit Chip und einigen zusätzlichen Funktionen würden sich sehr gut dafür eignen, da sie schon seit langer Zeit unabhängig von bestimmten Ländern oder Währungen eine internationale Akzeptanz und Verbreitung haben. Auch daran wird zurzeit gearbeitet. Der Unterschied zur oben vorgestellten Zahlungstransaktion ist nicht groß, da an die Stelle des Acquirers nun beispielsweise das Internet tritt. In erster Linie ist es lediglich notwendig, dass dem Kunden ein Chipkarten-Terminal zur Verfügung steht.

Die EMV-Spezifikation hat im Zahlungsverkehrsbereich die gleiche Bedeutung wie die GSM 11.11 Spezifikation für Chipkarten in Telekommunikationsanwendungen erlangt. Es gibt keine neuen Chipkarten-Betriebssysteme für Zahlungsverkehrs-Anwendungen am Markt, die nicht für sich in Anspruch nehmen, „EMV-kompatibel“ zu sein. Aufgrund der in Zukunft zu erwartenden Stückzahlen von Kreditkarten mit Chip wird diese Spezifikation den Mindeststandard für alle zukünftigen Anwendungen in diesem Bereich setzen.

12.5 Das ec-System in Deutschland

Deutschland unterscheidet sich im Zahlungsverkehr mit Karten anderer Länder dadurch, dass anstelle von Kreditkarten traditionell sehr viele Debitkarten (ec-Karten) verwendet werden. Mit dieser Art von Karte kann man nach Eingabe einer PIN an vielen Stellen be-

²⁹ Siehe auch Abschnitt 12.3.1 „CEN-Norm EN 1546“.

zahlen, und der dabei fällige Betrag wird dann unmittelbar von einem mit der Karte verbundenen Konto abgebucht. Der Händler muss für jede Zahlungstransaktion einen festen Betrag abführen, der allerdings nicht allzu hoch ist. Die Akzeptanz von Kreditkarten ist im Gegensatz dazu aufgrund der umsatzabhängigen Gebühren in Deutschland nur mittelmäßig. Daneben existiert noch die weit verbreitete Variante, dass der Kunde seine Einwilligung gibt, den Kaufpreis via Lastschrift auf das Konto des Händlers zu überweisen (POS ohne Zahlungsgarantie – POZ). Die ec-Karte dient dabei nur als Referenz zum Kundenkonto, auf das in diesem Fall online eine Bonitätsabfrage durchgeführt wird. Der Händler erhält in diesem Fall allerdings von seiner Bank keine Zahlungsgarantie, wie dies bei Transaktionen mit Kreditkarten und ec-Karten (ec-Cash, Geldkarte) der Fall ist.

Da die ec-Transaktionen in den meisten Fällen online von einem Hintergrundsystem autorisiert werden müssen, fallen zusätzlich noch Kosten für die Datenübertragung bzw. für die Anmietung einer Standleitung an. Da sich dies nur für Händler mit großem Umsatz und vielen Zahlungen mit ec-Karte rechnet, ist man seit einigen Jahren auf der Suche nach Verbesserungen des bestehenden Systems. Die Akzeptanz im Handel würde schlagartig steigen, wenn die hohen Kosten für die Telekommunikation wegfielen. Man benötigt also ein offline-fähiges System.

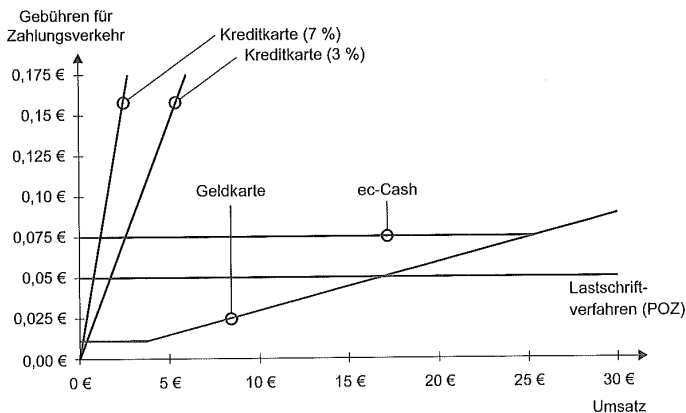


Bild 12.20 Die Gebühren des Händlers für den elektronischen Zahlungsverkehr in Deutschland. Im Lastschriftverfahren (POS ohne Zahlungsgarantie – POZ) werden umsatzunabhängig 5 Euro Cent pro Transaktion verrechnet. Dazu unterschreibt der Kunde die Einwilligung zu einer Lastschrift, die Bank des Händlers kann dann den Kaufbetrag vom Konto des Käufers abbuchen. Bei einem Kauf mit der Geldkarte muss der Händler mindestens 0,3 % des Kaufbetrags, jedoch mindestens 1 Euro Cent bezahlen. Benutzt der Kunde ec-Cash, muss der Händler 0,3 % vom getätigten Umsatz, mindestens jedoch 7,5 Euro Cent bezahlen. Bei Kreditkarten muss der Händler je nach Vertrag zwischen 3 % und 7 % des Umsatzes abführen, wobei es vertragsabhängig auch einen Mindestumsatz geben kann.

Der Zentrale Kreditausschuss (ZKA), eine Arbeitsgemeinschaft der Spitzenverbände der deutschen Kreditwirtschaft, beschloss 1993 eine Ausschreibung zum Konzept einer multifunktionalen Chipkarte (MFC),³⁰ die für den elektronischen Zahlungsverkehr geeignet sein

³⁰ Der Begriff „multifunktionale Chipkarte“ wurde in dieser Zeit und in diesem Zusammenhang geprägt.

sollte. Nachdem mehrere Firmen entsprechende Angebote abgegeben hatten, wurde eines davon ausgewählt und die Rechte daran vom ZKA erworben. Aufgrund geänderter technischer Rahmenbedingungen wurde es anschließend stark überarbeitet. Als Ergebnis der Revision gibt es nun eine Reihe von Spezifikationen für die ec-Karte mit Chip, von denen jede ein bestimmtes Gebiet abgedeckt. Leider sind diese Spezifikationen vertraulich, sodass es nicht möglich ist, detaillierte Informationen zu veröffentlichen. Im Folgenden ein kurzer Überblick über die Themenbereiche:

- Betriebssystem SECCOS
- Geldkarte
- ec-Cash mit Chip
- Digitale Signatur
- EMV
- elektronischer Fahrschein
- elektronischer Marktplatz
- Personalisierung

Diese Dokumente beschreiben eine Zahlungsverkehrskarte, deren Funktionalität der heutigen ec-Karte entspricht und noch zusätzlich eine elektronische Geldbörse enthält. Auch ist es nach Ausgabe der Karte möglich, beliebige neue Anwendungen in die Karte nachzuladen.

Vor der bundesweiten Einführung der neuen Karte gab es im Frühjahr 1996 in der Region Ravensburg und Weingarten (Nähe Bodensee) mit einem Einzugsgebiet von 250 000 Einwohnern einen groß angelegten Feldversuch mit etwa 100 000 ec-Karten und ungefähr 500 Terminals. Anschließend wurde im Herbst 1996 mit der groß angelegten deutschlandweiten Einführung der neuen Karte begonnen. In Deutschland sind mittlerweile etwa 50 Millionen ec-Karten mit Chip ausgegeben worden, die sukzessive alle drei Jahre ausgetauscht werden. Bis Ende 1998 wurden in Deutschland 250 000 Terminals installiert, an welchen jährlich 300 Millionen Transaktionen stattfinden. Der kumulierte Umsatz der magnetstreifen- und chipbasierten Transaktionen von ec-Cash und Geldkarte betrug im Jahr 1999 etwa 19 Milliarden Euro. Die 41 000 Geldausgabeautomaten in Deutschland wurden nahezu vollständig mit Chipkartenterminals aufgerüstet. Mittlerweile gibt es deutschlandweit etwa 30 000 Ladeterminals für die Geldkarte. Damit ging aus diesem Feldversuch weltweit eines der größten Zahlungsverkehrssysteme mit Chipkarten hervor.

2001 wurde eine statistische Erhebung zur elektronischen Geldbörse Geldkarte durchgeführt. Die dabei ermittelten Zahlen zeigen ein deutliches Bild des aktuellen Einsatzes. Von den 50 Millionen ausgegebenen Chipkarten mit der Geldkarten-Anwendung wurden aktiv nur ca. 4 Millionen genutzt, wobei etwa 14 Millionen Transaktionen in einem halben Jahr zustande kamen. Der durchschnittliche Ladebetrag betrug 30 Euro und 2,30 Euro war der durchschnittliche Bezahlungsbetrag.

Funktionen für den Benutzer

Die deutsche ec-Karte mit Chip hat für den Benutzer mehrere unterschiedliche Funktionen. Nach Eingabe einer PIN kann mit der Karte on- oder offline an einem entsprechenden Terminal bezahlt werden. Der anfallende Betrag wird dann von einem zugeordneten Kartenkonto bei der herausgebenden Bank abgebucht. Diese Anwendung heißt in diesem Kapitel einheitlich ec-Cash, obwohl es auch andere Bezeichnungen dafür gibt.

Natürlich kann mit der ec-Karte mit Chip auch Geld an den Geldausgabeautomaten der verschiedenen Banken abgehoben werden, doch gehört dies funktionell gesehen eigentlich noch in den ec-Cash-Bereich. Dass von der Chipkarte ebenfalls die unterschiedlichen Funktionen der Foyerautomaten, wie beispielsweise Kontoauszugsdrucker, unterstützt werden, ist selbstverständlich. Zusätzlich besitzt die Karte eine vorbezahlte elektronische Geldbörse namens Geldkarte, mit der auch ohne PIN-Eingabe bezahlt werden kann. Diese Börse existiert in einer anonymen und einer nicht anonymen Variante. Sie kann immer wieder an entsprechenden Ladeterminals (z. B. Terminal am Bankschalter, Geldausgabeautomaten) entweder gegen Bargeld oder gegen eine ec-Cash-Transaktion aufgeladen werden.

Eine Funktion der ec-Karte besteht auch darin, zusätzliche Anwendungen mit allen dazugehörigen Dateien und Kommandos nach Ausgabe der Karte an den Karteninhaber nachzuladen. Diese Funktion wurde jedoch bislang wenig genutzt, weil der Abstimmungsbedarf und die Zulassungsvoraussetzungen für eine neue Anwendung verhältnismäßig aufwendig sind.

Gesamtsystem im Überblick

Wie bei großen Zahlungsverkehrssystemen üblich, gibt es auch bei der neuen deutschen ec-Karte mit Chip kein zentrales Clearing. Es existieren 4 Rechenzentren, die den Abrechnungsvorgang zwischen Händler, Karteninhaber und den beteiligten Kreditinstituten durchführen. Im folgenden Bild 12.21 ist im Überblick das Clearing bei der Anwendung „elektronische Geldbörse“ in der nicht anonymen Variante aufgezeigt. Im Übrigen heißt die Clearinginstanz in Deutschland „Börsenevidenzzentrale“ (BEZ). Daneben gibt es noch zirka 25 so genannte Ladezentralen, die im Abgleich mit den Börsenevidenzzentralen Ladevorgänge auf Geldkarten durchführen.

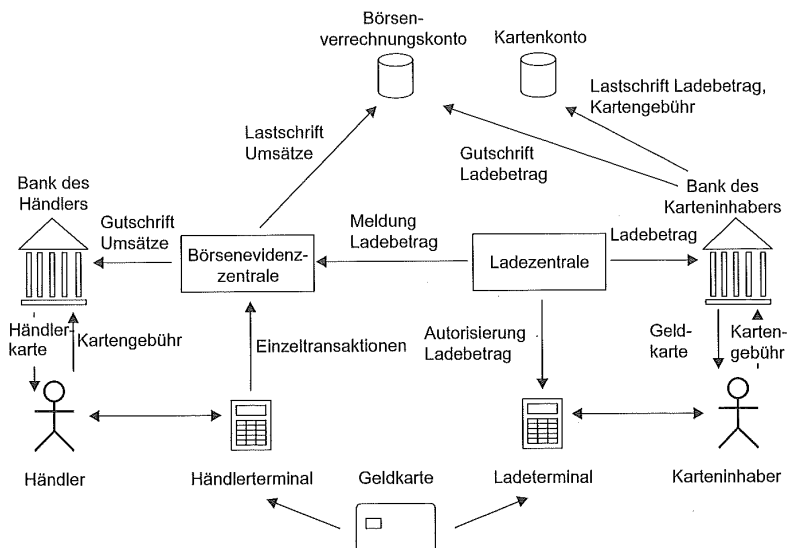


Bild 12.21 Grundlegender Aufbau des Zahlungsverkehrssystems für die deutsche Geldkarte, die eine kontenbezogene und somit nicht anonyme elektronische Geldbörse ist.

Der gesamte Zahlungsverkehr basiert auf zwei Konten: dem Börsenverrechnungskonto, welches immer den Saldo der elektronischen Geldbörse auf der Karte widerspiegelt, und dem Kartenkonto, das beispielsweise das Gehaltskonto des Karteninhabers sein könnte. Das Börsenverrechnungskonto ist ein so genanntes Schattenkonto (*shadow account*), das parallel zur elektronischen Geldbörse geführt wird. Wird ein Geldbetrag in die elektronische Geldbörse geladen, findet analog und zeitgleich eine entsprechende Buchung auf das Börsenverrechnungskonto/Schattenkonto statt, da dieser Vorgang grundsätzlich online ausgeführt werden muss. Je nachdem, ob der Bezahlvorgang on- oder offline stattgefunden hat, werden zeitgleich oder zeitversetzt die entsprechenden Buchungen auf dem Börsenverrechnungskonto vorgenommen. Der große Vorteil bei diesem kontengebundenen System liegt darin, dass der Börsensaldo bei Verlust oder Funktionsunfähigkeit der elektronischen Geldbörse mit einer gewissen zeitlichen Verzögerung rekonstruiert werden kann. Allerdings ist eine Anonymität der Zahlungen nicht mehr gewährleistet. Es gibt aber auch eine anonyme Variante der elektronischen Börse mit einem Schattenkonto, das jedoch ohne Bezug zu einem Kundenkonto ist.

ec-Karte mit Chip

Es gibt im deutschen ec-System mehrere unterschiedliche Kartentypen. Die Chipkarten für die normalen Bankkunden gliedern sich wiederum in mehrere Untergruppen auf:

- ec-Karte (ec-Cash und Geldkarte mit Kontobezug, d. h. nicht anonym)
- Geldkarte mit Kontobezug (d. h. nicht anonym)
- Geldkarte ohne Kontobezug (d. h. anonym)

In der ec-Karte befinden sich zwei DFs, die jeweils die Anwendung ec-Cash und Geldkarte enthalten.³¹ Bei den Karten mit einer elektronischen Börse fehlt das DF für ec-Cash. Der Kontobezug, d. h. ob die Karte anonym oder nicht anonym ist, wird lediglich durch bestimmte Datenelemente hergestellt.

Seitens der Händler gibt es für ihre Terminals eine Händlerkarte im ID-000 Format (Plug-In) mit allen notwendigen Kommandos und Dateien, um geldwerte Transaktionen abzuwickeln. Dies ist sozusagen das Sicherheitsmodul der Händlerterminals. Dabei existiert eine Besonderheit, die technisch durchaus eine gewisse Attraktivität besitzt. Im deutschen ec-System gibt es nämlich reale und virtuelle Händlerkarten. Die reale Händlerkarte ist die übliche Chipkarte, als Plug-In geformt. Die virtuelle Händlerkarte ist lediglich die Software-Simulation einer realen Händlerkarte. Diese läuft geschützt im Sicherheitsmodul (SAM) des Händlerterminals ab.

Ursprünglich war diese Lösung ein Kompromiss, damit Terminals ohne Steckplatz für Plug-In-Karten auch noch für das neue System verwendet werden konnten. Mittlerweile hat sich aber herausgestellt, dass diese Lösung einige technisch sehr positive Eigenschaften aufweist. Eine virtuelle Händlerkarte lässt sich problemlos über Fernwartung austauschen, da sie nur aus Software besteht. Zudem ist ihre Lebensdauer deutlich höher als die von realen Karten, da der nachteilige Einfluss der beschränkten Anzahl von EEPROM-Schreibzyklen nicht mehr auftritt. Schließlich ist ein gutes Hardware-Sicherheitsmodul mindestens so sicher wie eine Chipkarte, da aufgrund der gepufferten Stromversorgung die Sicherheitsmechanismen jederzeit aktiv sind.

³¹ Der Begriff Geldkarte ist leider nur im jeweiligen Kontext eindeutig. „Geldkarte“ bezeichnet sowohl die elektronische Geldbörse als Anwendung auf einer Chipkarte als auch die Chipkarte selber.

Das gesamte informationstechnische Konzept sowie das Sicherheitsmodell der Karte lehnt sich stark an die Normenreihe ISO/IEC 7816 an. Die ursprüngliche Version, heute Typ 1 Geldkarte genannt, beinhaltete einige spezifische Mechanismen, doch wurde dies mit den neueren Versionen abgelöst. Mittlerweile hat man ein vollständiges Chipkarten-Betriebssystem mit PKI-Funktionen spezifiziert, das den Namen SECCOS (*security card operating system*) trägt und alle wesentlichen Mechanismen der ISO/IEC 7816 Normen unterstützt. Sehr aufwendig wurden auch die Sicherheits- und Zugriffsmechanismen von ISO/IEC 7816-9 mit einbezogen, sodass die deutsche ec-Karte mit Chip momentan wohl die weltweit vollständigste Umsetzung dieser Norm ist. Auch Elemente aus der EMV-Spezifikation für Kreditkarten sind aus Kompatibilitätsgründen mitbestimmend für die Spezifikationen der ec-Karte gewesen.

Bei den technischen Rahmenwerten, die in den Spezifikationen vorgegeben worden sind, orientiert sich die Karte an vielen vorhandenen Standards. Die Größe entspricht selbstverständlich ID-1 und somit der heutigen ec-Karte. Außerdem ist sie als Hybridkarte mit Magnetstreifen und Chip ausgelegt, sodass ein Übergang von den heute üblichen Terminals mit Magnetstreifenleser zu den neueren Terminals mit Chipkartenkontaktiereinheit ohne Inkompatibilitäten möglich sein wird. Die Karte benutzt das Übertragungsprotokoll T = 1 mit PPS. Als kryptografischer Algorithmus findet der Triple-DES Verwendung. Sicherheitstechnisch ist noch interessant, dass sowohl die komplette Software als auch die Hardware der Chipkarte mit einem Sicherheitsgutachten anhand des ZKA-Kriterienkatalogs³² überprüft werden muss.

Die Dateiverwaltung unterstützt DFs auf mehreren Ebenen, sowie die Selektion mit Short-FID, FID und AID. Es werden die üblichen Dateistrukturen „transparent“, „linear fixed“, „linear variable“ und „cyclic“ unterstützt, die auch implizit in den dazu passenden Kommandos selektiert werden können. Die Größe der beiden Record-orientierten Dateistrukturen ist maximal 254 Records zu je 255 Byte. Dies wird natürlich momentan von keiner existierenden Anwendung ausgenutzt.

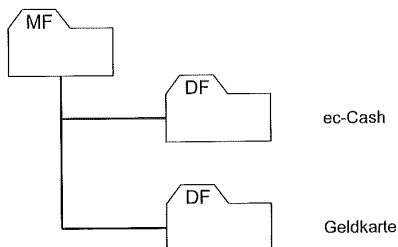


Bild 12.22 Der grundlegende Dateibaum in der deutschen ec-Karte mit Chip mit den beiden Anwendungen ec-Cash und Geldkarte.

Das Dateimanagement verwendet einen speziellen Mechanismus für die Zuordnung von EFs zu bestimmten Anwendungen. Durch diese Funktion, die durch zwei nicht genormte Kommandos realisiert ist, können EFs über die Grenzen von DFs hinweg einer Anwendung zugeordnet werden. Damit ist es möglich, innerhalb eines DFs mit der Short-FID ein EF zu

³² Siehe auch Abschnitt 9.3.1 „Evaluierung“.

selektieren, das sich in einem anderen DF befindet. Es kann damit ein EF mehreren unterschiedlichen DFs zugeordnet werden. Dies entspricht im Prinzip dem Alias-Mechanismus in vielen PC-Betriebssystemen. Dabei wird bezweckt, mehreren Anwendungen Dateien mit allgemeinen Informationen ohne aufwendige Selektion über Anwendungsgrenzen hinweg zur Verfügung zu stellen. Die insofern verschiedenen Anwendungen zugeordneten EFs können mit Short-FID oder FID selektiert und nach Erreichen der dazu notwendigen Sicherheitszustände gelesen und geschrieben werden. Alle Dateien haben bestimmte Zugriffszustände, die Lesen und Schreiben von vorher erreichten Zuständen (z. B. PIN-Eingabe) abhängig machen. Auch ist es mit diesem objektorientierten System möglich, Zugriffe auf Dateien abhängig von der Sicherung der Datenübertragung zu machen. Das heißt, es gibt Dateiattribute, die einen Zugriff mit Secure Messaging erzwingen können.

Die vorhandenen Kommandos teilen sich in vier Klassen auf: Die Kommandos nach ISO/IEC 7816-4, die jedoch einen gegenüber der Norm reduzierten Funktionsumfang haben. Die ec-spezifischen Administrationskommandos werden für Verwaltungszwecke der Karte benutzt. Damit können neue Dateien angelegt, vorhandene gelöscht und neue Kommandos in die Chipkarte eingebracht werden. Die dritte Kommandoklasse – die Ergänzungskommandos – benutzt man, um die Funktionalitäten für ec-Cash und Geldkarte zu erreichen. Administrations- und Ergänzungskommandos sind rein ec-spezifisch und haben im Prinzip keine Beziehung zur internationalen Normung. Als vierte Klasse wurden auch Initialisierungs- und Personalisierungskommandos spezifiziert.

Wie aus der vorangehenden überblickshaften Beschreibung ersichtlich ist, verfügt die ec-Karte über einen verhältnismäßig großen Funktionsumfang. Dies führt zwangsläufig zu einem hohen Speicherbedarf. Derzeit benutzt man deshalb als Zielhardware zum Großteil Mikrocontroller mit 100 kByte ROM, 32 kByte EEPROM und 2 kByte RAM. Eine reine Börsenkarte ohne ec-Cash-Funktionalität benötigt etwa 48 kByte ROM, 16 kByte EEPROM und 1 kByte RAM. Zusammengefasst bedeuten diese Speicheranforderungen, dass man verhältnismäßig große Chips verwenden muss, um den umfangreichen Programmcode sowie die 2,3 kByte Anwendungsdaten für die elektronische Börse Geldkarte und die 1,6 kByte Anwendungsdaten für ec-Cash unterzubringen.

Zusatzanwendungen

Das Betriebssystem der ec-Karte umfasst Kommandos und Mechanismen, um ausführbaren Programmcode nachzuladen. Letzterer muss allerdings individuell für jedes im Einsatz befindliche Betriebssystem und jeden Typ von Mikrocontroller sein, denn es kann nur adressabhängiger Maschinencode nachgeladen werden. Der aufgrund dieser Problematik notwendige logistische Aufwand beim Nachladen neuer Kommandos ist der Hauptgrund, warum dieser Mechanismus zurzeit nicht eingesetzt wird.

Für eine Zusatzanwendung müssen jedoch nicht zwangsläufig Programme in die Chipkarte geladen werden. In den meisten Fällen genügt es bereits, über Dateien mit entsprechenden Zugriffsrechten zu verfügen. Die Spezifikation der ec-Karte enthält Kommandos zur Erzeugung von Dateien. Der administrative Aufwand, Zusatzanwendungen individuell für jede Chipkarte online über eine Evidenzzentrale durchzuführen, ist jedoch sehr hoch, was dazu geführt hat, dass dieser Mechanismus sehr selten genutzt wird.

Stattdessen legt man bereits bei der Personalisierung der ec-Karten mehrere Dateien an, in denen zu einem Zeitpunkt nach Ausgabe der ec-Karten neue Anwendungen Platz fin-

den. Die Sparkassenorganisation hat dieses Konzept zur Verwaltung von Dateien für Zusatzanwendungen Space Manager genannt.

Resümee

Das deutsche ec-Kartensystem ist zurzeit eines der umfangreichsten und komplexesten Zahlungsverkehrssysteme mit Chipkarten. Dies betrifft nicht nur die Zahlungsabläufe, sondern auch die dahinterstehende Logistik bei Halbleiterherstellung, Personalisierung und Ausgabe der Karten. Immerhin müssen alle drei Jahre etwa 30 Millionen Chipkarten in einem Zeitraum von weniger als drei Monaten den Kunden erreichen.

Die sicherheitstechnischen Untersuchungen für die Hardware der Mikrocontroller und die Software der Betriebssysteme und Anwendungen setzen ebenfalls Maßstäbe, da die Zulassungsvoraussetzungen streng sind und laufend an neue Gegebenheiten (z. B.: DFA, SPA/DPA u. Ä.) angepasst werden. Interessant sind auch die technisch ausgetüftelten Kompatibilitätstests, die Sicherstellen müssen, dass die Software der verschiedenen Masken auf unterschiedlichen Mikrocontrollern problemlos mit den ebenfalls mannigfaltigen Terminals zusammenarbeitet.

Von den ursprünglich über zwanzig verschiedenen Masken auf über zehn verschiedenen Mikrocontrollern sind mittlerweile nur noch 6 Masken auf 3 Mikrocontrollern übrig geblieben, und es wird aller Voraussicht nach nur noch zwei Maskenhersteller geben. Hier sieht man deutlich, dass alle großen Systeme zur Konsolidierung neigen.

Es wurden und werden Feldversuche mit der Geldkarte in Frankreich, Luxemburg und Island durchgeführt. Eine weitere Perspektive für dieses System zeichnet sich durch verschiedene Pilotversuche ab, bei denen die Geldkarte als Zahlungsmittel im (deutschen Teil des) Internet(s) benutzt wird. Dazu benötigt der Kunde lediglich am heimischen Surf-PC ein einfaches und kostengünstiges Terminal mit dazugehöriger Software. Der Händler benutzt als Gegenstück ein Sicherheitsmodul oder spezielles Terminal, das über das Internet mit dem PC des Kunden verbunden ist.

Der Einsatz von über 50 Millionen Chipkarten beim deutschen ec-System hat Einfluss auf alle großen Zahlungsverkehrsprojekte mit Chipkarten. Die Erfahrungen, die man in Deutschland mit dieser Multiapplication-Chipkarte gemacht hat werden sowohl bei künftigen Chipkarten-Betriebssystemen als auch bei der dazugehörigen Mikrocontroller-Hardware in vielen Punkten der Anstoß für eine erhebliche Weiterentwicklung sein.

13 Chipkarten in der Telekommunikation

13.1	Überblick zu Mobilfunksystemen	735
13.1.1	Vielfachzugriffsverfahren (<i>multiple access</i>).	735
13.1.2	Zellulartechnik	739
13.1.3	Zelltypen	741
13.1.4	Trägerdienste (<i>bearer services</i>).	741
13.2	Das GSM-System	743
13.2.1	Die Spezifikationen	746
13.2.2	Systemarchitektur und Komponenten	748
13.2.3	Wichtige Datenelemente	751
13.2.4	Die SIM (<i>Subscriber Identity Module</i>).	754
13.2.5	GPRS (<i>General Packet Radio Service</i>)	794
13.2.6	Zukünftige Entwicklung.	795
13.3	Das UMTS-System :	796
13.4	Mikrobrowser	805
13.5	Die WIM (<i>Wireless Identification Module</i>)	808
13.6	Öffentliches Kartentelefon in Deutschland.	814

Telekommunikation bzw. der mittlerweile selten gebrauchte Begriff Nachrichtentechnik ist die Technik zum Austausch von Nachrichten über beliebige Entfernungen zwischen Menschen und/oder Maschinen. Sie entwickelte sich über Jahrhunderte von Boten, optischer Signaltechnik bis hin zur Kommunikation mittels drahtgebundener und drahtloser elektrischer Signale. Vorreiter waren meist Militär und Geschäftsleute, für die es überlebenswichtig war, möglichst schnell wichtige Nachrichten zu übermitteln. Den größten Entwicklungsschub hat die Telekommunikation jedoch erst in den letzten Jahren erfahren, als sie aufgrund verschiedener günstiger Rahmenbedingungen – wie Deregulierung des Marktes, preiswerte Herstellung von Endgeräten sowie allgemeinem wirtschaftlichem und technischem Aufschwung – ihren Weg zu einer echten Massen Anwendung fand.

Obwohl bereits 1753 erste Vorschläge veröffentlicht worden waren, um mittels Elektrizität Nachrichten zu übertragen, nahm man erst 1794 die erste Telegrafienlinie zwischen Paris und Lille in Betrieb. In den folgenden Jahrzehnten wurden in vielen Ländern und von zahlreichen Personen Erfindungen auf dem Gebiet der elektrischen Nachrichtenübermittlung gemacht. Im Folgenden seien hier nur die wichtigsten und interessantesten Erfindungen aufgeführt: S. Soemmerring konstruierte 1809 den elektrochemischen Telegrafen, Carl Friedrich Gauß und Wilhelm Weber erfanden 1833 den Induktionstelegrafen, Samuel Morse baute 1835 den schreibenden Telegrafen und publizierte 1840 sein Telegrafen-Alphabet (bekannt als Morse-Alphabet).

Alle diese Erfindungen und oftmals trickreichen Konstruktionen waren jedoch nur in der Lage, elektrische Signalisierungs-codes zu übertragen, nicht die menschliche Stimme. Dies wurde erst ab ca. 1860 möglich. Wie so oft bei großen Erfindungen kann man auch die Erfindung des Telefons keiner einzelnen Person zuordnen. Im Wesentlichen waren es jedoch der Lehrer an einer Gehörlosenschule Johann Philip Reis und der Taubstummenlehrer Alexander Graham Bell, die der Öffentlichkeit Geräte vorführten, mit denen Sprache auf

elektrischem Wege übertragen werden konnte.¹ Beide hatten ursprünglich für ihre Erfindung die Anwendung als Schulungsmittel zum Sprechen für Taube im Sinne. Die große Stärke von Alexander Graham Bell war vor allem die aggressive Vermarktung seiner Erfindung, sodass er auch der bekanntere von beiden Telefon-Erfindern wurde.

Das erste Telefonortsnetz wurde im Jahr 1878 in New Haven, USA aufgebaut. In Berlin ging 1881 das erste öffentliche Telefonortsnetz mit acht Teilnehmern in Betrieb, noch handvermittelt und selbstredend mit allen Eigenschaften der damaligen einfachen Elektrotechnik behaftet. Bereits 1908 wurde jedoch Deutschlands erstes automatisches Vermittlungsamt in Hildesheim mit Hebdrehwählern installiert. Ab dieser Zeit begann die weltweite Installation und Vernetzung der im Fachjargon PSTN (*public switched telephone network*) genannten öffentlichen vermittelten Telefonnetzwerke.

Nachdem es Guglielmo Marconi um 1896 gelungen war, die erste drahtlose Datenübertragung über einige Kilometer zustande zu bringen, dauerte es nur fünf Jahre, bis auch der Atlantik durch drahtlose Telegrafie überbrückt werden konnte. Die Erfindung von Verstärkerröhren begünstigte die technische Entwicklung von Sendern und Empfängern für Sprache und so gelang bereits 1915 die Übertragung von Sprache über den Atlantik. Auf der Rundfunkausstellung in Berlin 1929 wurde dann das erste Bildtelefon, noch in der Größe von drei heute üblichen Telefonhäuschen, vorgestellt.²

In den folgenden Jahren beschleunigte sich die technische Entwicklung in der Telekommunikation und im Mobilfunksektor, so dass es verhältnismäßig früh zur Inbetriebnahme der ersten Mobiltelefonnetze kam.

Die ersten PLMNs (*public land mobile network*) entstanden um 1950, damals für eine kleine und vor allem zahlungskräftige Bevölkerungsschicht. So nahm beispielsweise das erste öffentliche Mobilfunknetz (A-Netz) in Deutschland im Jahr 1958 seinen Betrieb auf. Es war handvermittelt, arbeitete analog im 150 MHz-Bereich und hatte bei einer Flächenabdeckung von ca. 80 % des Bundesgebietes eine Kapazität von maximal 10 500 Teilnehmern. Dieses erste Mobilfunknetz in Deutschland besaß noch keinen zellularen Aufbau, sondern basierte auf wenigen verteilten leistungsstarken Sende- und Empfangsstationen, in deren Versorgungsgebiet man mobil telefonieren konnte. Im Übrigen musste bei einem Anruf an den jeweiligen Mobilfunkteilnehmer dessen Aufenthaltsort ungefähr bekannt sein, damit die richtige Funkzone von der Handvermittlung ausgewählt werden konnte. Das A-Netz war bis 1977 in Betrieb und wurde durch die ebenfalls analogen Nachfolgesysteme B-Netz und anschließend C-Netz abgelöst.

Mit Beginn der 80er Jahre entstanden dann gleichzeitig in vielen Regionen der Welt zellular aufgebaute Mobiltelefonnetze, die jedoch alle zueinander inkompatibel waren und sowohl aufgrund der Anschaffungskosten der Endgeräte als auch der Gesprächsgebühren nur für eine zahlungskräftige Klientel ausgerichtet waren. Diese Netze werden heute als

¹ Einer nicht näher rekonstruierbaren Anekdote zufolge waren die 1863 von Johann Philip Reis ersten mit einem Telefon übertragenen Worte „Das Pferd frisst keinen Gurkensalat.“. Diesen etwas merkwürdigen Satz wählte er vermutlich deshalb, um sicherzustellen, dass die Übertragung der menschlichen Sprache auch wirklich funktionierte und die Person an der Gegenstelle die übertragenen Worte nicht raten konnte, was beispielsweise bei dem heute üblichen „Hello World“ leicht möglich gewesen wäre.

² Einen guten und komprimierten Überblick über die Geschichte der historischen Techniken zur Nachrichtenübermittlung finden Sie beispielsweise in dem Buch *Nachrichtentechnik* von Oskar Blumtritt [Blumtritt 97].

Mobilfunknetze der ersten Generation (1G) bezeichnet. Sie waren bereits zellenbasiert, die Datenübertragung auf der Luftschnittstelle wurde jedoch analog durchgeführt. Durch personalisierte Mobiltelefone wurden die Teilnehmer identifiziert, d. h. ein Mobiltelefon war jeweils einem bestimmten Teilnehmer fest zugeordnet. Als eines der ersten 1G-Systeme unterstützte das deutsche C-Netz, auch als C-450 bezeichnet, den Einsatz von Karten. In den ersten C-Netz Mobiltelefonen waren dies noch Magnetstreifenkarten, die jedoch sehr schnell durch Chipkarten ersetzt wurden. Damit war Endgerät und Teilnehmerschaft getrennt, was die damals noch übliche Personalisierung von Endgeräten überflüssig machte. Dies führte dazu, dass die im Verhältnis zur Chipkarte sehr teuren Endgeräte untereinander problemlos austauschbare Massenprodukte wurden. Aufgrund der dann einsetzenden starken Konkurrenz führte dies zu einem rasanten Preisverfall der Mobiltelefone, was wiederum im Interesse der Netzbetreiber hinsichtlich eines möglichst großen potenziellen Marktes liegt.

Die heterogenen Mobilfunksysteme der ersten Generation in Europa führten schließlich Anfang der achtziger Jahre zu dem Wunsch der europäischen Postbehörden, die vielen unterschiedlichen landesspezifischen Systeme zu vereinheitlichen. Die Absicht dahinter war, dass nur dann die Benutzung eines Mobiltelefons auch über Landesgrenzen hinweg technisch möglich werden würde und vor allem der Preis für europaweit einheitliche Systemkomponenten und Endgeräte durch die erhebliche größere Stückzahl stark sinken würde.

Dies führte nach einigen Jahren Spezifikationsarbeit schließlich zu dem weltweiten Mobilfunkstandard GSM (*Global System for Mobile Communications*).³ Nachdem die ersten GSM-Systeme 1991 ihren Testbetrieb und 1992 ihren Wirkbetrieb aufgenommen und auch sehr schnell die ursprünglichen Grenzen von Europa überschritten hatten, existierten weiterhin parallel dazu in vielen Gegenden der Welt noch andere zu GSM inkompatible Mobilfunksysteme. Die ITU (*International Telecommunications Union*) [ITU] begann vorausschauend aus diesem Grund bereits im Jahr 1985 mit Arbeiten zur weltweiten Vereinheitlichung und Funktionalitätserweiterung aller bestehenden Mobilfunksysteme. Dieses Konzept hatte den Namen FPLMTS (*future public land mobile telecommunication service*). Es ließ sich aber in der ursprünglichen weltweit einheitlichen Lösung nicht realisieren, sodass es 1995 in das Konzept IMT-2000 (*international mobile telecommunication at 2000 MHz*) überführt wurde. IMT-2000 lässt deutlich mehr Handlungsspielraum bei der Umsetzung und ist mittlerweile die Grundlage für das heutige UMTS und andere Mobilfunksysteme der dritten Generation.

Es ist weltweit üblich, die Technologie von Mobilfunknetzen mit einer Generationennummer zu systematisieren. Die Zählung der Generationen beginnt bei eins und umfasst ausschließlich zellular aufgebaute Netze. Frühe Vertreter von Mobilfunknetzen wie das deutsche A- oder B-Netz wären demnach Generation null, eine Bezeichnung, die aber unüblich ist. Als erste Generation (Abkürzung 1G) werden zellular aufgebaute Mobilfunknetze mit analoger Luftschnittstelle bezeichnet. Typische Vertreter für 1G Netze sind AMPS und das deutsche C-Netz. Unter der zweiten Generation (2G) versteht man zellular aufgebaute Mobilfunknetze mit digitaler Datenübertragung auf der Luftschnittstelle. Die beiden verbreitetsten 2G-Systeme sind GSM und CDMA. Funktionale Erweiterungen von GSM wie GPRS (*general packet radio system*) und EDGE (*enhanced data rates for GSM*

³ Ein Überblick zur Entstehungsgeschichte von GSM findet sich in Abschnitt 13.2 „Das GSM-System“.

and TDMA evolution), die in Richtung der dritten Generation führen, werden üblicherweise als 2,5 G Systeme bezeichnet. Die dritte Generation (3G) umfasst ebenfalls zellular aufgebaute Mobilfunknetze mit digitaler Luftschnittstelle, wobei diese gegenüber 2G-Systemen über erhebliche Erweiterungen hinsichtlich mobiler Datenkommunikation und internetkonformen Diensten verfügen. Typische 3G-Systeme sind UMTS und CDMA 2000. Beide gehören wiederum zur Familie von IMT-2000.

Details mit Relevanz für Chipkarten bei der Entstehung von GSM und UMTS-Mobilfunksystemen finden sich in den jeweiligen Kapitel über das GSM-System bzw. UMTS-System.

Mittlerweile sind bereits erste Bestrebungen für Konzepte von Mobilfunknetzen der vierten Generation (4G) im Gange. Zukünftige Telekommunikationssysteme werden mit Sicherheit eine deutlich höhere Bandbreiteneffizienz haben als heutige Systeme, da Frequenzspektren ein begrenztes Gut und auch in vielen Ländern bedingt durch die Methode der Versteigerungen auch ein sehr teures Gut sind. Im GSM-System wird für die Übertragungsbandbreite von 1 Bit/s etwa 5 Hz Frequenzbandbreite benötigt, woraus eine Frequenzeffizienz von 0,2 Bit/s pro Hz resultiert. Im UMTS-System ist dies bereits auf etwa 1 Bit/s pro Hz verbessert und erste Systeme im Forschungsstadium arbeiten bereits mit bis zu 30 Bit/s pro Hz.

Es existieren viele weitere Ideen und Vorschläge für 4G-Systeme, die jedoch noch keinen beständigen technischen Stand bzw. eine einheitliche Entwicklungsrichtung erreicht haben. Die beiden Hauptrahmenbedingungen dafür sind aber in ihren Grundzügen bekannt: weltweite Interoperabilität der Endgeräte und bei Bedarf hohe Datenübertragungsraten. Interessanterweise sind dies zwei Wünsche, die bereits Mitte der achtziger Jahre bei den Konzepten für FPLMTS bzw. IMT-2000 als wichtige Anforderungen benannt wurden.

Tabelle 13.1 Die wichtigsten Mobilfunksysteme mit einigen technischen Rahmenwerten. Ein mögliches Unterscheidungskriterium der Mobilfunksysteme sind die realen Datenübertragungsraten in Abhängigkeit von der Generation. Bei 2G-Systemen bewegt sich diese um 10 kBit/s, bei 2,5G-Systemen im Bereich von 64 – 144 kBit/s und bei 3G-Systemen von 384 – 2000 kBit/s.

System	Genera- tion	üblicher Frequenz- bereich	Luftschnitt- stelle	Besonderheiten
AMPS	1	≈ 850 MHz (824 – 894 MHz)	analog, FDMA	keine Chipkarte, Datenübertragungsrate 2,4 kBit/s
C-Netz (C-450)	1	≈ 450 MHz (450 – 466 MHz)	analog, FDMA	mit Chipkarte, Datenübertragungsrate 2,4 kBit/s
GSM	2	900 MHz, (890 – 960 MHz) 1800 MHz (1710 – 1880 MHz), 1900 MHz	digital, TDMA mit FDMA	mit Chipkarte (SIM), Datenübertragungsrate 9,6 – 14,4 kBit/s
D-AMPS (IS-54)	2	≈ 850 MHz (824 – 894 MHz)	TDMA	keine Chipkarte, Weiterentwicklung von AMPS, Datenübertragungsrate 8 kBit/s

Tabelle 13.1 (Fortsetzung)

System	Genera- tion	üblicher Frequenz- bereich	Luftschnitt- stelle	Besonderheiten
CMDA (IS-95)	2	≈ 850 MHz (824 – 894 MHz)	CDMA	keine Chipkarte
GPRS, EDGE	2,5	siehe GSM	siehe GSM	Erweiterung bestehender GSM-Netze für paketori- entierete Dienste, Daten- übertragungsrate bis 384 kBit/s
UMTS	3	2 000 MHz (1900 – 2170 MHz)	digital, WCDMA	mit Chipkarte (USIM), Datenübertragungsrate bis 2 MBit/s
CMDA 2000	3	2 000 MHz	digital, CDMA	optional mit Chipkarte (R-UMI), Datenübertra- gungsrate bis 2 MBit/s

13.1 Überblick zu Mobilfunksystemen

Dieser Abschnitt gibt einen technischen Überblick zu aktuellen Mobilfunksystemen, soweit er zum Verständnis für den Einsatz von Chipkarten in diesem Bereich notwendig ist. Bei Jörg Eberspächer et al. [Eberspächer 00], Bernhard Walke [Walke 00] und Raymond Steele et al. [Steele 2001] finden sich wesentlich detailliertere Beschreibungen aller technischen Aspekte der aktuell im Einsatz befindlichen Mobilfunknetze.

Es wird hier durchgehend der Begriff Mobilfunksysteme anstelle von Mobiltelefonsystemen verwendet, da bei allen neueren diesbezüglichen Systemen die reine Sprachübertragung mittlerweile nur noch einer von vielen möglichen Diensten ist und die Übertragung von beliebigen Daten zunehmend in den Vordergrund rückt.

13.1.1 Vielfachzugriffsverfahren (*multiple access*)

Die bei Mobilfunksystemen zur Verfügung stehende Bandbreite, auch Frequenzspektrum genannt, ist typischerweise auf einige zehn Megahertz beschränkt. Damit diese begrenzte Bandbreite möglichst vielen Teilnehmern quasi gleichzeitig zu Verfügung gestellt werden kann, müssen so genannte Vielfachzugriffsverfahren eingesetzt werden. Deren Aufgabe ist es, durch geschickte Ausnutzung von Funkübertragungsverfahren und Informationstechnologie einer möglichst großen Anzahl von Mobilstationen in einer Zelle Zugriff in akzeptabler Qualität auf das Netzwerk zu gewähren.

Grundsätzlich gibt es vier unterschiedliche Vielfachzugriffsverfahren. Sie unterscheiden sich sowohl im Realisierungsaufwand als auch in der optimalen Nutzung der zur Verfügung stehenden Bandbreite. Die vier Verfahren sind Frequenzvielfachzugriff (FDMA), Zeitvielfachzugriff (TDMA), Codevielfachzugriff (CDMA) und Raumvielfachzugriff (SDMA). Sie werden im Folgenden überblickshaft beschrieben.

Frequenzvielfachzugriff (FDMA – *frequency division multiple access*)

Beim Frequenzvielfachzugriffsverfahren erhält jeder Sender ein für ihn reserviertes Frequenzband aus dem gesamten zur Verfügung stehenden Frequenzbereich. In diesem Frequenzband darf der Sender kontinuierlich exklusiv senden. Damit arbeitet innerhalb einer Zelle jeder Sender auf einer unterschiedlichen Frequenz. Dies ist im Übrigen auch das gebräuchliche Verfahren bei herkömmlichen Funkgeräten, bei denen ein gemeinsam genutzter Kanal (d. h. Halbduplex-Verbindung) für die Kommunikation gebraucht wird. Für die beim Telefonieren übliche Vollduplex-Verbindung (d. h. gleichzeitiger Uplink zur Feststation und Downlink zur Mobilstation) sind natürlich in Konsequenz zwei Frequenzkanäle für eine Gesprächsverbindung notwendig. Dieses Verfahren eignet sich wegen des beschränkten technischen Aufwands verhältnismäßig gut für Mobilfunk mit analoger Datenübertragung.

Der Frequenzvielfachzugriff wurde beispielsweise auf der Luftschnittstelle zwischen Fest- und Mobilstation im deutschen C-Netz eingesetzt. Dabei war sowohl für den Downlink als auch für den Uplink ein Frequenzband mit einer Bandbreite von 4,44 MHz reserviert, das in 222 Frequenzkanäle zu je 20 kHz aufgeteilt war.

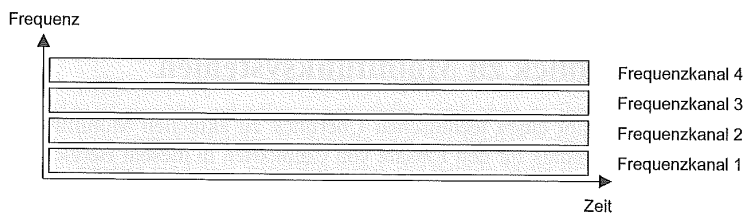


Bild 13.1 Zeit-Frequenz-Diagramm des Frequenzvielfachzugriffsverfahrens (FDMA). In diesem Beispiel entspricht jeder Frequenzkanal jeweils einem Übertragungskanal. Die Bereiche zwischen den Frequenzkanälen sind Schutzbänder zur Unterdrückung von Störungen zwischen einzelnen Frequenzbändern.

Zeitvielfachzugriff (TDMA – *time division multiple access*)

Beim Zeitvielfachzugriff werden die Daten quasi parallel von mehreren Sendern zu einem Empfänger auf einer Frequenz übertragen. Dazu erhält jeder Sender einen bestimmten Zeitschlitz (*time slot*), in dem er exklusiv, aber diskontinuierlich senden darf. Im GSM-System beträgt beispielsweise der zur Verfügung stehende Zeitschlitz für einen Signalmurk 577 μs ($= 15/26 \text{ ms}$), wobei der in diesem Zeitrahmen zu sendende Signalmurk eine Zeitdauer von 546 μs beansprucht. Die Differenz von 31 μs zwischen den beiden Werten wird als Schutzzeit zur Kompensation von kleineren zeitlichen Abweichungen verwendet. Die notwendige exakte Einhaltung des Zeitschlitzes erfordert eine sehr genaue und technisch aufwändige Synchronisation zwischen Sender und Empfänger. Zusätzlich sind beim Zeitvielfachzugriff die Signallaufzeiten zwischen Sender und Empfänger zu berücksichtigen. Beispielsweise beträgt der Laufzeitunterschied des Signals zwischen einer Mobilstation unmittelbar neben der Feststation und in 30 km Abstand ungefähr 100 μs . Diese Laufzeitunterschiede müssen in der Praxis durch verfrühtes Senden ausgeglichen werden, sodass die von der Mobilstation gesendeten Signale jeweils zeitgenau in dem für sie reservierten Zeitschlitz bei der Feststation eintreffen.

Diese durch zeitversetztes Senden auszugleichenden Laufzeitunterschiede sind im Übrigen auch der Grund für den maximalen Durchmesser einer Zelle im GSM-System. Die maximal vorgesehene Zeitspanne für den Ausgleich von Signallaufzeiten zwischen Basisstation und Mobilstation beträgt $116,3 \mu\text{s}$. Dies ist die maximal mögliche Zeit zum verfrühten Senden, sodass das Signal anschließend noch pünktlich innerhalb des festgelegten Zeitschlitzes beim Empfänger eintrifft. Daraus resultiert ein maximaler Zellenradius im GSM-System von etwa 35 km. Das verfrühte Senden bezeichnet man auch als *timing advance*.

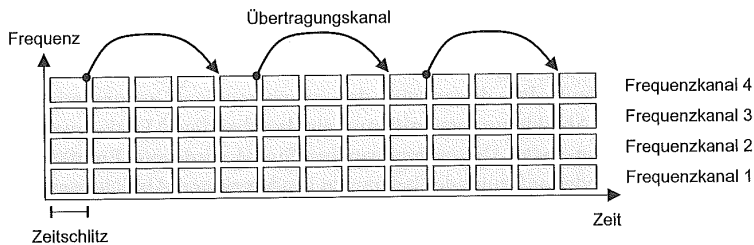


Bild 13.2 Zeit-Frequenz-Diagramm des Zeitvielfachzugriffsverfahrens (TDMA). In diesem Beispiel enthält ein Frequenzkanal jeweils vier Übertragungskanäle. Die grauen Rechtecke symbolisieren jeweils einen Signalburst. Die Bereiche zwischen den Frequenzkanälen bzw. Zeitschlitz sind Schutzbänder bzw. Schutzzeiten zur Unterdrückung von Störungen zwischen einzelnen Signalbursts.

Um die Auswirkungen von frequenzselektiven Störungen zu reduzieren, kann man das Zeitvielfachzugriffsverfahren mit einem Frequenzsprungverfahren (*frequency hopping*) kombinieren. Dabei wechseln Sender und Empfänger nach jedem Zeitschlitz in einer festgelegten Reihenfolge den Frequenzkanal, sodass Störungen in bestimmten Frequenzbereichen mit hoher Wahrscheinlichkeit nur einen einzelnen Signalburst beeinflussen. Diese Störungen können dann mit fehlerkorrigierenden Übertragungs-codes in vielen Fällen ausgeglichen werden.

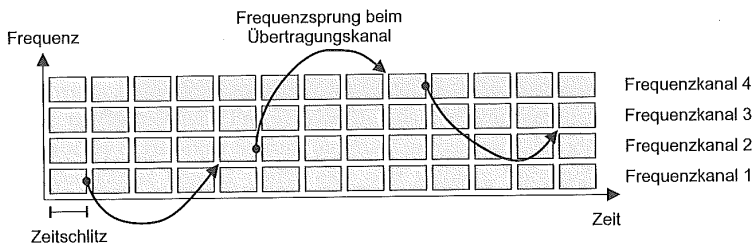


Bild 13.3 Zeit-Frequenz-Diagramm des Zeitvielfachzugriffsverfahrens (TDMA) mit Frequenzsprung. In diesem Beispiel enthält ein Frequenzkanal jeweils vier Übertragungskanäle. Die grauen Rechtecke symbolisieren jeweils einen Signalburst. Die Bereiche zwischen den Frequenzkanälen bzw. Zeitschlitz sind Schutzbänder bzw. Schutzzeiten zur Unterdrückung von Störungen zwischen einzelnen Signalbursts.

Der Zeitvielfachzugriff wird in Kombination mit dem Frequenzvielfachzugriff beispielsweise im GSM-System auf der Luftschnittstelle zwischen Fest- und Mobilstation verwendet. Dabei wird das zur Verfügung stehende Frequenzband von 25 MHz in 124 einzelne Kanäle mit je 200 kHz Bandbreite aufgeteilt. Jeder dieser Frequenzkanäle enthält wiederum 8 Gesprächskanäle. Damit können acht Mobilstationen gleichzeitig auf einem Frequenzkanal senden. Jeder Mobilstation steht dabei der Frequenzkanal alle 4,615 ms für die Zeitdauer von 0,577 ms zur Verfügung.

Codevielfachzugriff (CDMA – *code division multiple access*)

Der Codevielfachzugriff ist ein Vielfachzugriffsverfahren, bei dem von mehreren Sendern gleichzeitig Daten zu einem Empfänger innerhalb des gesamten zur Verfügung stehenden Frequenzspektrums übertragen werden. Die Grundlage des Codevielfachzugriffs ist dabei die Bandspreiztechnik. Bei dieser wird das schmalbandige Ursprungssignal durch eine senderspezifische Abbildungsvorschrift auf ein breitbandiges Funksignal gespreizt und dann breitbandig gesendet. Dieses breitbandige Signal wird vom Empfänger empfangen und kann dann wieder aufgrund der bekannten senderseitigen Abbildungsvorschrift in das schmalbandige Ursprungssignal transformiert werden. In der Ausprägung WCDMA (*wideband code division multiple access*) findet Uplink und Downlink in zwei voneinander unterschiedlichen Frequenzbereichen statt, deshalb wird diese CDMA-Variante oft auch als FD/CDMA (*frequency division/code division multiple access*) bezeichnet. Bei der Ausprägung TD/CDMA (*time division/code division multiple access*) wird die Trennung zwischen Uplink und Downlink durch die Benutzung unterschiedlicher Zeitschlitze realisiert.

Das Codevielfachzugriffsverfahren hat den Vorteil, dass es sehr unempfindlich gegenüber frequenzselektiven Störungen ist. Es gewährleistet auch einen schwachen Schutz vor unberechtigtem Abhören, falls die jeweilige senderindividuelle Abbildungsvorschrift dem Angreifer nicht bekannt ist.

Das Codevielfachzugriffsverfahren CDMA wird im UMTS-System in der Ausprägung WCDMA eingesetzt. Dort wird jeweils eine Bandbreite von 5 MHz für Uplink/Downlink benutzt.

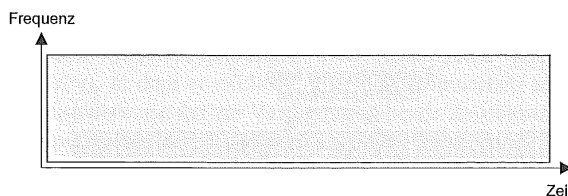


Bild 13.4 Zeit-Frequenz-Diagramm des Codevielfachzugriffsverfahrens (CDMA). Innerhalb des grauen Rechtecks des zur Verfügung stehenden Frequenzspektrums senden gleichzeitig mehrere Sender mit einem senderindividuell gespreizten Signal, das vom Empfänger wieder in das Ursprungssignal zurückgewandelt werden kann.

Raumvielfachzugriff (SDMA – *space division multiple access*)

Der Raumvielfachzugriff ist ein Vielfachzugriffsverfahren zur parallelen Datenübertragung von mehreren Sendern zu einem Empfänger auf einer Frequenz. Dazu verwenden die Sender richtungsselektive (adaptive) Antennen, die auf den jeweiligen Empfänger ausge-

richtet sind. Der dazu notwendige technische Aufwand ist relativ hoch und wird derzeit im Mobilfunkbereich bei den Feststationen nur verhalten eingesetzt. Die richtungsselektiven Antennen sind üblicherweise Antennen-Arrays, die eine so genannte elektronische Strahl-schwenkung durchführen können. Damit ist es nicht notwendig, Antennen physisch auf Empfänger auszurichten. Der Raumvielfachzugriff kann grundsätzlich mit den anderen Vielfachzugriffsverfahren kombiniert werden, wird aber zurzeit im Mobilfunkbereich aufgrund des ungünstigen Kosten-Nutzen-Verhältnisses selten verwendet.

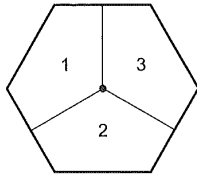


Bild 13.5 Schematische Darstellung des Raumvielfachzugriffsverfahrens (SDMA) anhand einer einzelnen Funkzelle. In der Mitte befindet sich die Basisstation mit drei um 120 Grad versetzt angeordneten Antennen, die jeweils überschneidend ein Drittel der gesamten Zelle funktchnisch abdecken.

13.1.2 Zellulartechnik

Das für ein Mobilfunksystem zur Verfügung stehende Frequenzband muss für eine sehr große Anzahl von Mobilstationen ausreichen. Es ist also nicht damit getan, eines oder mehrere Vielfachzugriffsverfahren einzusetzen, um damit ein größeres Gebiet mit vielen potenziellen Gesprächsteilnehmern versorgen zu wollen.

Am Beispiel des GSM-Systems sei dies nochmals kurz anhand einiger Eckdaten verdeutlicht: Aufgrund der für GSM zur Verfügung stehenden Bandbreite und des verwendeten Vielfachzugriffsverfahrens kann man berechnen, dass in der Theorie ungefähr maximal 1000 Teilnehmer gleichzeitig telefonieren könnten (124 Kanäle zu je 8 Zeitschlitten = 992 Gesprächskanäle). In der Praxis sind es deutlich weniger, da einige Kanäle beispielsweise für Signalisierungszwecke genutzt werden müssen.

Um nun mehreren Millionen Teilnehmern ein gleichzeitiges Gespräch zu ermöglichen, organisiert man das gesamte Mobilfunksystem eines Netzbetreibers in Zellen und weist jeder dieser Zellen eine bestimmte Anzahl von Frequenzkanälen zu. Dieses Verfahren, das um 1970 entwickelt wurde, bezeichnet man als Zellulartechnik. Die jeweils benachbarten Zellen erhalten mehrere unterschiedliche Frequenzkanäle aus dem zur Verfügung stehenden Frequenzspektrum, sodass es zu keiner Störung aufgrund in Nachbarzellen gleichzeitig benutzter identischer Frequenzkanäle kommt. Durch eine oder mehrere Nachbarzellen räumlich getrennte Zellen können dann ohne Störungsproblematik wieder die gleichen Frequenzen nutzen. Diese Frequenzwiederverwendung (*frequency reuse*) ist einer der grundlegenden Eckpfeiler der Zellulartechnik. Ein nützlicher Nebeneffekt der Zellulartechnik ist aber auch, dass sich die notwendige Sendeleistung eines Mobiltelefons mit kleiner werdenden Zellen reduziert, was sowohl Stromverbrauch als auch Elektrosmog des Mobiltelefons senkt.

Mehrere Zellen mit unterschiedlich zugewiesenen Frequenzkanälen lassen sich als so genannte Cluster zusammenfassen. Üblich sind Cluster mit 3, 4, 7, 12 und 21 Zellen. Bei der

Netzwerkplanung wird dann der gesamte durch das Netzwerk abzudeckende Bereich mit diesen Clustern überzogen. Dabei werden einzelne Zellen vereinfacht als Hexagone und nicht als Kreise dargestellt, sodass ein Netzwerkplan einem Bienenwabennmuster gleicht.

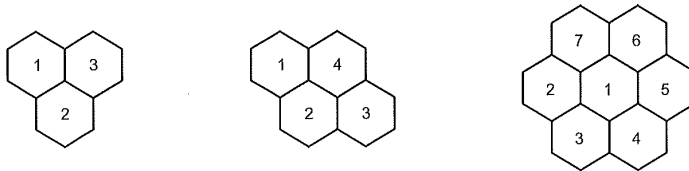


Bild 13.6 Grafische Darstellung von typischen Clustern mit der Zellenanzahl 3, 4 und 7. Die durchnummerierten Zellen besitzen jeweils eine bestimmte Anzahl von jeweils unterschiedlichen Frequenzkanälen.

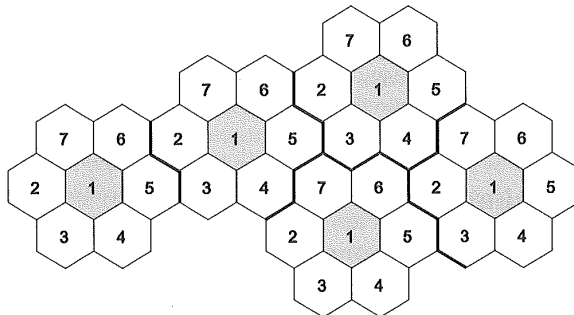


Bild 13.7 Grafische Darstellung der vollständig funktchnischen Abdeckung einer Region mit Zellen unter Verwendung von 7 Zellen pro Cluster. Es ist deutlich zu sehen, dass zwischen allen Zellen mit gleichen Frequenzkanälen mindestens zwei weitere Zellen mit anderen Frequenzkanälen liegen.

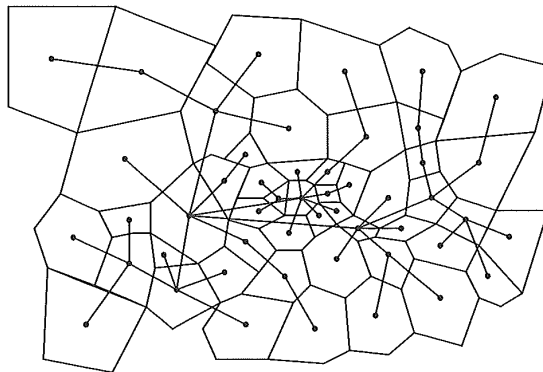


Bild 13.8 Beispiel für einen Ausschnitt aus einem Zellenplan für ein reales Netz. Mit den unterschiedlichen Zellengrößen wird sichergestellt, dass auch bei unterschiedlich hoher Netzlast (verursacht i. d. R. durch hohe Teilnehmerdichten) eine akzeptable Netzabdeckung erreicht wird. In diesem Zellenplan befindet sich beispielsweise im Mittelpunkt ein Verkehrsknotenpunkt mit einer hohen Dichte an Mobiltelefonen.

13.1.3 Zelltypen

Entscheidend für eine kostengünstige Netzinfrastruktur bei bestmöglicher Netzabdeckung sind die verwendeten Zelltypen. Die folgenden Angaben für Zellgrößen sind ungefähre Näherungswerte und sollen vor allem die grundlegende Anwendung der verschiedenen Zelltypen verdeutlichen.

Für die weiträumige Abdeckung von Gebieten mit geringer auftretender Netzlast bei wenig abschirmenden Gelände werden Großzellen mit typischen Durchmessern von 10 km bis 40 km verwendet. Typischerweise findet man sie im ländlichen Bereich mit niedriger Bevölkerungsdichte und flachem Gelände. Makrozellen mit einem typischen Durchmesser von 1 km bis 10 km sind die nächstkleinere Größe. Ihr Einsatz ist beispielsweise im Zentrum kleinerer Siedlungen und in vielen Vorstädten von Ballungsgebieten.

Mikrozellen mit einem Durchmesser von 0,1 km bis 1 km finden sich oftmals an Verkehrsknotenpunkten oder innerstädtischen Gebieten mit hohem Gesprächsaufkommen. Der kleinste Zelltyp wird Pikozone genannt und hat einen Durchmesser von 50 m bis 100 m. Er wird oft innerhalb von Gebäuden, beispielsweise in Lokalen, eingesetzt.

Es gibt als Sonderform der Zelltypen noch die Schirmzelle. Dies ist eine Zelle in der ein oder mehrere weitere Zellen mit geringerer Größe enthalten sind. Die Schirmzelle versorgt die Bereiche, die von den in ihr enthaltenen kleineren Zellen nicht abgedeckt werden können. Dieser Sondertyp einer Zelle kommt beispielsweise bei lokal eng begrenzten hohen Gesprächsaufkommen in einer Gegend mit ansonsten wenig Netzlast zum Einsatz. Die Schirmzelle und die von ihr überdeckten kleineren Zellen müssen selbstverständlich unterschiedliche Frequenzkanäle benutzen.

Als Exempel für die Anwendung einer Schirmzelle sei eine einzelne Skihütte mit Getränke- und Essensverkauf in den Alpen innerhalb eines weiträumigen Skigebietes aufgeführt. Die Skihütte würde man netztechnisch mit einer Pikozone versorgen und das restliche darumliegende Skigebiet mit einer Schirmzelle. Bei manchen Skigebieten wäre es allerdings noch günstig, an den Talstationen jeweils zumindest eine Pikozone zu installieren, damit man sich die Wartezeit am Lift mit Telefonieren vertreiben kann.

Eine weitere Zellsonderform ist die Selektivzelle, die keine symmetrische Kreisfläche von 360° abdeckt, sondern nur einen Ausschnitt eines Kreises. Dieser Zelltyp wird beispielsweise vor Unterführungen montiert und kann durch die dem Lichtstrahl eines Scheinwerfers ähnliche Zellenform sehr gut von außen die Unterführung mobilfunktechnisch erschließen. Innerhalb von Tunnels sind Selektivzellen ebenfalls das probate Mittel der Wahl.

Um ein Land in der Größe von Deutschland (alle 16 Bundesländer) mit ca. 360 000 km² Fläche für etwa 5 Millionen Teilnehmer vollständig zu versorgen, benötigt man ca. 3000 Basisstationen. Bei einer Teilnehmerzahl von 20 Millionen steigt die Anzahl der Basisstationen auf etwa 12 000 bei nahezu hundertprozentiger Netzabdeckung. Bei einem Netz dieser Größenordnung kommen etwa 60 Mobilvermittlungsstellen (MSC) zum Einsatz. Im Übrigen hatten Mitte 2001 alle vier deutschen GSM-Netze zusammen ca. 52 000 Basisstationen mit 164 000 Zellen.

13.1.4 Trägerdienste (*bearer services*)

Aufgabe der Trägerdienste bei einem Mobilfunksystem ist es, Sprache und Daten zwischen Endgerät und Hintergrundsystem zu übertragen. Zur akustischen Übertragung von Sprache in

der heute üblichen Telefonqualität wird eine minimale Frequenzbandbreite von 3100 Hz benötigt, welche auch als Fernsprechbandbreite bezeichnet wird. Diese Bandbreite ergibt sich daraus, dass die menschliche Sprache vornehmlich den Frequenzbereich zwischen 300 Hz und 3400 Hz nutzt. Im GSM-System wird aus diesem kontinuierlichem Signal ein komprimierter und diskontinuierlicher Datenstrom erzeugt, der im *full-rate* Modus eine Datenübertragungsrate von 13 kBit/s beansprucht. Diese Übertragungsrate ist ausreichend für eine halbduplex Telefonverbindung in üblicher Fernsprechqualität, d. h. für einen uplink oder einen downlink. Sollen beide Gesprächsteilnehmer, wie gewohnt, gleichzeitig sprechen können, so benötigt man die im Telefonbereich übliche Vollduplex-Verbindung, was in der Konsequenz zu einer notwendigen Nettogesamtübertragungsrate in beiden Richtungen von 26 kBit/s führt. In der Praxis ist die notwendige Übertragungsrate durch Steuerungs- und aufwändige Fehlerkorrekturcodes erheblich größer. Im GSM-System beträgt deshalb die notwendige Bruttoübertragungsrate für uplink und downlink ca. 34 kBit/s.

Trägerdienste für Datenübermittlung ermöglichen im Idealfall einen hinsichtlich der beiden übertragenen Instanzen transparenten Datenaustausch, sodass die Kommunikationsprotokolle der dazwischenliegenden Stationen seitens der Endteilnehmer nicht berücksichtigt werden müssen. Im GSM-System werden als Trägerdienste SMS, USSD, CSD, HSCSD und GPRS verwendet. Die Verfügbarkeit dieser Dienste hängt sowohl von der technischen Ausstattung des jeweiligen Endgeräts als auch vom eingebuchten GSM-Netzwerk ab.

SMS (*Short Message Service*)

Wird der Kurznachrichtendienst SMS als Träger verwendet, müssen die zu übertragenden Daten in Pakete von maximal 140 Byte Länge in die maximal 176 Byte langen SMS-Nachrichten eingepackt werden. Die 36 Byte Differenz wird für Übertragungsinformationen benötigt. Der SMS-Dienst gewährleistet bei mehreren aufeinander folgend versendeten Kurznachrichten leider nicht das Eintreffen in der ursprünglichen Reihenfolge. Eine garantierte Übertragungsrate kann bei diesen paketerorientierten Datenübertragungsdiensten nicht angegeben werden, da sie einerseits sehr stark von der aktuellen Netzauslastung abhängt und andererseits auch davon, ob die Mobilstation im Heimatnetz eingebucht ist oder nicht. Typische Laufzeiten für Kurznachrichten bewegen sich von 0,5 Sekunden bis zu 20 Sekunden. Umgerechnet entspricht dies einer Bruttoübertragungsrate zwischen 2,8 kBit/s und 94 Bit/s.

USSD (*unstructured supplementary services data*)

Der Trägerdienst USSD ist ähnlich dem Kurznachrichtendienst SMS beschaffen. Die Hauptunterschiede bestehen darin, dass USSD ein verbindungsorientierter Dienst ist und die maximale Nachrichtenlänge 182 Byte umfasst.

CSD (*circuit switched data*)

Der verbindungsorientierte Trägerdienst CSD bei GSM ist nichts anderes als die Übertragung von Daten anstelle von Sprache auf der Luftschnittstelle. Die Daten werden dazu entsprechend über ein Modem in die Mobilstation eingespeist. Die typischen Bruttoübertragungsraten sind entweder 9,6 kBit/s oder 14,4 kBit/s.

HSCSD (*high speed circuit switched data*)

Mit dem technisch analog CSD aufgebauten verbindungsorientierten Trägerdienst HSCSD nach GSM 02.34 können theoretisch und in der Summe bis zu 76800 Bit/s Bruttodaten im Uplink und Downlink übertragen werden.

GPRS (*general packet radio system*)

Der paketorientierte Trägerdienst GPRS nach GSM 01.60 und GSM 02.60 ermöglicht eine theoretisch maximale Datenübertragungsrate im downlink und uplink bis zu 115,2 kBit/s.

13.2 Das GSM-System

Die Chipkarte im GSM-Mobiltelefon – die SIM (*subscriber identity module*) – war und ist der Vorreiter hinsichtlich Funktionalität und Speichergrößen. Das rührt zum einen daher, dass Chipkarten für in der Herstellung mehrere hundert Euro teure mobile Endgeräte deutlich weniger preissensitiv sind als beispielsweise Chipkarten für den elektronischen Zahlungsverkehr oder Chipkarten im Gesundheitswesen. Ein weiterer die Chipkarten-Technologie bestimmender Faktor ist die allgemein hohe Evolutionsrate des gesamten Telekommunikationsbereiches. Diese technologische und normative Vorreiterrolle, welche die SIM gegenüber allen anderen Chipkarten-Anwendungen derzeit innehat, ist der Grund, weshalb dieses Thema hier ausführlich dargestellt ist.

Das GSM-System wurde seit dem kommerziellen Start im Jahr 1992 innerhalb weniger Jahre der weltweite Standard für Mobilfunk. Dazu gehört sowohl die Übertragung von Gesprächen als auch von Daten, die zurzeit immer noch hauptsächlich in Form von Kurznachrichten (*short message service* – *SMS*) übertragen werden. Mitte des Jahres 2001 waren in 171 Ländern insgesamt 400 Mobiltelefonnetze mit über 565 Millionen Teilnehmern nach der GSM-Norm in Betrieb. Pro Monat wurden mehr als 20 Milliarden Kurznachrichten übertragen.⁴ Mobilfunknetze nach der GSM Norm haben manchmal noch landes-typische Bezeichnungen. In Deutschland beispielsweise werden die vier in Betrieb befindlichen GSM-Netze D-Netz (900 MHz und 1800 MHz GSM-Varianten) und E-Netz (1800 MHz-Variante) genannt, in Österreich wird GSM zum Teil auch als A-Netz bezeichnet.

Die Spezifikation des GSM-Systems startete 1982 im Rahmen von CEPT (*Conférence Européenne des Postes et Télécommunications*). Die Aufgabe war die Erstellung einer Spezifikation für ein länder- und betreiberübergreifendes Mobiltelefonnetz. Im Laufe der Zeit kristallisierte sich dann heraus, dass ein länder- und betreiberübergreifendes digitales zelluläres, im 900 MHz-Band arbeitendes und ISDN-kompatibles Mobilfunknetz zu spezifizieren war. Dazu wurde eine „Groupe Spécial Mobile“ gegründet, und somit war auch die ursprüngliche Abkürzung GSM geboren. 1986 wurde dann für die Koordination der Spezifikationserstellung ein so genannter „GSM Permanent Nucleus“ mit Sitz in Paris ins Leben gerufen, der später auch für die verschiedensten Tests von Systemkomponenten verantwortlich war. Aus technischer Sicht interessant ist, dass damals für GSM einige Technologien ausgewählt wurden, die völlig neu und in der Praxis unerprobt waren. So war die Luftschnittstelle mit Zeitvielfachzugriff in Kombination mit Frequenzvielfach-

⁴ Einen guten Überblick zu den jeweils aktuellen statistischen Zahlen und zu allen Netzbetreibern findet sich bei GSM World [GSM].

zugriff und digitaler Datenübertragung für großräumige Mobilfunkanwendungen völliges Neuland. Diese Entscheidungen führten vor allem in der Systemaufbauphase zu vielen technischen Problemen, aus heutiger Sicht stellt sich dies jedoch als Glücksgriff heraus, da mit GSM ein zukunftsfähiges System ohne technischen Ballast aus den Anfangszeiten des Mobilfunks geschaffen wurde.

Die gemeinsame vertragsrechtliche Grundlage aller GSM-Netzbetreiber ist das so genannte Memorandum of Understanding (MoU), das erstmals 1987 von 15 europäischen Netzbetreibern unterzeichnet wurde. Die GSM Association ist ein weltweit agierendes Gremium mit Sitz in Dublin und London zur Abstimmung von Mobilfunksystemen. Sie wurde 1987 in Kopenhagen gegründet und ist verantwortlich für die Entwicklung und den Einsatz der GSM-Normen. Die GSM Association repräsentiert über 500 Netzbetreiber, Hersteller und Zulieferer der GSM-Industrie. 1989 wurden die unter Leitung des GSM Permanent Nucleus entstandenen Spezifikationen und die dazugehörigen Arbeitsgruppen bei dem damals neu gegründeten ETSI (*European Telecommunications Standards Institute*) eingegliedert und werden seitdem dort weitergeführt. Im Jahr 1990 waren alle Spezifikationen von GSM Phase 1 in akzeptabler Form fertiggestellt und wurden eingefroren.

Mit der Spezifikation der Chipkarte für GSM, das SIM (*subscriber identity module*), wurde im Januar 1988 in der Expertengruppe SIMEG (*subscriber identity module expert group*) begonnen. Sie setzte sich aus Vertretern von Karten-, Mobiltelefonherstellern und Netzbetreibern zusammen. Die SIMEG erstellte im Rahmen von ETSI die Spezifikation für die Schnittstelle zwischen Chipkarte und Mobiltelefon, diese Spezifikation trägt den Namen GSM 11.11. Unter Beibehaltung der Aufgaben und Kompetenzen wurde 1994 aus der SIMEG die SMG9 (*special mobile group 9*) gegründet, welche bis 2000 das Mandat der Weiterentwicklung und Pflege aller SIM-Spezifikationen innehatte. Die SMG9 wurde dann im Jahr 2000 aufgelöst und ihre Aufgaben auf zwei neu gegründete Expertengruppen verteilt. Alle generischen Themen im Bereich von Chipkarten für die Telekommunikation werden von der ETSI-Expertengruppe EP SCP (*ETSI project smart card platform*) bearbeitet. Für die applikationsspezifische Schnittstelle Mobiltelefon zu SIM bzw. USIM ist die 3GPP Expertengruppe T3 verantwortlich.⁵

Erstmals wurde im Jahr 1991 auf der ITU Telekommunikationsmesse in Genf ein in Betrieb befindliches GSM-Netzwerk gezeigt, bei dem auf der Messe ca. 11 000 Verbindungen ohne größere Probleme geschaltet wurden.

1992 wurden dann in mehreren europäischen Ländern (Dänemark, Finnland, Frankreich Deutschland, Italien, Portugal und Schweden) die ersten GSM-Systeme in den Echtbetrieb überführt. Damals waren es etwa 250 000 Teilnehmer. In diesem Jahr wurde auch das erste Roaming Abkommen zwischen zwei Netzbetreibern unterzeichnet, und der erste Nicht-Europäische Netzbetreiber unterzeichnete das MoU, d. h. er entschied sich offiziell für das GSM-System. Bereits ein Jahr später, Ende 1993, wurde der einmillionste Teilnehmer gezählt. In diesem Jahr startete das erste GSM-1800-Netzwerk in Großbritannien und das erste Roaming-Abkommen kam zustande. 1995 nahm dann das erste GSM-1900-Netzwerk seinen Betrieb in den USA auf, und Ende Juli 1998 wurde der 100millionste GSM-Teil-

⁵ Ein ausführlicher Überblick über die interessante Geschichte der Expertengruppen zur Normung von SIM, USIM und UICC findet sich bei Klaus Vedder „The Subscriber Identity Module Past – Present – Future“ in [Hillebrand 2002].

nehmer gezählt. Mitte 2001 gab es weltweit 500 Millionen Teilnehmer und man rechnet mit 1 Milliarde Teilnehmer im Jahr 2005.

Die GSM-Spezifikationen wurden in den Jahren 1991 bis 1992 erweitert und decken mit GSM 1800 (früher DCS – *digital cellular system*) auch den 1800 MHz Frequenzbereich (uplink: 1710 – 1785 MHz, downlink: 1805 – 1880 MHz, Wellenlänge ca. 16,6 cm) und mit GSM 1900 (früher PCS – *personal communication system*) den 1900 MHz Frequenzbereich (uplink: 1850 – 1910 MHz, downlink: 1930 – 1990 MHz, Wellenlänge ca. 15,8 cm) ab. GSM im ursprünglichen 900 MHz Frequenzband (uplink: 880 – 915 MHz, downlink: 925 – 960 MHz, Wellenlänge ca. 33,3 cm) wird seitdem als GSM 900 bezeichnet. Aufgrund der höheren Frequenz und einer geringeren Sendeleistung beträgt der Durchmesser einer Zelle bei diesen höherfrequenten Systemen nur maximal 20 km. Das Einsatzgebiet fokussiert sich aufgrund dessen vor allem auf Gebiete mit hoher Teilnehmerdichte und weniger auf Gegenden mit geringer Teilnehmerdichte. Der hauptsächliche Unterschied zwischen dem ursprünglichem GSM im 900 MHz-Frequenzband und GSM 1800 bzw. GSM 1900 sind die Sende- und Empfangsteile beiderseits der Luftschnittstelle (*air interface*).

Die niedrigen Datenübertragungsraten im GSM-System von 9600 Bit/s bzw. mit einem verbesserten Codec auf der Luftschnittstelle von 14400 Bit/s haben schon verhältnismäßig früh eine Schwachstelle des Systems zu Tage gebracht. Der stark wachsende Bedarf der mobilen Teilnehmer an zu übertragenden Datenmengen hat dieses Manko noch weiter verschärft. Aus diesem Grund wurde im GSM-System ein Evolutionspfad für höhere Übertragungsraten und vor allem auch für paketorientierte Übertragungsdienste spezifiziert. Die nächste Entwicklungsstufe von GSM ist das verbindungsorientierte HSCSD (*high speed circuit switched data*). Mit HSCSD-Technik lassen sich durch zusätzliche Nutzung vorhandener Zeitslots der Luftschnittstelle durch Kanalbündelung eine theoretische Datenübertragungsrate von bis zu $8 \cdot 9600$ Bit/s (76800 Bit/s) (Uplink und Downlink) erreichen. Bestehende GSM-Netze lassen sich mit relativ geringem Aufwand durch Erweiterungen in den Basisstationen sowie spezielle Mobiltelefone auf HSCSD erweitern. Der Nachteil ist jedoch, dass der Bedarf an Übertragungskanälen maximal bis auf das 8-fache steigt, weshalb HSCSD wohl keine große Zukunft beschert ist.

Das paketorientierte GPRS (*general packet radio system*) ist der nächste Schritt in der Evolution von GSM. GPRS bietet eine paketorientierte Verbindung mit einer Datenübertragungsrate bis zu 115,2 kBit/s (downlink und uplink) durch Bündelung der 8 vorhandenen Zeitschlitze zu je 14400 Bit/s. Ein Mobiltelefon mit GPRS-Technologie ist hinsichtlich der Datenübertragung ständig im Netz eingebucht und damit zu jedem Zeitpunkt für Datenübertragungen verfügbar. Aus diesem Grund eignet sich GPRS auch sehr gut für diskontinuierliche Datenübertragungen. Von Nachteil ist der verhältnismäßig große Aufwand im Bereich der Feststationen. GPRS wird auch als 2,5 G-Technologie bezeichnet und hat gute Aussichten, ein wesentlicher Faktor für die Verlängerung der Lebensdauer bestehender GSM-Systeme zu werden.

Als letzte Ausbaustufe von GSM-Netzwerken ist EDGE (*enhanced data rates for GSM and TDMA evolution*) vorgesehen. Unter Beibehaltung der existierenden Netzinfrastruktur können durch ein anderes Modulationsverfahren auf der Luftschnittstelle GSM-Mobiltelefone nach EDGE-Standard mit einer Datenübertragungsrate von bis zu 384 kBit/s an Basisstationen angebunden werden. Inwieweit EDGE in Zukunft als Konkurrenztechnolo-

gie zu den dann existierenden 3G-Systemen wie UMTS ein Rolle spielen wird, ist zurzeit noch nicht absehbar.

Der designierte Nachfolger von GSM ist UMTS, das in seiner Grundarchitektur wiederum auf GSM aufbaut, also keine grundsätzlich neue Mobilfunktechnologie wie seinerzeit GSM darstellt.

13.2.1 Die Spezifikationen

Um das GSM-System technisch vollständig zu beschreiben, bedarf es einer Vielzahl von aufeinander abgestimmten und voneinander abhängigen Spezifikationen. In der Summe sind es ungefähr 130 Einzelspezifikationen mit einem Gesamtumfang von über 6000 Seiten.

Gerade im Zusammenhang mit dem GSM-System werden oft die beiden Termini Spezifikation und Norm durcheinandergewürfelt. Im Falle GSM treffen jedoch tatsächlich beide Bezeichnungen zu. Da der Herausgeber das Normungsinstitut ETSI ist, haben die Papiere formal den Status einer Norm. Gleichzeitig sind sie in ihrer technischen Beschreibung aber so strikt, dass quasi alle auf sie basierenden Implementierungen zueinander kompatibel sind. Dies ist jedoch eine typische Eigenschaft einer Spezifikation. Im Folgenden wird in der Regel der in diesem Fall eindeutige Begriff Spezifikation gebraucht. Aus diesem Grund wird in diesem Buch durchgängig auf das in Fachkreisen übliche GSM-Nummernschema (z. B. GSM 11.11) referiert und nicht auf die Referenznummern der entsprechenden und inhaltlich absolut identischen ETSI-Normen (z. B. TS 100977).

Die Weiterentwicklung des GSM-Systems wird durch aufeinander aufbauende Phasen beschrieben. In Phase 1 wurden von 1992 an neben anderen die Basisdienste Sprachübertragung, Rufweiterleitung, Roaming und Kurznachrichtendienste (SMS) realisiert. In Phase 2 kamen ab 1996 zu den Diensten der Phase 1 unter anderem die Dienste Konferenzschaltung, Gesprächsweitergabe, Rufnummernübermittlung und GSM im 1800 MHz Frequenzband hinzu. Anschließend wurden diese Dienste in Phase 2+ u. a. mit der Funktionalität des SIM Application Toolkit, HSCSD (*high speed circuit switched data*) und GPRS (*general packet radio system*) ergänzt.

Die GSM-Spezifikationen führen wie bei Spezifikationen üblich ein eigenes Fachvokabular ein. Dieses ist in diversen Abkürzungsverzeichnissen und Glossars technisch genau definiert und hat auch nur im Bereich von GSM ihre Gültigkeit. Einen Überblick gibt hier die GSM 1.04 (Abbreviations and acronyms). Auf Grund dieses Fachvokabulars ist der Einstieg für den Neuling am Anfang in aller Regel etwas mühselig, da er beim Studium der Spezifikationen in den entsprechenden Abschnitten laufend die Erklärung der Abkürzungen nachschlagen muss.

Die Grundlage für das GSM-Sicherheitsmodul im Endgerät hat die Bezeichnung GSM 02.17 (SIM Functional Characteristics) und beinhaltet eine verhältnismäßig abstrakte Beschreibung der funktionalen Anforderungen an eine SIM. Auf dieser Spezifikation baut das wichtigste chipkartenspezifische Papier im GSM-System, die GSM 11.11 (Specification of the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface), auf. In ihr ist auf über 170 Seiten die Schnittstelle zur SIM exakt und eindeutig spezifiziert. Es handelt sich dabei um eine reine Schnittstellenspezifikation, die keine Vorgaben hinsichtlich der eigentlichen Implementierung macht.

Die Spezifikation der elektrischen Parameter für Chipkarten in 3-Volt-Technologie und 1,8-Volt-Technologie finden sich ergänzend zur GSM 11.11 in GSM 11.12 (Specification of the 3 Volt Subscriber Identity Module – Mobile Equipment (SIM – ME) interface) und GSM 11.18 (Specification of the 1.8 Volt Subscriber Identity Module – Mobile Equipment (SIM – ME) interface).

Neben diesen vor allem die Basisfunktionalität der SIM erläuternden Spezifikationen existiert noch die GSM 11.14 (Specification of the SIM Application Toolkit for the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface), welche die Beschreibung einer Plattform für sichere Mehrwertdienste auf der SIM enthält. Die Bezeichnung dafür ist SIM Application Toolkit (SAT). Sie wurde 1996 veröffentlicht und bietet vor allem Netzbetreibern die Möglichkeit, eigene Anwendungen zur Steuerung des Mobiltelefons auf die Chipkarte zu laden. In der GSM 11.14 ist im Detail festgelegt, wie beispielsweise die Ansteuerung des Displays, die Abfrage der Tastatur, das Versenden von Kurznachrichten (*short message service – SMS*) und weitere Funktionen im Zusammenhang mit einer entsprechenden Zusatzanwendung auf der SIM durchgeführt werden müssen.

In der Anforderungsspezifikation GSM 02.48 (Specification of security mechanisms for the SIM application toolkit, stage 1) und der darauf aufbauenden Spezifikation GSM 03.48 (Specification of security mechanisms for the SIM application toolkit, stage 2) werden zwei wesentliche Sicherheitsmechanismen für die SIM eingeführt. Als ersten Inhaltspunkt werden Sicherheitsmechanismen für eine abhör- und manipulationssichere Ende-zu-Ende-Kommunikation zwischen Hintergrundsystem und SIM festgelegt, welche in der Praxis vor allem zur sicheren Datenübertragung über die Luftschnittstelle (*over the air – OTA*) eingesetzt wird. Als zweiten Inhaltspunkt enthält die GSM 03.48 die Beschreibung des grundsätzlichen Mechanismus eines Remote File Management (RFM) und Remote Applet Management. Die Beschreibung ist vom Ansatz her trägerunabhängig, doch in der GSM 03.48 am Beispiel der Übertragung mittels SMS aufgezeigt.

Chipkarten mit Java haben gerade im Telekommunikationsumfeld sehr schnell Fuß gefasst, weshalb dies auch verhältnismäßig früh Niederschlag in den GSM-Spezifikationen gefunden hat. Die Grundlage für alle Chipkarten-Betriebssysteme mit ausführbarem Programmcode ist die GSM 02.19. Sie enthält eine Aufstellung aller grundlegenden Dienste eines sprachunabhängigen API für ausführbaren Programmcode in der SIM. Aufbauend auf dieser Norm spezifiziert die GSM 03.19 eine detaillierte Umsetzung für die Realisierung eines Java Card APIs für SIMs auf der Basis der Java Card 2.1-Spezifikationen. Diese Norm ist das zentrale Dokument für den Einsatz von Java Card bei GSM. Ergänzend dazu legt die GSM 11.13 Testumgebung, Testanwendungen, Testabläufe, Testabdeckung und die einzelnen Testfälle fest. Die beschriebenen Tests zielen ausnahmslos auf die informationstechnischen Aspekte einer Java Card für GSM.

Die GSM-Spezifikationen im Bereich des SIM erfahren keine Weiterentwicklung mehr, da die Funktionalität der SIM für die aktuellen Belange im GSM-System vollständig ausreicht. Die einzigen Änderungen, die an den diesbezüglichen Spezifikationen noch routinemäßig vorgenommen werden, ist die Klärung interpretierbarer Stellen. Seit ca. 1999 liegt der Fokus auf der Normung der UICC (*universal integrated circuit card*) mit der Anwendung USIM (*universal subscriber identity module*), welche vor allem im Rahmen von 3GPP vorangetrieben wurde.

Tabelle 13.2 Aufstellung der wichtigsten Normen für die SIM und SIM-relevanter Dienste.⁶

GSM 02.09	Security Aspects
GSM 02.17	SIM Functional Characteristics
GSM 02.19	Subscriber Identity Module Application Programming Interface (SIM API); Service description; Stage 1
GSM 02.48	Specification of security mechanisms for the SIM application toolkit, stage 1
GSM 03.19	Digital cellular telecommunications system (Phase 2+); Subscriber Identity Module Application Programming Interface (SIM API); SIM API for Java Card; Stage 2
GSM 03.48	Specification of security mechanisms for the SIM application toolkit, stage 2
GSM 09.91	Interworking aspects of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface between Phase 1 and Phase 2
GSM 11.11	Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface
GSM 11.12	Specification of the 3 Volt Subscriber Identity Module - Mobile Equipment (SIM - ME) interface
GSM 11.13	Test specification for SIM API for Java card
GSM 11.14	Specification of the SIM Application Toolkit for the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface
GSM 11.17	Subscriber Identity Module (SIM) conformance test specification
GSM 11.18	Specification of the 1.8 Volt Subscriber Identity Module - Mobile Equipment (SIM - ME) interface

13.2.2 Systemarchitektur und Komponenten

Jedes GSM-Netz lässt sich immer in drei generelle Teilsysteme aufgliedern. Diese sind in der Spezifikation GSM 01.02 (General description of a GSM Public Land Mobile Network (PLMN)) im Überblick beschrieben. Die drei Untersysteme sind: das Funkteilsystem (radio subsystem - RSS), das Vermittlungsteilsystem (network and switching subsystem - NSS) und das Betreiberteilsystem (operation subsystem - OSS).

Das Funkteilsystem setzt sich wiederum aus dem Endgerät (*mobile station* - MS) und dem Feststationenteilsystem (*base station subsystem* - BSS) zusammen. Das Endgerät besteht aus den physikalisch und logisch getrennten Teilen ME (*mobile equipment*) und SIM (*subscriber identity module*). Das Mobile Equipment ist das Radio- und Verschlüsselungsteil mit der Benutzerschnittstelle, und die SIM ist die nach GSM-Nomenklatur korrekte Bezeichnung für die GSM-spezifische Chipkarte. Beide zusammen ergeben das funktionsfähige Mobiltelefon.

In der Regel befindet sich im Mittelpunkt jeder Zelle das Feststationenteilsystem in Form der Feststation. Dieses hat die Aufgabe, einerseits den Kontakt über die Luftschnittstelle mit den Mobiltelefonen herzustellen und andererseits die Einspeisung zu den übergeordne-

⁶ Eine allgemeinere Aufstellung aller GSM-Normen im Umfeld der SIM findet sich im Normenverzeichnis des Anhangs. Alle GSM-Normen können im Übrigen kostenlos über den Web-Server von ETSI [ETSI] bezogen werden.

ten Teilen des Netzes vorzunehmen. Die Feststation setzt sich aus einer oder mehreren Base Transceiver Stations (BTS) und einem Base Station Controller (BSC) zusammen. Die Base Transceiver Station ist der eigentliche Sende- und Empfangsteil mit der entsprechenden Antenne und den dazugehörigen Hochfrequenzkomponenten. Ein typisches Empfangsmodul einer Base Transceiver Station hat acht Kanäle bei 200 kHz Bandbreite und kann somit theoretisch parallel acht aktive Verbindungen zu Mobilstationen aufrecht erhalten: In der Praxis sind nur sieben aktive Verbindungen üblich, da ein Kanal meist für die Verwaltungskommunikation reserviert ist. Üblicherweise werden ein, drei oder sechs Empfängermodule pro Base Transceiver Station installiert. Eine oder mehrere Base Transceiver Stations werden wiederum von einem Base Station Controller verwaltet. Eine typische Anordnung sind drei um 120 Grad zueinander versetzt angeordnete Base Transceiver Stations, die mit einem Base Station Controller verbunden sind. Bewegt sich eine Mobile Station vom Sende-Empfangsbereich einer Base Transceiver Station zu einer anderen Base Transceiver Station und beide sind dem gleichen Base Station Controller zugeordnet, so kann dieser nach Signalisierung an die zuständige Mobilvermittlungsstelle die Initiierung der Weitergabe (*handover*) selbstständig durchführen.

Die Datenübertragung über die Luftschnittstelle ist verschlüsselt, besitzt im *full-rate* Modus eine Netto-Übertragungsrate von 13 kBit/s und benutzt ein verlustbehaftetes Kompressionsverfahren mit technisch ausgefeilten Fehlerkorrekturmechanismen wie Frequenzsprungverfahren, Faltungscodierung und Interleaving.

Das Vermittlungsteilsystem besteht im Wesentlichen aus der Mobilvermittlungsstelle und dem Besucherregister (*visitor location register* – VLR). Die Verwaltung von mehreren Feststationenteilsystemen wird von einer Mobilvermittlungsstelle (*mobile switching center* – MSC) erledigt. Diese Vermittlungsstelle bildet das Bindeglied zwischen den an sie angeschlossenen Feststationenteilsystemen, anderen Mobilvermittlungsstellen und natürlich auch an das öffentliche Telefonnetz. Die Mobilvermittlungsstelle übernimmt Aufbau, Verwaltung und Abbau von Verbindungen, die Gebührenverrechnung und auch die Betreuung von Zusatzdiensten wie Rufweiterleitung, Rufsperrung und Konferenzschaltung. Im Besucherregister VLR befinden sich Informationen aller aktuell im Bereich der jeweiligen Mobilvermittlungsstelle befindlichen mobilen Endgeräte. Diese Informationen sind unter anderem notwendig, um beispielsweise einen Anruf an ein bestimmtes Mobiltelefon an das richtige Feststationenteilsystem und an die richtige Funkzelle weiterzuleiten. Das Besucherregister VLR führt auch die Endgeräte von Teilnehmern aus Fremdnetzen, sofern sich diese im Falle von Roaming in das Netz der jeweiligen Mobilvermittlungsstelle eingebucht haben.

Die oberste Hierarchieebene in einem GSM-System ist das Betreiberteilsystem. Es besteht aus dem Betriebs- und Wartungszentrum (*operation and maintenance centre* – OMC), dem Authentisierungszentrum (*authentication centre* – AuC), dem Heimatregister (*home location register* – HLR) und dem Gerätekennzeichnungsregister (*equipment identity register* – EIR). Das Betriebs- und Wartungszentrum ist für den laufenden Netzbetrieb, die Teilnehmerverwaltung und Gebührenabrechnung zuständig. Das Authentisierungszentrum ist die Sicherheitsinstanz auf Netzseite, sozusagen das Gegenüber zur SIM auf der Mobilseite. Sie erzeugt und verwaltet alle für den Betrieb des Systems notwendigen Schlüssel und Algorithmen vor allem für die Authentisierung der Mobile Stations (d. h. der SIMs). Eine weitere zentrale Instanz ist das Heimatregister HLR, die sowohl alle Teilnehmerdaten

als auch die Lokalisierungsdaten für das jeweilige Endgerät enthält. Das Gerätekennzeichnungsregister EIR ist das Äquivalent zum Heimatregister HLR, nur nicht für die Teilnehmer, sondern für die Endgeräte. Es enthält essenzielle Daten, wie z. B. die Seriennummer aller im Netz vertretenen Endgeräte.

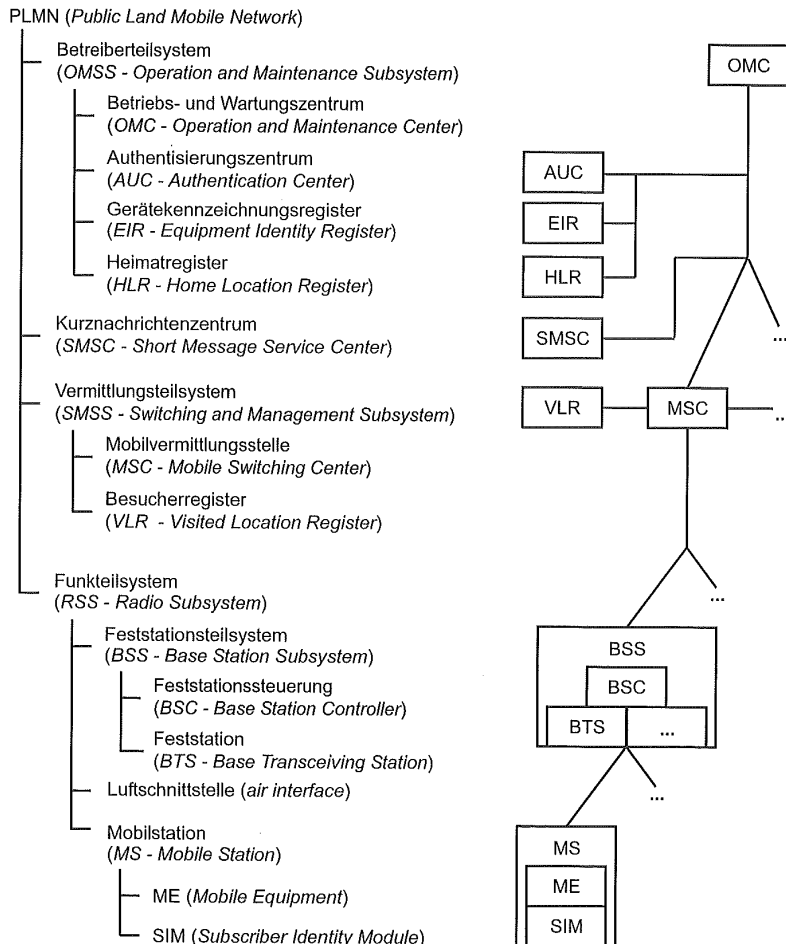


Bild 13.9 Die grundlegende Architektur eines typischen Mobilfunknetzes nach GSM Standard GSM 01.02. In diesem Beispielaufbau sind die Datenbanken EIR und HLR an zentraler Stelle angeordnet. Da die Konfiguration des PLMN dem jeweiligen Netzbetreiber in vielen Punkten selbst überlassen ist, können beispielsweise Datenbanken bei Bedarf (z. B. wg. hoher Netzlast) auch verteilt mehreren MSC zugeordnet werden. Zum besseren Verständnis ist die Verbindung zu einem SMSC (Short Message Service Center) eingezeichnet, obwohl dies kein direkter GSM-Systembestandteil ist. Eine oder mehrere Funkteilsysteme können einen Location Area (LA) und eine oder mehrere Vermittlungsteilsysteme eine Service Area (SA) formen.

Tabelle 13.3 Die für den Betrieb des GSM-Systems wesentlichen Datenbanken mit den wichtigsten darin enthaltenen Datenelementen

Datenbank	Datenelemente
Heimatregister HLR (<i>home location register</i>)	Teilnehmerdaten (<i>subscriber information</i>) IMSI (<i>international mobile subscriber identity</i>) MSISDN (<i>mobile station ISDN number</i>) Einschränkungen bei Diensten (z. B. Roaming nicht erlaubt) Abonnierte Dienste Parameter für Zusatzdienste Informationen über das Gerät des Teilnehmers Authentifizierungsdaten (d. h. RAND, SRES, Kc Tripel) (implementierungsabhängig) Lokalisierungsdaten (<i>mobile location information</i>) MSRN (<i>mobile station roaming number</i>) Adresse des aktuellen VLR (falls verfügbar) Adresse des aktuellen MSC (falls verfügbar) TMSI (falls verfügbar)
Besucherregister VLR (<i>visited location register</i>)	Teilnehmerdaten (<i>subscriber information</i>) IMSI (<i>international mobile subscriber identity</i>) MSISDN (<i>mobile station ISDN number</i>) Parameter für Zusatzdienste Informationen über das Gerät des Teilnehmers Authentifizierungsdaten (d. h. RAND, SRES Tupel) (implementierungsabhängig) Lokalisierungsdaten (<i>mobile location information</i>) TMSI (<i>temporary mobile subscriber identity</i>) MSRN (<i>mobile station roaming number</i>) LAI (<i>location area information</i>) TMSI (falls verfügbar)
Gerätekennezeichnungs- register EIR (<i>equipment identity register</i>)	IMEI (<i>international mobile equipment identity</i>) aller Endgeräte (<i>white list</i>) IMEI von zu meldenden Endgeräten (<i>greylist</i>) IMEI von gesperrten Endgeräten (<i>blacklist</i>)

13.2.3 Wichtige Datenelemente

Der folgende Absatz enthält eine Auswahl wichtiger Datenelemente, die vor allem im Zusammenhang mit der SIM und ihren Funktionen stehen. Die Decodierung der beschriebenen Datenelemente findet sich bei der Beschreibung der typischen Dateien einer SIM.

Codierung alphanumerischer Zeichen

Alphanumerische Zeichen in dem ursprünglichen mitteleuropäischen GSM-System wurden und werden mit einer ASCII angelehnten 7-Bit Codierung dargestellt. Diese ist in der GSM-Spezifikation GSM 03.38 definiert. Die weltweite Verbreitung von GSM machte jedoch eine Erweiterung des Zeichensatzes notwendig. Deshalb wird für Zeichen, die sich

nicht mit dem westeuropäischen Zeichensatz nach GSM 03.38 darstellen lassen, der UCS (*universal character set*)-Zeichensatz in der 16 Bit breiten Untervariante UCS-2 benutzt. Mit diesem Zeichensatz lassen sich die wichtigsten Zeichen aller lebenden Sprachen darstellen.⁷

Für die Codierung von UCS-2 Zeichenketten sind aus Gründen der Speicherplatzminimierung drei unterschiedliche Schemata festgelegt. Das bevorzugte, doch am meisten Speicher benötigende Schema 1 wird durch '80' als erstes Byte gekennzeichnet. Anschließend folgen die 16 Bit breiten UCS-2 Zeichen mit dem höherwertigsten Byte zuerst. Nicht benutzte Bytes werden auf 'FF' gesetzt.

Schema 2 wird durch das erste Byte mit dem Wert '81' indiziert. Das zweite Byte beinhaltet die Anzahl der Zeichen der Zeichenkette. Die beiden folgenden Bytes repräsentieren einen 16-Bit-Zeiger im UCS Zeichensatz, welcher als Offset zu einer sprachentypischen Auswahl von Zeichen innerhalb von UCS verwendet wird. Bei diesem Zeiger sind die Bits 1 bis 7 und Bit 16 auf Null gesetzt. Falls Bit 8 eines der folgenden Zeichen den Wert 1 hat, sind Bit 1 bis 7 dieses Bytes auf den Zeiger zu addieren und der so ergänzte 16-Bit-Zeiger weist zu dem jeweiligen konkreten Zeichen im UCS-Zeichensatz. Sollte Bit 8 den Wert 0 haben, so handelt es sich um ein Zeichen des 7-Bit-Zeichensatzes nach GSM 03.38.

Schema 3 wird durch das Anfangsbyte '82' gekennzeichnet. Analog zu Schema 2 beinhaltet das zweite Byte wiederum die Länge der Zeichenkette und das dritte und vierte Byte repräsentieren einen vollständigen 16-Bit-Zeiger auf die UCS-Zeichentabelle. Falls im nachfolgenden Byte das Bit 8 den Wert 1 hat, sind die folgenden 7 Bits auf den Zeiger zur eindeutigen Bestimmung des UCS-Zeichens zu addieren. Hat das Bit 8 den Wert 0, handelt es sich um ein 7-Bit-codiertes GSM-03.38-konformes Zeichen.

Diensteerkennungen (*SIM service table – SST*)

Die SST enthält eine Tabelle mit zusätzlich zum Sprachdienst möglichen bzw. freigeschalteten Diensten, wie z. B. Kurznachrichtendienst oder Festrufnummerndienst.

Festrufnummern (*fixed dialling numbers – FDN*)

Festrufnummern sind eine besondere Form von Rufnummern, die auch dann angewählt werden können, wenn alle anderen Rufnummern auf dem Mobiltelefon gesperrt sind.

ICCID (*ICC identification*)

Die ICCID ist eine eindeutige Identifikationsnummer der Chipkarte, BCD-codiert, 10 Byte lang und kann bei Bedarf von rechts mit 'F' aufgefüllt sein.

IMEI (*international mobile equipment identity*)

Die IMEI ist eine eindeutige Gerätenummer der Mobilstation, hat 15 Stellen und belegt damit üblicherweise 8 Byte. Sie setzt sich aus sechs Stellen *type approval code*, zwei Stellen Herstellercode, sechs Stellen Seriennummer und einer Prüfziffer zusammen. Die IMEI ist im Mobiltelefon und an zentraler Stelle im Gerätekennezeichnungsregister EIR gespeichert.

⁷ Siehe auch Abschnitt 4.2 „Codierung alphanumerischer Daten“.

IMSI (*international mobile subscriber identity*)

Die IMSI ist die im GSM-System eindeutige Teilnehmeridentität, ist BCD-codiert und hat eine Länge von 9 Byte, die bei Bedarf von rechts mit 'F' aufgefüllt ist. Sie setzt sich zusammen aus dem Landecode MCC (*Mobile Country Code*), dem Netzwerkcode MNC (*Mobile Network Code*) und einer laufenden Nummer, die vom Netzbetreiber vergeben wird. Die IMSI wird im Regelfall nie im Klartext über die Luftschnittstelle übertragen, um zu verhindern, dass der Aufenthaltsort einer Mobilstation mit bekannter IMSI unberechtigt verfolgt werden kann. Anstelle der IMSI wird deshalb zur Identifizierung in der Regel die TMSI in Verbindung mit der LAI benutzt.

Ki (*key individual*), Kc (*key cipher*)

Die Schlüssel Ki und Kc sind geheime Schlüssel für symmetrische Kryptoalgorithmen. Ki ist dabei der kartenindividuelle Schlüssel für die kryptografische Berechnung der Authentizität des SIMs, und Kc dient zur Verschlüsselung der Daten auf der Luftschnittstelle zwischen Mobilstation und Feststation.

Kurznachrichten (*short message service – SMS*)

Kurznachrichten sind ein Dienst bei dem mit einer maximalen Nachrichtenlänge von 160 alphanumerischen Zeichen über den Signalisierungskanal Daten zwischen Netzwerk und Mobilstation übertragen werden können. Der SMS-Dienst wird nicht nur zur Übermittlung von Kurznachrichten für die Teilnehmer benutzt, sondern auch als Trägerdienst zur Übertragung von Daten an das Mobiltelefon oder die SIM beispielsweise bei WAP oder bei OTA-Diensten.

Kurzurufnummern (*abbreviated dialling numbers – ADN*)

Kurzurufnummern sind im Mobiltelefon oder der SIM mit zusätzlichen Informationen gespeicherte Rufnummern, die über ein Menü oder spezielle Tasten einfach und schnell gewählt werden können.

LAI (*location area information*)

Die LAI ist eine eindeutige Ortsinformation der Mobilstation und wird dazu benutzt, um in Verbindung mit der TMSI eine eindeutige Teilnehmeridentität zu erzeugen. Sie setzt sich aus 3 Stellen *Country Code* (CC), 2 Stellen *Mobile Network Code* (MNC) und einem max. fünfstelligen *Location Area Code* (LAC) zusammen.

MSISDN (*mobile station ISDN number*)

Die MSISDN ist die Rufnummer der Mobilstation. Sie ist unabhängig von der Teilnehmeridentität IMSI.

TMSI (*temporary mobile subscriber identity*)

Die TMSI ist eine zeitlich und örtlich beschränkte Teilnehmeridentität mit einer Länge von 4 Bytes. Sie wird zum Schutz der Teilnehmeridentität benutzt. Die TMSI ist nur in Verbindung mit der Ortsinformation LAI eindeutig. Die TMSI wird vom VLR vergeben und auch dort gespeichert.

13.2.4 Die SIM (*Subscriber Identity Module*)

Die SIM ist ein obligatorisches Sicherheitsmodul, das sich austauschbar in mobilen Endgeräten des GSM-Mobilfunksystems befindet. Sie ist in der Spezifikation GSM 02.17 folgendermaßen definiert: „Die SIM ist eine Entität, welche die Identität des Teilnehmers enthält. Die primäre Funktion der SIM ist die Sicherstellung der Echtheit der Mobilstation gegenüber dem Netzwerk.“

Neben diesen Hauptaufgaben als Träger der Identität des Teilnehmers, die durch eine PIN-Prüfung verwirklicht wird, und der Authentisierung der Mobilstation gegenüber dem Netzwerk hat die SIM noch eine Reihe weiterer Funktionen. Sie ermöglicht die manipulationssichere Ausführung von Programmen, die Speicherung von Daten, wie beispielsweise Rufnummern, Kurznachrichten und persönlichen Einstellungen des Mobiltelefons. Weiterhin ist sie Träger für sichere Zusatzdienste im Rahmen der mobilen Kommunikation.

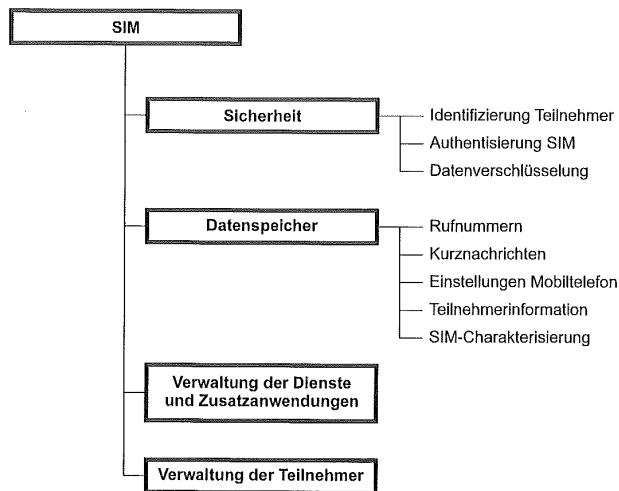


Bild 13.10 Klassifizierungsbaum der grundlegenden Funktionen des SIM im GSM-System.

Das SIM kann im GSM-System in zwei verschiedenen Kartenformaten vorkommen. Bei Mobiltelefonen, die einen öfteren Wechsel des SIMs vorsehen, wird das ID-1 Format verwendet. Die Idee dahinter war das Familien- oder Firmenmobiltelefon mit einer Karte pro Benutzer. Mobiltelefone, bei denen das SIM nur sehr selten gewechselt werden soll und die sehr kleine Abmessungen haben, verwenden ein Plug-In-SIM im ID-000 Format. Die Unterschiede zwischen den beiden Kartengrößen liegen aber ausschließlich in der physischen Größe der Karte. In ihren logischen oder physikalischen Eigenschaften sind beide absolut identisch. Seit Mitte der 90er-Jahre wurden Mobiltelefone zu einem mehr oder minder personenbezogenen Gerät, was auch Auswirkungen auf die eingesetzten Kartengrößen hatte, da ein benutzerabhängiger Wechsel der Karten damit nicht mehr notwendig war. 1995 war bereits die Hälfte aller verkauften ID-1-Karten (seit 1998 praktisch alle) mit einer Stanzung zum Herausbrechen der ID-000 Karten versehen.

Die Kommunikation zwischen Mobile Equipment und SIM läuft mit dem Übertragungsprotokoll T = 0 in den Standardparametern nach ISO/IEC 7816-3. Die *convention* der Datenübertragung kann von der Karte im ATR frei gewählt werden. Ein PPS ist vorgesehen und wird zur Erhöhung der Datenübertragungsrate oftmals auch benutzt. Der von vielen Mobiltelefonen typischerweise benutzte Teiler (*clock rate conversion factor*) ist 64, woraus eine Übertragungsrate von 78 kBit/s bei 5 MHz Taktfrequenz resultiert. Vereinzelt wird sogar bereits Teiler 31 (≈ 156 kBit/s bei 5 MHz Taktfrequenz) eingesetzt. Das T = 0 Kommunikationskommando GET RESPONSE weist indes aus historischen Gründen zwei Inkompatibilitäten zu ISO/IEC auf.⁸ Können mit GET RESPONSE Daten vom Terminal abgeholt werden, so wird dies seitens der SIM mit '9F' im SW1-Byte signalisiert und nicht, wie bei ISO/IEC 7816-3 definiert, mit '61'. GET RESPONSE weist jedoch noch eine weitere Besonderheit auf. Nach GSM 11.11 können mit einem GET RESPONSE die von der SIM bereitgestellten Daten Byte für Byte abgeholt werden. In der SIM wird dementsprechend ein Zeiger im Sendepuffer mitgeführt. Nach ISO/IEC 7816-3 ist dies nicht möglich, nach dieser Norm können die mit GET RESPONSE abzuholenden Daten vom ersten Byte an oder als gesamter Block vom Terminal angefordert werden. Die beiden Inkompatibilitäten führen jedoch in der Praxis zu keinen signifikanten Problemen.

1998 wurde zum zehnjährigen Bestehen der Normen für die SIM von der SMG9 der in Bild 13.11 aufgezeigte Leitspruch veröffentlicht. Er zeigt einerseits ziemlich deutlich, welche Bedeutung und Größe das GSM-System damals bereits erreicht hatte und andererseits wie stolz man auf eines der wesentlichen Elemente des Systems – das Subscriber Identity Module – ist.

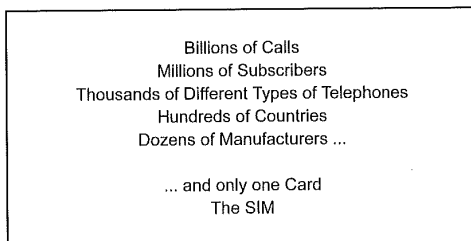


Bild 13.11 Leitspruch der SIM-Normungsgruppe SMG9 im Jahr 1998 zu ihrem zehnjährigen Bestehen.

Die Spezifikationen für die SIM waren die Grundlage vieler weiterer Spezifikationen für Chipkarten im Mobilfunkbereich. Die wichtigsten sind im Folgendem kurz zusammengefasst:

Im Jahr 1992 wurde im Rahmen der europäischen ETSI-Normung von DECT (*digital enhanced cordless telecommunications*), einem Standard für schnurlose Telefone in Zellulartechnik im 1,9 GHz-Bereich, die erste Version der Spezifikation für das DAM (*DECT Authentication Module*) veröffentlicht. Diese Spezifikation wurde 1995 unter der ETSI-Nummerierung ETS 300 331 eingefroren. Das DAM wurde in der Praxis leider aber nie als

⁸ Streng genommen ist hier *de facto* die ISO/IEC 7816-3 zu GSM 11.11 nicht kompatibel, da Letztere chronologisch die Erste war. Doch ist normungstechnisch eine ISO/IEC-Norm höherwertiger, weshalb hier *de jure* GSM nicht kompatibel ist.

Produkt umgesetzt, da es laut Spezifikation optional ist und damit bei allen Herstellern den Kostenreduktionsprogrammen für die mobilen Endgeräte zum Opfer fiel.

Im digitalen Bündelfunksystem TETRA (*terrestrial trunked radio*) [TETRA] ist ebenfalls optional eine Chipkarte namens TETRA-SIM vorgesehen, deren Spezifikation auf der SIM der GSM-Mobiltelefone basiert. Die Spezifikation EN 300 812 für die TETRA-SIM sieht auch vor, dass diese bei Bedarf als Anwendung auf der UICC realisiert werden kann. Die TETRA-SIM ist jedoch optional, weshalb sie auch als Softwarelösung im Endgerät existieren kann.

Eine weitere Chipkarte, deren Spezifikation auf denen der SIM gründet, ist die R-UIM (*removable user identity module*) in 3G-Mobilfunksystemen, die im Rahmen von 3GPP2 (*third generation partnership project 2*) definiert wurden, wie beispielsweise dem CDMA 2000 Mobilfunksystem. Die R-UIM ist in den entsprechenden Endgeräten optional vorgesehen und hat eine ähnliche Funktionalität wie die SIM. Sie ist in den Normen TIA/EIA/IS-820 und TIA/EIA/IS-839 spezifiziert. Ein wesentlicher Unterschied zur SIM ist der Kryptoalgorithmus CAVE (*cellular authentication, voice privacy and encryption*), der, wie sein ausgeschriebener Name schon aussagt, für eine Vielzahl von kryptografisch gesicherten Funktionen in der R-UIM eingesetzt wird. Angelehnt an das SIM Application Toolkit, ist für die R-UIM auch ein UIM Application Toolkit (UATK) spezifiziert.

Im satellitengestützten Inmarsat [Inmarsat] Mobiltelefonnetz, das sich seit Anfang der 80er-Jahre in Betrieb befindet, werden mittlerweile ebenfalls modifizierte GSM-Karten als Grundlage zur Feststellung der Teilnehmeridentität benutzt. Eine andere Erweiterung der SIM, die sich durch wenige zusätzliche Dateien und einen speziellen Kryptoalgorithmus auszeichnet, ist die Chipkarte für das weltweite Mobilfunknetz Iridium [Iridium]. Dieses besteht im Endausbau aus 66 in 780 km Höhe kreisenden Satelliten, welche das Äquivalent zu den GSM-Basisstationen sind. Die benutzte Frequenz auf der Luftschnittstelle zwischen Mobilteil und Satelliten liegt bei 1616 MHz. Ob dieses technisch interessante und durchaus anspruchsvolle Mobilfunksystem mittel- und langfristig Bestand haben wird, hängt von der etwas misslichen finanziellen Lage seiner Betreibergesellschaft ab.

Kommandos einer SIM

Die GSM 11.11 Spezifikation definiert 22 operative Kommandos für die SIM, die durch die Class 'A0' identifiziert werden.⁹ Die Kommandos können in die Kategorien Sicherheit, Operationen auf Dateien und SIM Application Toolkit eingeteilt werden. Einen Überblick hierzu zeigt Tabelle 13.4.¹⁰

Tabelle 13.4 Die in GSM 11.11 festgelegten Chipkarten-Kommandos für die SIM

Kommando	Kurzbeschreibung
Kategorie Sicherheit	
CHANGE CHV	Ändern der PIN
DISABLE CHV	Abschalten der PIN-Abfrage
ENABLE CHV	Einschalten der PIN-Abfrage

⁹ Aufteilung der Kommandos siehe auch Abschnitt 6.5.1 „Struktur der Kommando-APDUs“.

¹⁰ Zur Beschreibung der typischen Chipkarten-Kommandos siehe auch Kapitel 7 „Kommandos von Chipkarten.“

Tabelle 13.4 (Fortsetzung)

Kommando	Kurzbeschreibung
RUN GSM ALGORITHM	Ausführen des GSM-spezifischen Kryptoalgorithmus
UNBLOCK CHV	Zurücksetzen des abgelaufenen PIN-Fehlbedienungs Zählers
VERIFY CHV	Überprüfen der PIN
Kategorie Operationen auf Dateien	
INCREASE	Erhöhen eines Zählers in einer Datei
INVALIDATE	Reversibles Blocken einer Datei
READ BINARY	Lesen aus einer Datei mit „transparenter“ Struktur
READ RECORD	Lesen aus einer Datei mit Record-orientierter Struktur
REHABILITATE	Entblocken einer Datei
SEEK	Suchen eines Textes in einer Datei mit Record-orientierter Struktur
SELECT	Auswahl einer Datei
STATUS	Lesen von verschiedenen Informationen zur aktuell selektierten Datei
UPDATE BINARY	Schreiben in eine Datei mit „transparenter“ Struktur
UPDATE RECORD	Schreiben in eine Datei mit Record-orientierter Struktur
Kategorie SIM Application Toolkit	
ENVELOPE	Übergabe von Daten an eine Zusatzanwendung der SIM im Rahmen des SIM Application Toolkit.
FETCH	Abholung eines SIM Application Toolkit-Kommandos von der SIM an das Mobile Equipment.
TERMINAL PROFILE	Aufstellung aller Funktionalitäten des Mobile Equipment hinsichtlich des SIM Application Toolkit.
TERMINAL RESPONSE	Übergabe der Antwort des Mobile Equipment an ein vorangegangenes SIM Application Toolkit Kommando der SIM.
Kategorie Sonstige	
GET RESPONSE	T=0 spezifisches Kommando zur Anforderung von Daten von der Chipkarte
SLEEP	Obsoletes Kommando, um die Chipkarte in einen energiesparenden Zustand zu versetzen

Bei der Eingabe der 4-stelligen PIN, die im Übrigen bei GSM die Bezeichnung CHV (*card holder verification*) hat, gibt es eine Besonderheit. Mit einem speziellen Kommando (DISABLE CHV) und der richtigen PIN können weitere PIN-Abfragen der Benutzer-PIN abgeschaltet werden, sodass es nicht mehr notwendig ist, die PIN vor dem Einbuchen ins Mobilfunknetz einzugeben. Der Nachteil, dass verlorene Karten bis zur Sperrung durch den Netzbetreiber widerrechtlich zum Telefonieren verwendet werden können, liegt in der Verantwortung des Benutzers. Mit einem weiteren Kommando (ENABLE CHV) kann der Benutzer die Abfrage der PIN bei Bedarf auch wieder einschalten.

Im Normalfall hat eine SIM zwei CHVs. Die Idee dabei ist eine Unterscheidung zwischen Kartenbenutzer und Kartenbesitzer, um damit unterschiedliche Funktionen zu ermöglichen bzw. bestimmte Funktionen nur dem Kartenbesitzer zu erlauben. Am Beispiel des EF_{FDN} mit den Festrufnummern sei dies hier kurz verdeutlicht. Dem Kartenbenutzer ist nur die CHV 1 bekannt, was zum Wählen der in EF_{FDN} gespeicherten Rufnummern aus-

reicht. Der Kartenbesitzer kennt jedoch auch die CHV 2 und kann aufgrund der Zugriffsbedingung für UPDATE RECORD – was eine korrekte Überprüfung von CHV 2 voraussetzt – auch die Einträge im EF_{FDN} ändern. Damit ließe sich zum Beispiel für die Mobiltelefone der Kinder der wählbare Rufnummernkreis über EF_{FDN} einschränken, da diese zum Telefonieren nur die CHV 1 kennen müssen. Die Eltern könnten dann mit dem Wissen um die CHV 2 die wählbaren Rufnummern auch bearbeiten.

Aus Kompatibilitätsgründen unterstützen alle SIMs noch das Kommando SLEEP, obwohl dieses seit vielen Jahren obsolet ist. Seine ursprüngliche Aufgabe – Sparen von elektrischer Energie im Endgerät – wird mittlerweile von der Hardware der Chipkarten-Mikrocontroller bzw. dem Betriebssystem übernommen.

Das Kommando STATUS wird neben seiner Funktion zur Anforderung von Informationen über die aktuell selektierte Datei noch für einen weiteren Zweck gebraucht. Das Mobile Equipment sendet STATUS in regelmäßigen Zeitabständen von zirka 30 Sekunden zur SIM, um ihr Vorhandensein festzustellen. Bleibt eine Antwort der SIM innerhalb von 5 Sekunden auf ein gesendetes STATUS-Kommando aus, so wird diese abgeschaltet und die Verbindung beendet. Zusätzlich ist in der Regel noch in irgendeiner Form ein mechanischer Kontakt existent, um einen Wechsel der SIM während des Betriebs zu detektieren bzw. zu verhindern.

Die einschlägigen GSM-Spezifikationen legen keine administrativen Kommandos zur Verwaltung von Dateien fest. Dies war ursprünglich auch nicht notwendig, da das Erzeugen und Löschen von Dateien lange Zeit aus Speicherplatzgründen bei Chipkarten-Betriebssystemen nicht möglich war. Mittlerweile hat sich dies jedoch grundlegend geändert, sodass nun diese (im Grunde genommen sehr wichtigen) Verwaltungsfunktionen für Dateien zur Verfügung stehen. Damit können dateibasierte Anwendungen auch über Remote File Management zu beliebigen Zeitpunkten auf SIMs nachgeladen werden, vorausgesetzt, es ist genügend Freispeicher für Dateien vorhanden.

Beschrieben sind die administrativen Kommandos in der TS 102.222, einer Spezifikation aus dem 3GPP-Umfeld und ursprünglich für die USIM konzipiert. Da sich die administrativen Kommandos für chipkartenbasierte Dateisysteme zwischen SIM und USIM jedoch nicht fundamental voneinander unterscheiden, hat sich in der Praxis vorgenannte Norm auch im SIM-Umfeld fest etabliert.

Tabelle 13.5 Die in TS 102.222 festgelegten Chipkarten-Kommandos für die Verwaltung von Anwendungen auf Telekommunikations-Chipkarten.

Kommando	Kurzbeschreibung
ACTIVATE FILE	Entblocken einer Datei
CREATE FILE	Erzeugen einer neuen Datei
DEACTIVATE FILE	Reversibles Blocken einer Datei
DELETE FILE	Löschen einer Datei
TERMINATE CARD USAGE	Irreversibles Sperren einer Chipkarte
TERMINATE DF	Irreversibles Sperren eines DFs
TERMINATE EF	Irreversibles Sperren eines EFs

Dateien einer SIM

Das SIM hat ein hierarchisch aufgebautes Dateisystem mit einem MF und zwei DF direkt unter dem MF. Sowohl unter dem MF als auch in den DFs befinden sich EFs mit den Daten für die Anwendung. Die möglichen Dateistrukturen für die EFs sind transparent, linear fixed und zyklisch.

Die File Identifier (FID) der SIM-Dateien weisen die Besonderheit auf. So hat das erste Byte aller DFs unter dem MF immer den Wert '7F', DFs direkt unter dem DF GSM hingegen den Wert '5F' und EFs den Wert '5F'. EFs direkt unter dem MF müssen als erstes Byte des FIDs den Wert '2F' haben und EFs unter dem DF TELEKOM den Wert '6F' bzw. EFs im DF MExE den Wert '4F'. Diese Konventionen sind zum großen Teil Überbleibsel aus den Anfängen der Mikroprozessor-Chipkarten und haben heute längst keine praktische Bedeutung mehr.

Die Zugriffsbedingungen auf alle Dateien sind zustandsorientiert und werden pro Datei separat für die vier Zugriffsbefehle READ, UPDATE, INVALIDATE und REHABILITATE festgelegt. Es existieren 16 verschiedene Zustände für den Dateizugriff, der Sicherheit nach aufsteigend eingeteilt von 0 bis 15. Zustand 0 als Zugriffsbedingung für eine Datei bedeutet ALWAYS, d. h. auf diese Datei darf mit dem entsprechenden Zugriffsbefehl immer zugegriffen werden. Zustand 15 drückt das andere Extrem aus, er besagt, dass niemals (NEVER) mit einem entsprechenden Zugriffsbefehl auf die Datei zugegriffen werden darf. Zustand 1 bedeutet nur Zugriff nach erfolgreicher Prüfung von CHV 1, d. h. PIN Nummer 1. Analog verhält es sich mit Zustand 2, bei dem vorab CHV 2 erfolgreich geprüft sein muss. Zustand 3 wird zurzeit nicht benutzt und bleibt für zukünftige Verwendung reserviert. Die Zustände 4 bis 14 sind administrativen Zwecken vorbehalten, d. h. der Netzbetreiber kann mithilfe der Überprüfung spezieller PINs oder Authentisierungen auf Dateien mit diesen Zugriffsbedingungen zugreifen.

Direkt unter dem MF sind alle EFs angeordnet, die allgemeine Informationen über die Chipkarte an sich beherbergen, wie z. B. eine eindeutige Seriennummer (ICCID) der Karte. Im DF TELEKOM befinden sich alle für das GSM-System relevanten EFs, das typische Beispiel dafür sind die Kurzrufnummern. Das DF GSM beherbergt hingegen alle EFs mit netzbetreiberspezifischen Informationen, wie etwa die IMSI.

In der GSM 11.11 Spezifikation sind in der Summe 70 verschiedene EFs festgelegt, wobei lediglich 12 davon mit insgesamt ca. 110 Byte Dateninhalt obligatorisch sind. Die restlichen EFs sind optional und je nach Netzbetreiber und den angebotenen Diensten im Dateisystem der SIM vorhanden. Zusätzlich zu diesen spezifizierten Dateien kann der jeweilige Netzbetreiber eigene Dateien für Wartungs- oder Administrationszwecke im Dateibaum der SIM ablegen. Von dieser Möglichkeit wird im Übrigen auch intensiv Gebrauch gemacht, sodass in der Praxis typischerweise etwa 40 Dateien mit etwa 12 kByte Nutzdaten auf einer SIM vorhanden sind.

Es existieren im Dateibaum der SIM einige EFs, welche besonders oft geschrieben werden müssen. Ein Beispiel dafür ist ein EF namens LOCI (*location information*). In dieser Datei wird neben der jeweils gültigen TMSI (*temporary mobile subscriber identity*) noch eine zusätzliche Ortsinformation, die LAI (*location area information*), abgelegt. Die Daten in dieser Datei müssen sowohl bei jedem Wechsel der Feststation als auch bei jedem Gespräch geändert werden. Aus diesem Grund muss ein SIM-Betriebssystem ein spezielles Dateiattribut unterstützen, das *high update activity* heißt. Technisch wurde dies durch meh-

rere Dateibodies unter einem Dateiheader realisiert. Im Falle eines auftretenden Defekts eines Dateibodies wurde vom Betriebssystem automatisch auf einen Ersatz-Dateibody umgeschaltet. Entstanden ist dieses Dateiattribut noch in der Zeit, als EEPROM-Seiten nur 10000 garantierte Schreib-/Löschzyklen hatten. Durch die technologische Weiterentwicklung der EEPROM-Zellen erreicht die Zyklenzahl jedoch den Bereich einer halben Million, weshalb diese Attribut *de facto* obsolet geworden ist. Die GSM-Spezifikationen enthalten es freilich weiterhin, wobei die Chipkarten-Betriebssysteme Dateien mit diesem Attribut heute nicht anders behandeln als Dateien ohne.

Ursprünglich war beabsichtigt, die GSM-Chipkarten alle zwei Jahre auszutauschen, um Ausfällen durch die begrenzte Anzahl von Schreib-/Löschzyklen des EEPROM zuvorzukommen. Da sich in diesem Bereich aber bislang nahezu keine Probleme ergaben, ersetzen

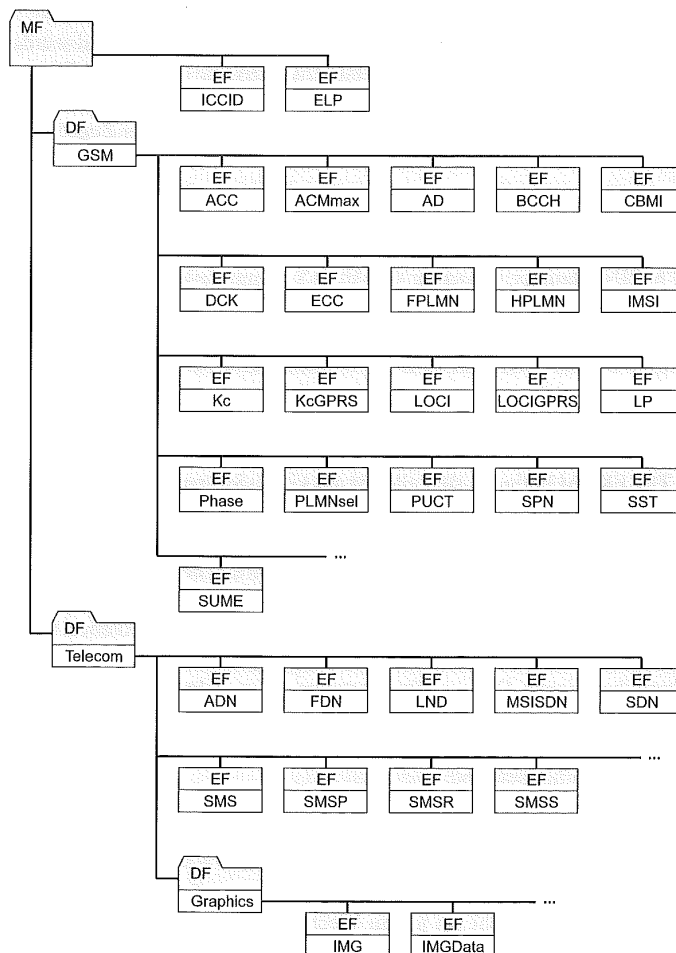


Bild 13.12 Die wichtigsten Dateien einer SIM im Überblick.

die meisten Anwendungsanbieter die Chipkarten nur mehr im Falle des Ausfalls. Dies erspart dem jeweiligen Anbieter erhebliche Kosten, da er sich logistisch nur noch um den Austausch defekter Karten kümmern muss. Des Weiteren reduziert sich dadurch die Anzahl auszutauschender Karten betr chtlich, da die meisten Karten eine wesentlich h here Lebensdauer als zwei Jahre haben. Dies senkt betr chtlich die Beschaffungskosten, da eine Chipkarte nur noch dann durch eine neue ersetzt wird, wenn sie nicht mehr funktioniert. Die Praxis hat gezeigt, dass ein Kartenaustausch etwa alle 5 bis 7 Jahre notwendig ist.

Tabelle 13.6 Typische Dateien einer SIM nach GSM 11.11 mit Codierung der Datenelemente und decodierten erl uterten Beispielen

MF		Wurzelverzeichnis
	Beschreibung	Dies ist das Ursprungsverzeichnis f�r die gesamte SIM.
	Datei	FID = '3F00'
DF_{TELECOM}		Verzeichnis Telekom
	Beschreibung	In diesem Verzeichnis sind alle Dateien zusammengefasst, die spezifisch f�r die Dienste sind.
	Datei	FID = '7F10'
DF_{GSM}		Verzeichnis GSM
	Beschreibung	In diesem Verzeichnis sind alle Dateien zusammengefasst, die spezifisch f�r das GSM-Netzwerk sind.
	Datei	FID = '7F20' oder FID = '7F21' (aus Kompatibilit�tsgr�nden zu �lteren GSM1800 Mobiltelefonen)
DF_{GRAPHICS}		Verzeichnis Grafiken
	Beschreibung	In diesem Verzeichnis sind alle Dateien zusammengefasst, die Grafik-Informationen enthalten.
	Datei	FID = '5F50'
MF.EF_{ELP}		Erweiterung f�r bevorzugte Sprache (<i>ELP – extended language preference</i>)
	Beschreibung	In dieser Datei ist eine erweiterte Aufstellung der bevorzugten Sprachen f�r die Benutzerschnittstelle gespeichert.
	Datei	FID = '2F05'; Struktur: transparent, Dateigr��e: 2*x Byte; Zugriffe: READ: immer; UPDATE: CHV 1
	Codierung	jeweils zwei alphanumerische Zeichen als L�ndercode nach ISO 639 im GSM 03.38 Alphabet Byte 1, 2: h�chstpriore Sprache ... Byte 2*x-1, 2x niedrigstpriore Sprache
	Beispiel	'64 65' ⇒ h�chstpriore Sprache: deutsch '65 6E' ⇒ zweith�chstpriore Sprache: englisch '66 72' ⇒ dritth�chstpriore Sprache: franz�sisch '65 73' ⇒ niedrigstpriore Sprache: spanisch

Tabelle 13.6 (Fortsetzung)

MF.EF_{ICCID}	Identifikationsnummer der Chipkarte (ICCID – ICC identification)	
Beschreibung	In dieser Datei ist eine eindeutige Identifikationsnummer der Chipkarte gespeichert.	
Datei	FID = '2FE2'; Struktur: transparent, Dateigröße: 10 Byte Zugriffe: READ: immer; UPDATE: nie	
Codierung	laufende Nummer, BCD-codiert, linksbündig ausgerichtet und bei Bedarf rechts mit 'F' aufgefüllt Byte 1, Bit 1 ... Bit 4: Digit 1 Byte 1, Bit 5 ... Bit 8: Digit 2 Byte 2, Bit 1 ... Bit 4: Digit 3 usw.	
Beispiel	'98 94 20 00 00 10 81 85 39 11' ⇒ 89 49 02 00 00 01 18 58 93 11	
DF_{GSM}.EF_{ACM}	kumulierte Gebühreneinheiten (ACM – accumulated call meter)	
Beschreibung	In dieser Datei ist die ab einem bestimmten Startzeitpunkt gesammelte Anzahl von Gebühreneinheiten gespeichert.	
Datei	FID = '6F39'; Struktur: cyclic, n * 3 Bytes Zugriffe: READ: CHV 1; UPDATE: CHV 2	
Codierung	Byte 1 ... 3: gesammelte Anzahl von Gebühreneinheiten	
DF_{GSM}.EF_{ACMmax}	maximale kumulierte Anzahl von Gebühreneinheiten (ACM – accumulated call meter maximum)	
Beschreibung	In dieser Datei ist die maximal mögliche Anzahl von Gebühreneinheiten gespeichert.	
Datei	FID = '6F37'; Struktur: transparent, 3 Bytes Zugriffe: READ: CHV 1; UPDATE: CHV 2	
Codierung	Byte 1 ... 3: maximale Anzahl von Gebühreneinheiten	
DF_{GSM}.EF_{FPLMN}	verbotene Netzbetreiber (FPLMN – forbidden public land mobile network)	
Beschreibung	In dieser Datei ist eine Liste der verbotenen Netzbetreiber gespeichert.	
Datei	FID = '6F7B'; Struktur: transparent, 12 Bytes Zugriffe: READ: CHV 1; UPDATE: CHV 1	
Codierung	Byte 1 ... 3: verbotenes PLMN Nr. 1 Byte 4 ... 6: verbotenes PLMN Nr. 2; ... Datenaufbau und Beispiele: siehe EFPLMNsel	
Beispiel	'FF FF FF FF FF FF FF FF 62 F2 20' '62 F2' ⇒ MCC ⇒ '262' ⇒ Deutschland '10' ⇒ MCC ⇒ '01' ⇒ Deutschland D1	
DF_{GSM}.EF_{HPLMN}	Zeitintervall für die Suche nach dem Heimatnetz (HPLMN – home public land mobile network search period)	
Beschreibung	In dieser Datei ist ein Zeitintervall für die Suche nach dem Heimatnetz gespeichert.	
Datei	FID = '6F31'; Struktur: transparent, 1 Byte Zugriffe: READ: CHV 1; UPDATE: ADM	
Codierung	Zeitintervall für die Suche nach dem Heimatnetz in Minuten; Codierung nach GSM 02.11	

Tabelle 13.6 (Fortsetzung)

Beispiel '05 ' ⇒ Suche Heimatnetz alle 5 Minuten	
DF_{GSM}EF_{IMSI}	internationale Teilnehmernummer (<i>IMSI – international mobile subscriber identity</i>)
Beschreibung	In dieser Datei ist die internationale Teilnehmernummer gespeichert.
Datei	FID = '6F07'; Struktur: transparent, Dateigröße: 9 Byte Zugriffe: READ: CHV 1; UPDATE: ADM
Codierung	Byte 1: Länge der IMSI in Byte Byte 2, Bit 1 ... Bit 3: °100° Byte 2, Bit 4: Parität der IMSI, Codierung nach GSM 04.08 Byte 2, Bit 5 ... Bit 8: Digit 1 der IMSI Byte 3 ... 9: Digit 2 ... Digit 10 der IMSI IMSI = MCC MNC Lfd. Nr. des Netzbetreibers, BCD-codiert und bei Bedarf rechts mit 'F' aufgefüllt Codierung von MCC, MNC siehe EF _{PLMNsel}
Beispiel	'08 92 62 01 71 00 10 92 67' '08' ⇒ Länge ⇒ 8 Byte '9' '2 62' ⇒ MCC ⇒ Deutschland '01' ⇒ MNC ⇒ Deutschland D1 '71 00 10 92 67' ⇒ Lfd. Nr. des Netzbetreibers
DF_{GSM}EF_{Kc}	Schlüssel Kc
Beschreibung	In dieser Datei ist der Schlüssel Kc für die Datenverschlüsselung auf der Luftschnittstelle gespeichert.
Datei	FID = '6F20'; Struktur: transparent, Dateigröße: 9 Byte Zugriffe: READ: CHV 1; UPDATE: CHV 1
Codierung	Byte 1 ... 8: Schlüssel Kc; Byte 9, Bit 1 ... Bit 3: Folgenummer des Schlüssels
DF_{GSM}EF_{LOCI}	Ortsinformation (LOCI – location information)
Beschreibung	In dieser Datei werden Informationen über den aktuellen Ort des Mobiltelefons gespeichert.
Datei	FID = '6F7E'; Struktur: transparent, Dateigröße: 11 Byte Zugriffe: READ: CHV 1; UPDATE: CHV 1
Codierung	Byte 1 ... 4: TMSI (<i>temporary mobile subscriber identity</i>) Byte 5 ... 9: LAI (<i>location area information</i>) Byte 10: TMSI TIME (ab Phase 2 nicht mehr benutzt) Byte 11: Location Update Status (b3 ... b1 = 000: aktualisiert; b3 ... b1 = 001: nicht aktualisiert; b3 ... b1 = 010: verbotenes PLMN; b3 ... b1 = 011: verbotene Location Area Codierung nach GSM 04.08
Beispiel	'5F 40 96 46 62 F2 10 80 04 FF 00' '5F 40 96 46' ⇒ TMSI '62 F2 10 80 04' ⇒ LAI 'FF' ⇒ TMSI TIME ⇒ nicht benutzt '00' ⇒ Location Update Status ⇒ aktualisiert

Tabelle 13.6 (Fortsetzung)

DF_{GSM}EF_{LP}		Bevorzugte Sprache (LP – language preference)
Beschreibung		In dieser Datei ist eine Aufstellung der bevorzugten Sprachen für die Benutzerschnittstelle gespeichert.
Datei		FID = '6F05'; Struktur: transparent, Dateigröße: x Byte; x ≥ 1 Zugriffe: READ: immer; UPDATE: CHV 1
Codierung		nach GSM 03.38 Byte 1: höchstpriorie Sprache ... Byte x: niedrigstpriorie Sprache
Beispiele für Sprachen		gültig für GSM- Alphabet nach 03.38 '00': deutsch '01': englisch '02': italienisch '03': französisch '04': spanisch '05': holländisch '06': schwedisch '07': dänisch '08': portugiesisch '09': finnisch '0A': norwegisch '0B': griechisch '0C': türkisch '0D': ungarisch '0E': polnisch '0F': unspezifizierte Sprache
Beispiel		'00 01 03 05' '00' ⇒ höchstpriorie Sprache: deutsch '01' ⇒ zweithöchstpriorie Sprache: englisch '03' ⇒ dritthöchstpriorie Sprache: französisch '05' ⇒ niedrigstpriorie Sprache: holländisch
DF_{GSM}EF_{PHASE}		Phaseninformation
Beschreibung		In dieser Datei ist die von der SIM unterstützte Phase gespeichert.
Datei		FID = '6FAE'; Struktur: transparent, Dateigröße: 1 Byte Zugriffe: READ: immer; UPDATE: ADM
Codierung		Byte 1: 00 – Phase 1; 02 – Phase 2
Beispiel		'02' ⇒ Phase 2
DF_{GSM}EF_{PLMNsel}		bevorzugte Netzbetreiber (PLMNsel – public land mobile network selector)
Beschreibung		In dieser Datei ist eine Liste der bevorzugten Netzbetreiber gespeichert.
Datei		FID = '6F30'; Struktur: transparent, 3*n Bytes (n ≥ 8) Zugriffe: READ: CHV 1; UPDATE: CHV 1
Codierung		Byte 1 ... 3: PLMN der höchsten Auswahlpriorität Byte 4 ... 6: PLMN der zweithöchsten Auswahlpriorität Datenaufbau: 2 Byte MCC (Mobile Country Code) 1 Byte MNC (Mobile Network Code); Codierung nach GSM 04.08; BCD-codiert, high und low Nibble vertauscht; 'FFFFFF' ⇒ Eintrag nicht benutzt
Beispiele für MCC		262: Deutschland 208: Frankreich 234: Großbritannien 222: Italien 232: Österreich 310: USA

Tabelle 13.6 (Fortsetzung)

Beispiele für deutsche MNC	01: Deutschland D1 02: Deutschland D2 03: Deutschland E plus 07: Deutschland Viag Intercom Beispiel '62 F2 20 72 F0 10 32 F4 01 32 F2 30 32 F0 10 62 F2 10 62 F0 20 42 F0 10 22 F8 10' Rest 'FF' '62 F2' ⇒ MCC ⇒ '262' ⇒ Deutschland '20' ⇒ MCC ⇒ '02' ⇒ Deutschland D2 usw.
DF_{GSM}·EF_{PUCT}	Preis der Einheit und Währung (PUCT – price per unit and currency table)
Beschreibung	In dieser Datei ist für die aktuelle Übersicht der Telefonkosten der Preis pro Telefoneinheit sowie die Währung gespeichert.
Datei	FID = '6F41'; Struktur: transparent, Dateigröße: 5 Byte Zugriffe: READ: CHV 1; UPDATE: CHV 1 oder CHV 2
Codierung	Byte 1 ... 3: Währungscode, zeichencodiert mit GSM-Alphabet Byte 4, 5: Preis pro Einheit; Preis pro Einheit = EPPU * 10 ^{EX} EPPU: Elementary Price per Unit, EX: Exponent EPPU-Teil: B4.b1: 2 ⁴ B4.b2: 2 ⁵ B4.b3: 2 ⁶ B4.b4: 2 ⁷ B4.b5: 2 ⁸ B4.b6: 2 ⁹ B4.b7: 2 ¹⁰ B4.b8: 2 ¹¹ B5.b1: 2 ⁰ B5.b2: 2 ¹ B5.b3: 2 ² B5.b4: 2 ³ B5.b5: Vorzeichen des Exponenten 0: +; 1: - Exponent-Teil EX: B5.b6: 2 ⁰ B5.b7: 2 ¹ B5.b8: 2 ²
Beispiele	'44 45 4D 01 57' '44 45 4D' ⇒ Währungscode ⇒ "DEM" '01 57' = °0000 0001° °0101 0001° ⇒ Preis pro Einheit ⇒ 17 * 10 ⁻² = 0,17
DF_{GSM}·EF_{SPN}	Name des Diensteanbieters (SPN – service provider name)
Beschreibung	In dieser Datei ist der Name des Diensteanbieters gespeichert.
Datei	FID = '6F46'; Struktur: transparent, Dateigröße: 17 Byte Zugriffe: READ: immer; UPDATE: ADM
Codierung	Byte 1: Bedingungen für Displayanzeige '00' Anzeige des PLMN-Namens ist nicht erforderlich '01' Anzeige des PLMN-Namens ist erforderlich Byte 2 ... 17: Service Provider Name, codiert nach GSM 03.38, linksbündig und rechts bei Bedarf mit 'F' aufgefüllt
Beispiel	'01 50 72 6F 76 69 64 65 72 20 41' '01' ⇒ Anzeige des PLMN-Namens ist erforderlich '50 72 6F 76 69 64 65 72 20 41' ⇒ Name des Diensteanbieters ⇒ "Provider A"

Tabelle 13.6 (Fortsetzung)

DF_{GSM}EF_{SST}	Dienststeerkennungen (SST – SIM service table)
Beschreibung	In dieser Datei sind in einer Tabelle zusätzlich zum Sprachdienst mögliche und freigeschaltete Dienste angegeben.
Datei	FID = '6F38'; Struktur: transparent, Dateigröße: ≥ 2 Byte Zugriffe: READ: CHV 1; UPDATE: ADM
Codierung	Byte 1, Bit 1, 2: Service Nr. 1 Byte 1, Bit 3, 4: Service Nr. 2 Byte 1, Bit 5, 6: Service Nr. 3 Byte 1, Bit 7, 8: Service Nr. 4 Byte 2, Bit 1, 2: Service Nr. 5 usw. Codierung der Bits: b1, b3, b5, b7 = 1/0 Service möglich/nicht möglich b2, b4, b6, b8 = 1/0 Service aktiviert/nicht aktiviert
Beispiele für Service	Service Nr. 1: Abschalten der CHV Prüfung Service Nr. 2: Kurzrufnummern (ADN) Service Nr. 3: Festrufnummern (FDN) Service Nr. 4: Kurznachrichten (SMS) Service Nr. 18: Servicerufnummern (SDN) Service Nr. 35: Statusreport für Kurznachrichten Service Nr. 38: GPRS Service Nr. 39: Image (IMG)
Beispiel	'DF 3F DF FF 03' = °1101 1111° °0011 1111° °1101 1111° °1111 1111° °0000 0011° °11° ⇒ Abschalten der PIN-Prüfung möglich und aktiviert °11° ⇒ Kurzrufnummern möglich und aktiviert °01° ⇒ Festrufnummern möglich und nicht aktiviert °11° ⇒ Kurznachrichten möglich und aktiviert usw.
DF_{GSM}DF_{GRAPHICS}EF_{IMG}	Verweise auf Bilddateien (IMG – image)
Beschreibung	In dieser Datei sind Verweise auf Dateien mit Grafiken für die Anzeige auf dem Display des Mobiltelefons gespeichert.
Datei	FID = '4F20'; Struktur: linear fixed, 9*x + 2 Bytes Zugriffe: READ: CHV 1; UPDATE: ADM
Codierung	Byte 1: Anzahl der Verweise auf Bilddateien Byte 2 ... 10: Beschreibung des Verweis auf Bilddatei 1 Byte 11 ... 19: Beschreibung des Verweis auf Bilddatei 2 usw. 9*x + 2: RFU

Tabelle 13.6 (Fortsetzung)

Codierung der Verweise	Byte 1:	Breite des Bildes in Pixels
	Byte 2:	Höhe des Bildes in Pixels
	Byte 3:	Codierungsschema des Bildes
	Byte 4, 5:	FID des jeweiligen EF _{IMGData}
	Byte 6, 7:	Offset zu den Bilddaten in EF _{IMGData}
	Byte 8, 9:	Größe der Bilddaten in EF _{IMGData} in Byte
DF_{GSM}·DF_{GRAPHICS}· EF_{IMGDataX} Bilddaten (IMGData – image data)		
Beschreibung	In diese Dateien ist jeweils eine Grafik als Bitmap für die Anzeige auf dem Display des Mobiltelefons gespeichert.	
Datei	FID = '4Fxx'; Struktur: transparent, x Bytes Zugriffe: READ: CHV 1; UPDATE: ADM	
Codierung	Byte 1 ... x:	Bilddaten
DF_{TELECOM}·EF_{ADN} Kurzrufnummern (ADN – abbreviated dialling numbers)		
Beschreibung	In dieser Datei sind die Kurzrufnummern gespeichert. Jeder Record enthält einen Namen sowie die dazugehörige Rufnummer.	
Datei	FID = '6F3A'; Struktur: linear fixed, Recordgröße: n + 14 Byte Zugriffe: READ: CHV 1; UPDATE: CHV 1	
Codierung	Byte 1 .. n	Zeichencodierung des Namens, Codierung nach GSM 03.38
	Byte n + 1	Länge der BCD-codierten Rufnummer in Byte
	Byte n + 2	Art der Rufnummer, Codierung nach GSM 04.08 z. B.: '81' unbekannter Rufnummertyp, ISDN Rufnummernplan z. B.: '91' Internationaler Rufnummertyp, ISDN Rufnummernplan
	Byte n + 3 ... n + 12	Rufnummer BCD-codiert, oberes und unteres Nibble im Byte vertauscht
	Byte n + 13, n + 14	Zeiger auf zusätzliche Informationen zu diesem Eintrag in EF _{CCP} und EF _{EXT1} , i. d. R. nicht benutzt (d. h. 'FF')
nicht benutzte Byte werden auf 'FF' gesetzt		
Beispiel 1	Recordinhalt: '57 4F 4C 46 47 41 4E 47 FF FF FF FF FF FF FF 07 91 94 98 69 35 24 46 FF FF FF FF FF FF' '57 4F 4C 46 47 41 4E 47' ⇒ „Wolfgang“ 'FF FF FF FF FF FF FF FF' ⇒ nicht benutzt '07' ⇒ Länge der Rufnummer, d.h. 7 Byte '91' ⇒ Internationale Rufnummer, ISDN Rufnummernplan '94 98 69 35 24 46' ⇒ Rufnummer 49 89 96 53 42 64 'FF FF FF FF' ⇒ nicht benutzt 'FF FF' ⇒ EF _{CCP} und EF _{EXT1} nicht benutzt	

Tabelle 13.6 (Fortsetzung)

Beispiel 2		Recordinhalt: '57 4F 4C 46 47 41 4E 47 FF FF FF FF FF FF FF 07 81 80 99 56 43 62 F4 FF FF FF FF FF FF' '57 4F 4C 46 47 41 4E 47' ⇒ „Wolfgang“ 'FF FF FF FF FF FF FF FF' ⇒ nicht benutzt '07' ⇒ Länge der Rufnummer, d. h. 7 Byte '81' ⇒ unbekannter Rufnummertyp, ISDN Rufnummernplan '80 99 56 43 62 F4' ⇒ Rufnummer 089 96 53 42 64 'FF FF FF FF' ⇒ nicht benutzt 'FF FF' ⇒ EF _{CCP} und EF _{EXT1} nicht benutzt
DF _{TELECOM} •EF _{FDN}	Festruftnummern (FDN – fixed dialling numbers)	
Beschreibung	In dieser Datei können bei Bedarf Fest-Rufnummern gespeichert werden. Diese werden dann verwendet, wenn vom Teilnehmer nur bestimmte Rufnummern gewählt werden dürfen.	
Datei	FID = '6F3B'; Struktur: linear fixed, Recordgröße: n + 14 Byte Zugriffe: READ: CHV 1; UPDATE: CHV 2	
Codierung	analog EFADN	
Beispiel	siehe EFADN	
DF _{TELECOM} •EF _{LND}	letzte gewählte Rufnummer (LND – last number dialled)	
Beschreibung	In dieser Datei sind die letzten gewählten Rufnummern gespeichert.	
Datei	(optionale Datei) FID = '6F44'; Struktur cyclic, Recordgröße: n + 14 Byte Zugriffe: READ: CHV 1; UPDATE: CHV 1	
Codierung	analog EF _{ADN}	
DF _{TELECOM} •EF _{MSISDN}	ISDN Nummer der Mobilstation (MSISDN – mobil station ISDN number)	
Beschreibung	In dieser Datei ist die Rufnummer der Mobilstation gespeichert.	
Datei	FID = '6F40'; Struktur: linear fixed, Recordgröße: n + 14 Byte Zugriffe: READ: CHV 1; UPDATE: CHV 1	
Codierung	analog EFADN	
DF _{TELECOM} •EF _{SDN}	Service Rufnummern (SDN – service dialling numbers)	
Beschreibung	In dieser Datei sind Service-Rufnummern gespeichert. Dies sind beispielsweise Rufnummern für Telefon- oder Fahrplanauskunft.	
Datei	FID = '6F49'; Struktur: linear fixed, Recordgröße: n + 14 Byte Zugriffe: READ: CHV 1; UPDATE: ADM	
Codierung	analog EFADN	
DF _{TELECOM} •EF _{SMS}	Kurznachrichten (SMS – short message service)	
Beschreibung	Diese Datei gehört zum Kurznachrichtendienst. In ihr sind die Kurznachrichten vom Netzwerk und zum Netzwerk gespeichert.	
Datei	FID = '6F3C'; Struktur: linear fixed, Recordgröße: 176 Byte Zugriffe: READ: CHV 1; UPDATE: CHV 1	

Tabelle 13.6 (Fortsetzung)

Codierung	Byte 1: Status des jeweiligen Records: '00' = freier Record '01' = Nachricht vom Netzwerk kommend und gelesen '03' = Nachricht vom Netzwerk kommend und noch zu lesen '05' = Nachricht an das Netzwerk gesendet '07' = Nachricht zum Versenden an das Netzwerk Byte 2 – 176: Nachricht; Codierung nach GSM 03.40; nicht benutzte Bytes am Ende des Records werden auf 'FF' gesetzt								
Codierung von Nachrichten vom Netzwerk an das Mobiltelefon	Byte 2: Anzahl der Bytes der Rufnummer des SMSC inklusive Rufnummerntyp folgende 2 ... 12 Bytes: Rufnummer SMSC ('81' unbekannter Rufnummerntyp d.h. kein "+", '91' Internationaler Rufnummerntyp d. h. "+"), Daten sind nibbelweise vertauscht folgendes Byte: Steuerungsinformationen (i. d. R. '04') folgendes Byte: Anzahl der Digits der Rufnummer des Absenders exklusive Rufnummerntyp folgende 2 ... 12 Byte: Rufnummer Absender, Daten sind nibbelweise vertauscht folgendes Byte: Protokollkennzeichen ('00' = Textnachricht) folgendes Byte: Datencodierung ('00' = GSM Standard-Alphabet) folgende 7 Byte: Zeitstempel SMSC, Daten sind nibbelweise vertauscht, Jahr Monat Tag Stunde Minuten Sekunden Zeitzone ('00' = GMT) folgendes Byte: Anzahl der Zeichen der Nachricht folgende 1 ... 140 Bytes: Nachricht (bei GSM-Standard-Alphabet ist der Textteil komprimiert, d. h. 7 Bits werden durchgehend in Bytes gepackt)								
Codierung von Nachrichten vom Mobiltelefon an das Netzwerk	Byte 2: Anzahl der Bytes der Rufnummer des SMSC folgende 2 ... 12 Bytes: Rufnummer SMSC ('81' unbekannter Rufnummerntyp d. h. kein "+", '91' Internationaler Rufnummerntyp d. h. "+"), Daten sind nibbelweise vertauscht folgendes Byte: Relativzeit vom Mobiltelefon (i. d. R. 'FF') folgendes Byte: Nachrichtenreferenz folgende 2 ... 12 Byte: Rufnummer des Ziels, Daten sind nibbelweise vertauscht folgendes Byte: Protokollkennzeichen ('00' = Textnachricht) folgendes Byte: Datencodierung ('00' = GSM Standard-Alphabet) folgendes Byte X: Gültigkeitsdauer t der Nachricht <table><tr><td>1 ... 143</td><td>$t = (X + 1) * 5 \text{ min}$</td></tr><tr><td>144 ... 167</td><td>$t = 12 \text{ h} + (X - 143) * 30 \text{ min}$</td></tr><tr><td>168 ... 196</td><td>$t = (X - 166) * 1 \text{ Tag}$</td></tr><tr><td>197 ... 255</td><td>$t = (X - 192) * 1 \text{ Woche}$</td></tr></table> folgendes Byte: Anzahl der Zeichen der Nachricht folgende 1 ... 140 Bytes: Nachricht (bei GSM-Standard-Alphabet ist der Textteil komprimiert, d. h. 7 Bits werden durchgehend in Bytes gepackt)	1 ... 143	$t = (X + 1) * 5 \text{ min}$	144 ... 167	$t = 12 \text{ h} + (X - 143) * 30 \text{ min}$	168 ... 196	$t = (X - 166) * 1 \text{ Tag}$	197 ... 255	$t = (X - 192) * 1 \text{ Woche}$
1 ... 143	$t = (X + 1) * 5 \text{ min}$								
144 ... 167	$t = 12 \text{ h} + (X - 143) * 30 \text{ min}$								
168 ... 196	$t = (X - 166) * 1 \text{ Tag}$								
197 ... 255	$t = (X - 192) * 1 \text{ Woche}$								

Tabelle 13.6 (Fortsetzung)

Beispiel für eine SMS vom Netzwerk an das Mobiltelefon	'01 07 91 94 71 01 67 05 00 04 0C 91 94 71 71 46 53 42 00 00 00 60 52 31 63 15 00 17 C8 A0 93 28 AC 0E 91 20 62 51 0A 1A 22 93 D0 65 50 4A 2D 3A 01' restlicher Record ist 'FF' '01' ⇒ Nachricht vom Netz kommend und gelesen '07' ⇒ Anzahl der Bytes der Rufnummer des SMSC inklusive Rufnummerntyp '91 94 71 01 67 05 00' ⇒ +49 17 10 76 50 00 Rufnummer SMSC '04' ⇒ keine weiteren Nachrichten '0C' ⇒ 12 ⇒ Anzahl der Digits der Rufnummer des Absenders exklusive Rufnummerntyp: 12 '91 94 71 71 46 53 42' ⇒ +49 17 17 64 35 24, Rufnummer Absender '00' ⇒ Textnachricht '00' ⇒ GSM Standard-Alphabet '00 60 52 31 63 15 00' ⇒ 00 06 25 13 36 51 00 Zeitstempel SMSC ⇒ 25.06.00 13:36:51 Zeitzone 0 (= GMT) '17' ⇒ 23 ⇒ Anzahl der Zeichen der Nachricht: 23 'C8 A0 93 28 AC 0E 91 20 62 51 0A 1A 22 93 D0 65 50 4A 2D 3A 01' ⇒ Nachricht: "Handbuch der Chipkarten"
Beispiel für eine SMS vom Mobiltelefon an das Netzwerk	'07 02 81 F0 11 FF 00 81 00 00 00 08 D7 27 D3 78 0C 3A 8F FF' restlicher Record ist 'FF' '07' ⇒ Nachricht zum Versenden an das Netzwerk '02' ⇒ Anzahl der Bytes der Rufnummer des SMSC inklusive dieser Längenangabe ☎ '81' ⇒ unbekannte Rufnummer ☎ 'F0' ⇒ Steuerungsinformation '11' ⇒ Relativzeit vom Mobiltelefon 'FF' ⇒ Nachrichtenreferenz ☎ '00' ⇒ Länge der Rufnummer des Ziels: 0 ☎ '81' ⇒ unbekannte Rufnummer ☎ '00' ⇒ Textnachricht '00' ⇒ GSM Standard-Alphabet '00' ⇒ Gültigkeitsdauer ☎ '08' ⇒ Anzahl der Zeichen der Nachricht: 8 'D7 27 D3 78 0C 3A 8F FF' ⇒ Nachricht: "WOLFGANG" Anmerkung 1: Der Aufbau des Datensatzes ist abhängig von der Implementierung im jeweiligen Mobiltelefon und nicht allgemein gültig. Anmerkung 2: Nach dem Lesen dieses SMS-Datensatzes aus der SIM werden die mit ☎ gekennzeichneten Datenelemente vor dem Absenden vom Mobiltelefon ergänzt. Nachdem die Nachricht an das Netzwerk verschickt wurde, wird das erste Byte dieses Datensatzes von '07' nach '05' geändert.
DF _{TELECOM} · EF _{SMSP}	Einstellungen für die Kurznachrichten (<i>SMSP – short message service parameters</i>)
Beschreibung	Diese Datei gehört zum Kurznachrichtendienst. In ihr sind die Einstellungen für das Versenden von Kurznachrichten gespeichert.
Datei	FID = '6F42'; Struktur: linear fixed, Recordgröße: 28 + n Byte Zugriffe: READ: CHV 1; UPDATE: CHV 1

Tabelle 13.6 (Fortsetzung)

DF _{TELECOM} EF _{SMSS}		Zustand der gespeicherten Kurznachrichten (SMSS – short message service status)
Beschreibung		Diese Datei gehört zum Kurznachrichtendienst. In ihr ist der Zustand der gespeicherten Kurznachrichten gespeichert.
Datei		FID = '6F43'; Struktur: linear fixed, Recordgröße: 2+n Byte Zugriffe: READ: CHV 1; UPDATE: CHV 1
Codierung		Byte 1: letzte verwendete Referenznummer der SMS-Nachricht nach GSM 03.40 Byte 2: b1 = 0: Speicherkapazität für SMS in der SIM ist nicht vorhanden; b1 = 1: Speicherkapazität für SMS in der SIM ist vorhanden; b2 ... b7: RFU, sind auf 1 gesetzt
Beispiel		'70 FF' '70' ⇒ letzte verwendete Referenznummer der SMS-Nachricht 'FF' ⇒ Speicherkapazität im SIM vorhanden

Authentisierung der SIM

Neben der Speicherung von Daten ist eine Hauptaufgabe der SIM die Authentisierung gegenüber dem GSM-Netzwerk. Es handelt sich dabei um eine einseitige Authentisierung des SIMs durch das Hintergrundsystem. Die SIM prüft also nicht, ob das Hintergrundsystem echt ist, sondern nur das Hintergrundsystem, ob die SIM echt ist. Bei bestätigter Echtheit der SIM weiß der Netzbetreiber, dass er das Gespräch gegenüber dem Eigentümer des Mobiltelefons abrechnen kann. Diese einseitige Authentisierung hat jedoch den Nachteil, dass der Benutzer des Mobiltelefons nicht sicher sein kann, dass er auch mit einem echten Netz und nicht mit einem fingierten Netz verbunden ist. Aus diesem Grund ist es möglich, mit einem entsprechendem technischen Gerät, das IMSI-Catcher genannt wird, auch ohne Kenntnis der geheimen Schlüssel Gespräche abzuhören.

Das Funktionsprinzip eines IMSI-Catchers beruht darauf, dass er sich durch Aufbau einer eigenen Funk-Zelle als fingierte Basisstation der Luftschnittstelle dazwischenschaltet, er sich dem Mobiltelefon gegenüber als Basisstation ausgibt und gegenüber der echten Basisstation als Mobiltelefon. Bei einer gegenseitigen Authentisierung und anschließender Verschlüsselung aller Gesprächsdaten zwischen SIM und Hintergrundsystem wäre dieser Angriff nicht möglich.

Die Identifizierung des SIM geschieht mit einer im gesamten GSM-System einzigartigen Nummer, die maximal 8 Byte lang ist und IMSI (*international mobile subscriber identity*) genannt wird. Anhand dieser Nummer kann der Teilnehmer vom System weltweit in allen GSM-Netzen identifiziert werden. Um die Identität des Teilnehmers im Netz so vertraulich wie möglich zu machen, wird, wann immer die Möglichkeit besteht, statt der IMSI eine TMSI (*temporary mobile subscriber identity*) verwendet. Diese wird vom Besuchsregister VLR erzeugt und ist damit auch nur innerhalb eines Teils des jeweiligen GSM-Netzes gültig. Zusammen mit der LAI (*location area information*) ist die TMSI jedoch im gesamten GSM-Netz eindeutig. Für alle weiteren Identifizierungsabläufe wird nach Vergabe der TMSI nur noch diese verwendet. Die Zuordnung IMSI zur TMSI wird für die laufende Verwendung im Besuchsregister VLR gespeichert. Sollte im Besuchsregister VLR die TMSI nicht bekannt sein, so muss in diesem Ausnahmefall die IMSI im Klartext zur Identifizierung des Teilnehmers über die Luftschnittstelle übertragen werden.

Aus der IMSI können die kartenindividuellen Schlüssel für Authentisierung und Verschlüsselung der Daten auf der Luftschnittstelle abgeleitet werden. Die Verschlüsselung der Daten auf der Luftschnittstelle kann allerdings nicht von der SIM geleistet werden, da die Berechnungs- und Übertragungskapazität einer Chipkarte nicht für die Echtzeitverschlüsselung von Sprachdaten ausreicht. Stattdessen errechnet die SIM einen abgeleiteten und temporären Schlüssel für die Übertragungsverschlüsselung und gibt ihn an das Mobile Equipment weiter. Dieses ist mit einer leistungsfähigen Verschlüsselungseinheit in Form eines Signalprozessors ausgestattet, welche die Sprachdaten auf der Luftschnittstelle in Echtzeit ver- und entschlüsseln kann. Die verschlüsselten Daten der Luftschnittstelle werden üblicherweise durch die Feststationsteuerung BSC wieder in Klartext entschlüsselt.

Möchte ein Teilnehmer nun ein Gespräch führen, stellt seine Mobile Station eine Verbindung zu der am besten zu empfangenden Basisstation her und übergibt dieser aus dem Speicher der SIM die TMSI zusammen mit der LAI oder im Ausnahmefall die IMSI. Ist der Teilnehmer in seinem Heimatnetz, wird vom Authentisierungszentrum AuC ein Triple mit Authentisierungs- und Verschlüsselungsdaten erzeugt. Dieser Datensatz beinhaltet den Schlüssel für die Datenverschlüsselung auf der Luftschnittstelle *Kc* (*ciphering key*), eine Zufallszahl *RAND* und das dazugehörige Ergebnis *SRES* (*signed response*). Diese Vorgehensweise hat den Vorteil, dass der geheime Schlüssel *Ki* (*individual key*) und die z. T. geheimen Authentisierungsalgorithmen nie das Authentisierungszentrum AuC verlassen müssen. Dieses Triple wird anschließend an das Heimatregister HLR weitergereicht.

Hält sich nun das Mobiltelefon im Heimatnetz auf, wird das Triple *Kc*, *RAND* und *SRES* zum entsprechenden Besucherregister VLR gesendet. Dort wird dann durch die Mobilvermittlungsstelle MSC das Ergebnis *SRES* von der SIM angefordert und anschließend mit dem vom AuC erhaltenen *SRES* verglichen. Sind beide Werte gleich, so ist die SIM authentisiert und es kann mit der Verschlüsselung der Daten auf der Luftschnittstelle mittels Kryptoalgorithmus A5 und dazugehörigem Schlüssel *Kc* begonnen werden.

Befindet sich jedoch das zu authentisierende Mobiltelefon nicht im Heimatnetz so wird das Triple zum entsprechenden Fremdnetz übertragen und kann dort analog wie im Heimatnetz verwendet werden. In diesem Fall wird die Raffinesse dieses Authentisierungs- und Verschlüsselungsschemas deutlich. Die Kryptoalgorithmen A3 und A8 sind nämlich netzbetreiberspezifisch und könnten in einem Fremdnetz selbst bei Kenntnis der geheimen Schlüssel nicht berechnet werden. Lediglich der Kryptoalgorithmus A5 für die Verschlüsselung der Daten der Luftschnittstelle ist einheitlich im GSM-System, sodass mit Kenntnis von *Kc* diese auch entsprechend kryptografisch gesichert werden kann.

Die im GSM-System gebrauchten Kryptoalgorithmen sind in aller Regel geheim, was in diesem System die einzige Abweichung von Kerckhoffs Prinzip¹¹ darstellt. Alle anderen Informationen über das System sind öffentlich zugänglich. Für die netzbetreiberspezifischen Kryptoalgorithmen A3 und A8 wurde ursprünglich oft ein Algorithmus namens COMP 128 benutzt, welcher jedoch im Jahr 1998 gebrochen wurde, da er eine zu geringe Schlüssellänge hatte. Im Nachhinein war dies im Übrigen ein Beleg für die Sinnhaftigkeit von Kerckhoffs Prinzip, da die zu kurze Schlüssellänge wohl für Kryptologen war. Der Kryptoalgorithmus COMP 128 wird auch heute noch eingesetzt, allerdings in verbesserter Form und nun COMP 128-2 genannt. Der bei GSM einheitliche Kryptoalgorithmus A5 ist

¹¹ Siehe auch Abschnitt 4.7 „Kryptologie“.

eine Stromchiffre (*stream cipher*) und setzt sich aus drei LFSR (*linear feedback shift register*) der Längen 19, 22 und 23 zusammen [Anderson 01], welche über die TDMA-Rahmennummer weitergeschaltet werden.

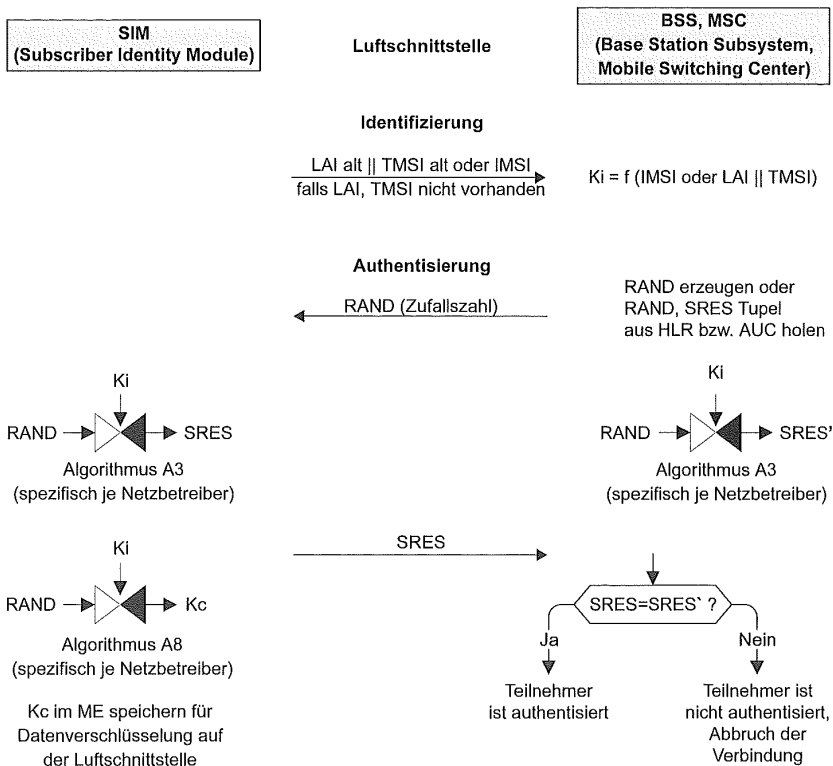


Bild 13.13 Die Identifizierung und anschließende Authentifizierung der SIM durch das GSM-Hintergrundsystem mittels der netzbetreiberspezifischen Kryptoalgorithmen A3 und A8. Der Schlüssel K_c wird zur späteren Verschlüsselung der Daten auf der Luftschnittstelle zwischen Mobile Station und Base Station benutzt.

Abläufe beim Ein- und Ausschalten des Mobiltelefons

Im Folgenden werden überblicksartig und mit starkem Fokus auf die SIM die Abläufe beim An- bzw. Ausschaltens des Mobiltelefons aufgezeigt.

Beim Einschalten des Mobiltelefons wird nach den Hardware-Selbsttestroutinen das mehrere Megabyte große Betriebssystem hochgefahren. Zur Ablenkung des ungeduldigen Benutzers wird diese mehrere Sekunden dauernde Zeitspanne oft durch mehr oder weniger interessante Animationen auf dem Display überbrückt. Ist das Betriebssystem dann voll aktiv, so ist einer der nächsten Schritte die Initiierung der Einschaltsequenz für das SIM. Nach der Einschaltsequenz folgen diverse Maßnahmen zur Einstellung der optimalen Über-

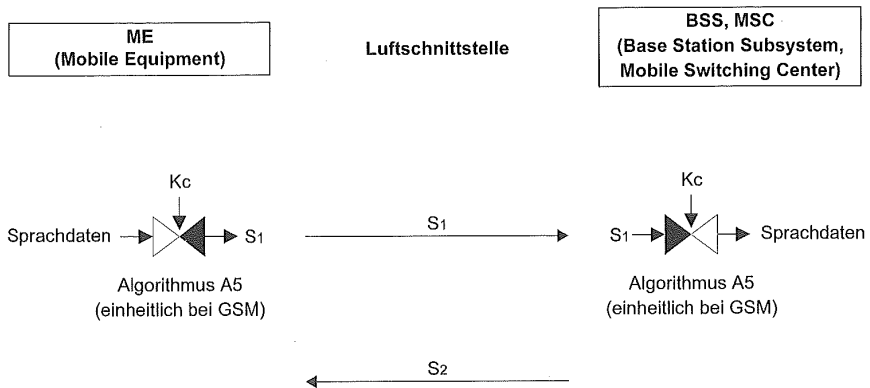


Bild 13.14 Die Verschlüsselung der Daten auf der Luftschnittstelle zwischen Mobile Station und Base Station benutzt mittels des Kryptoalgorithmus A5 und des geheimen Schlüssels K_c . Diesem Ablauf muss eine Identifizierung und Authentisierung der SIM durch das GSM-Hintergrundsystem vorausgehen.

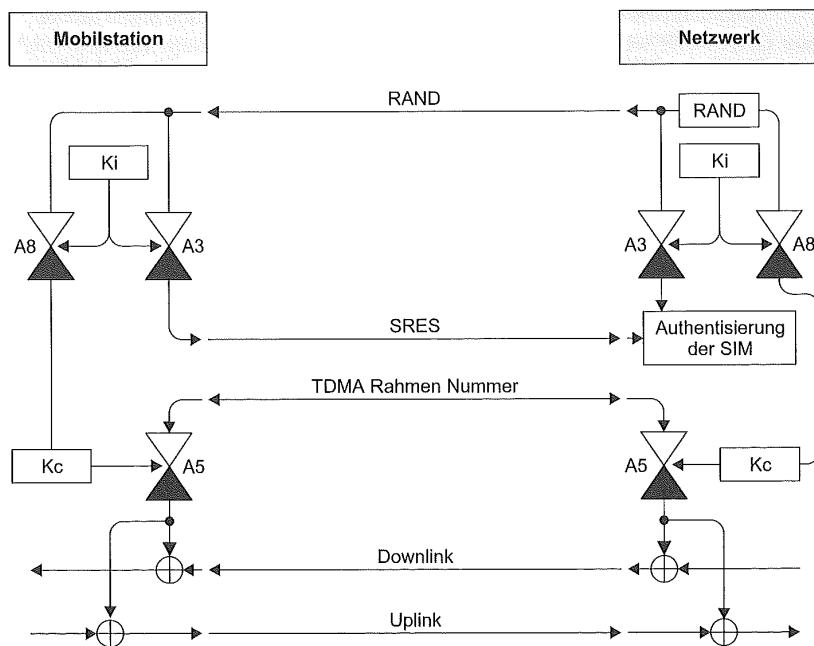


Bild 13.15 Funktionaler Überblick über die kryptografischen Funktionen von SIM, Mobile Equipment und Hintergrundsystem bei GSM.

tragungsparameter, wie ATR-Analyse und PPS-Prozedur.¹² Anschließend ist es mittlerweile üblich, eine so genannte virtuelle SIM im Speicher des Mobiltelefons aufzubauen. Dazu liest das Mobiltelefon viele Informationen, wie etwa die Kurzruftnummern, aus den Dateien der SIM und trägt sie in entsprechende Datenfelder des Mobiltelefons ein. Der Zweck dieser Maßnahme dient der Gewährleistung eines schnellen Lese- und Schreibzugriffs auf Daten der SIM. Dies wäre sonst durch die niedrige Übertragungsgeschwindigkeit zwischen Mobiltelefon und SIM sowie durch die zeitintensiven EEPROM-Schreibzugriffe nicht gewährleistet. Deshalb arbeitet das Mobiltelefon üblicherweise vor allem mit den in ihrem Speicher befindlichen Kopien der Daten der SIM. Selbstverständlich ist dieses Vorgehen nicht mit allen Daten der SIM möglich, so muss etwa eine PIN-Prüfung oder Authentisierung immer in Verbindung mit der SIM durchgeführt werden, da die für diese Aktionen relevanten Daten der SIM diese nie verlassen dürfen.

Ein die Lebenserwartung der SIM erheblich steigender Nebeneffekt beim Einsatz von virtuellen SIMs ist die Tatsache, dass viele EEPROM-Schreibzugriffe schlichtweg entfallen. Damit wird der Beanspruchung von bestimmten Dateien auf der SIM durch häufige Schreibzugriffe erheblich reduziert. Ein typisches Beispiel dafür ist die Datei EF_{LOC1} mit Informationen über den aktuellen Aufenthaltsort des Mobiltelefons. Die EEPROM-Zellen dieser Datei werden bei Mobiltelefonen mit häufigen Zellenwechsel besonders stark gestresst, sodass sie auch das Attribut „high update activity“ hat. Werden diese Daten vor allem in den RAM-Zellen des Mobiltelefons aktualisiert, so ist das Problem einer zu hohen Anzahl von Schreibzugriffen auf das EEPROM der SIM nicht mehr von großer Bedeutung.

Die Daten der virtuellen SIM im Speicher des Mobiltelefons werden nach kritischen Operationen in die SIM zurückgeschrieben, sodass diese anschließend wieder über aktuelle Daten verfügt. Oft wird dies vom Endgerät asynchron durch eine Task des Betriebssystems niedriger Priorität durchgeführt, sodass der Benutzer davon nichts bemerkt. Diese zeitlichen Abgleiche an kritischen Punkten sind auch deshalb relevant, weil bei einem plötzlichen Stromausfall, wie er beispielsweise durch Abnahme des Akkus entsteht, das SIM immer annähernd aktuelle Daten in seinem EEPROM aufbewahren sollte. Es wäre in diesem Beispiel schließlich ziemlich ärgerlich, wenn in diesem Fall plötzlich alle seit dem letzten Einschalten des Mobiltelefons mühevoll eingegebenen Rufnummern verloren gingen.

Die für den Benutzer sichtbare Aktion beim Abschaltung des Mobiltelefons ist meist nur eine kurze Zeichentricksequenz auf dem Display. Nebenbei werden jedoch alle Daten der virtuellen SIM in die physische SIM geschrieben, sodass diese wieder auf dem aktuellen Stand ist. Dann folgt eine Abschaltsequenz für die SIM und anschließend wird das Betriebssystem des Mobiltelefons heruntergefahren.

Die im vorangehenden Text beschriebenen Abläufe und Mechanismen sind nicht Teil der GSM-Spezifikationen. Deshalb sind sie in den verschiedenen Mobiltelefonen auch durchaus unterschiedlich realisiert und sollten nur als eine mögliche und technisch zweckmäßige Verwirklichung verstanden werden. Auch darf man gerade bei GSM nicht übersehen, dass durchaus 10 Jahre alte Mobiltelefone noch immer im Einsatz sind. Bei ihnen kann man mit Sicherheit davon ausgehen, dass sie über keine virtuelle SIM verfügen sondern alle Lese- und Schreibzugriffe direkt auf die SIM durchreichen.

¹² Siehe auch Abschnitte 6.2 „Answer to Reset – ATR und Abschnitt 6.3 Protocol Parameter Selection – PPS“.

Tabelle 13.7 Typische Aktivitäten eines Mobiltelefons im Zusammenhang mit der SIM in zeitlich korrekter Reihenfolge. Die dargestellten Aktivitäten und Kommandosequenzen entsprechen einem typischen Mobiltelefon, wobei aber zu beachten ist, dass diesbezügliche Details in der Regel von den GSM-Spezifikationen dem Mobiltelefonhersteller überlassen sind. Die im Mobiltelefon verwendete SIM besitzt bis auf die Dateien für Kurzrufnummern und Kurznachrichten im Wesentlichen nur die zum Telefonieren notwendige Funktionalität. Bei einem erweiterten Funktionsumfang der SIM bzw. des Mobiltelefons würden die Aktivitäten der beiden Kommunikationspartner entsprechend zunehmen.

Benutzer schaltet das Mobiltelefon ein	
führe Einschaltsequenz des SIM aus	Einschalten der SIM
empfangen ATR	ermittle das Vorhandensein der SIM und die Parameter des Übertragungsprotokolls
führe PPS aus	schalte bei Bedarf Parameter des Übertragungsprotokolls um
Das Mobiltelefon hat nun eine funktionierende Kommunikationsverbindung zur SIM errichtet.	
SELECT DF _{GSM} GET RESPONSE	selektiere Verzeichnis GSM und hole Informationen über das Verzeichnis
SELECT EF _{PHASE} READ BINARY	selektiere und lese EF mit Phaseninformation
SELECT EF _{LP} GET RESPONSE READ BINARY	selektiere, hole Informationen über den Dateiaufbau und lese EF mit bevorzugter Sprache
Benutzer gibt PIN ein	
VERIFY CHV STATUS	prüfe die PIN und frage anschließend den Stand der Fehlbedienungsanzahl ab
SELECT EF _{SST} GET RESPONSE READ BINARY	selektiere, hole Informationen über den Dateiaufbau und lese EF mit Dienststeerkennungen
TERMINAL PROFILE	übergebe der SIM Informationen über die Eigenschaften des Mobiltelefons (dies ist wichtig für SIM Toolkit-Anwendungen)
SELECT MF	selektiere Wurzelverzeichnis
SELECT EF _{ICCID} GET RESPONSE READ BINARY	selektiere, hole Informationen über den Dateiaufbau und lese EF mit Identifikationsnummer der Chipkarte
SELECT DF _{GSM}	selektiere Verzeichnis GSM
SELECT EF _{IMSI} GET RESPONSE READ BINARY	selektiere, hole Informationen über den Dateiaufbau und lese EF mit internationaler Teilnehmernummer
SELECT EF _{AD} GET RESPONSE READ BINARY	selektiere, hole Informationen über den Dateiaufbau und lese EF AD (<i>administrative data</i>) mit Verwaltungsdaten der Mobilstation

Tabelle 13.7 (Fortsetzung)

SELECT EF _{LOCI} READ BINARY	selektiere und lese EF mit Ortsinformation
SELECT EF _{KC} READ BINARY	selektiere und lese EF mit Schlüssel Kc
SELECT EF _{BCCH} READ BINARY	selektiere und lese EF BCCH (<i>broadcast control channels</i>) mit netzwerkspezifischen Informationen
SELECT EF _{FPLMN} READ BINARY	selektiere und lese EF mit verbotenen Netzbetreibern
SELECT EF _{HPLMN} READ BINARY	selektiere und lese EF mit Zeitintervall für die Suche nach dem Heimatnetz
SELECT DF _{TELECOM}	selektiere Verzeichnis Telekom
SELECT EF _{SMSS} GET RESPONSE READ BINARY	selektiere, hole Informationen über den Dateiaufbau und lese EF mit Zustand der gespeicherten Kurznachrichten
SELECT EF _{SMSP} GET RESPONSE READ BINARY	selektiere, hole Informationen über den Dateiaufbau und lese EF mit den Einstellungen für die Kurznachrichten
SELECT EF _{SMS} GET RESPONSE n * READ RECORD	selektiere, hole Informationen über den Dateiaufbau und lese alle n Records des EFs mit den Kurznachrichten
SELECT EF _{ADN} GET RESPONSE n * READ RECORD	selektiere, hole Informationen über den Dateiaufbau und lese alle n Records des EFs mit den Kurzzurufnummern
Das Mobiltelefon ist nun bereit für Gesprächsaufbau oder Datenübertragung.	
Benutzer führt ein Gespräch	
SELECT DF _{GSM}	selektiere Verzeichnis GSM
RUN GSM ALGORITHM GET RESPONSE	führe die Authentisierung der SIM gegenüber dem Hintergrundsystem aus
SELECT EF _{KC} UPDATE BINARY	selektiere ED und schreibe in das EF den aktualisierten Schlüssel Kc
SELECT EF _{LOCI} UPDATE BINARY	selektiere EF und schreibe in das EF die aktualisierte Ortsinformation
SELECT EF _{BCCH} UPDATE BINARY	selektiere und schreibe in das EF BCCH (<i>broadcast control channels</i>) netzwerkspezifische Informationen
Benutzer schaltet das Mobiltelefon aus	
SELECT EF _{LOCI} UPDATE BINARY	selektiere EF und schreibe in das EF die aktualisierte Ortsinformation
SELECT EF _{BCCH} UPDATE BINARY	selektiere und schreibe in das EF BCCH (<i>broadcast control channels</i>) netzwerkspezifische Informationen

Beispiel für eine typische Kommandosequenz

Das Lesen von Rufnummern aus einem EF mit Record-orientierter Struktur, wie beispielsweise dem EF_{ADN}, ist ein praxisnahes Beispiel für eine typische Kommandosequenz. Als erster Schritt ist die entsprechende Datei im richtigen Verzeichnis zu selektieren. Da die Anzahl der Records der Datei dem Netzbetreiber überlassen ist, muss als Nächstes die Dateigröße ermittelt werden, und daraus wird zusammen mit der Länge eines Records die Anzahl der Einträge errechnet. Anschließend können mit READ RECORD und der Nummer des jeweiligen Records alle Datensätze mit den Rufnummern gelesen werden. Bild 13.16 verdeutlicht dies im Detail.

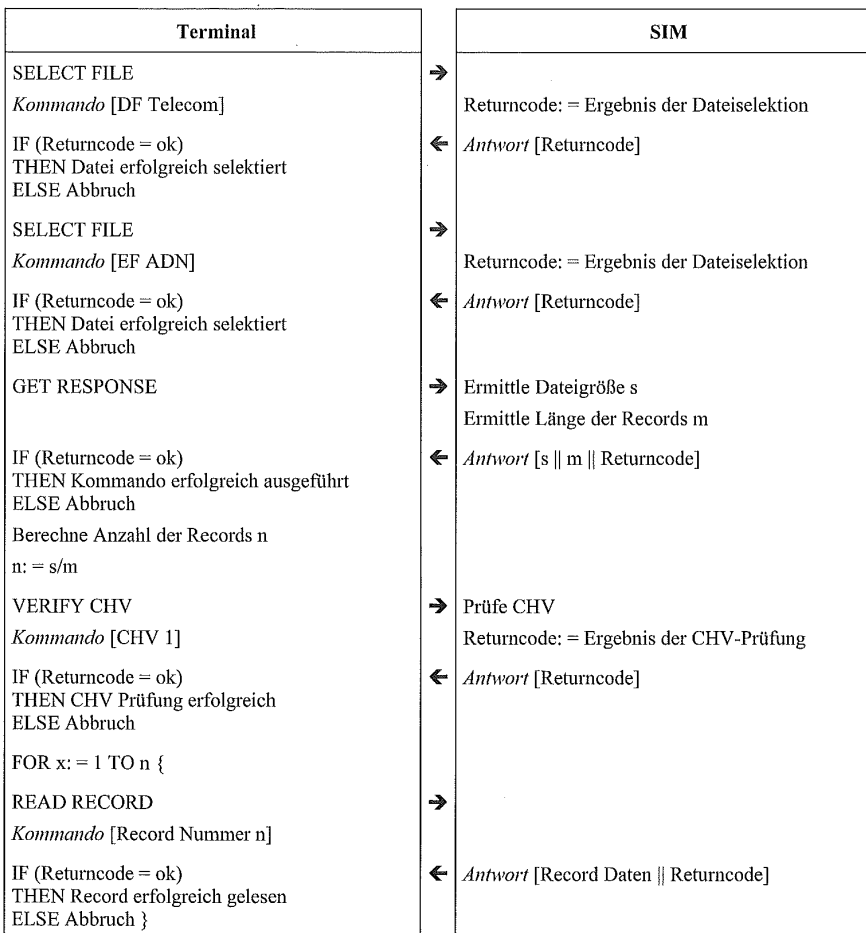


Bild 13.16 Prinzipieller Ablauf der Kommandos beim Lesen der Kurzzufnummern aus dem EF_{ADN}. Der beschriebene Ablauf enthält nur die wesentlichen Aspekte und es ist stets der Gutfall angenommen.

SIM Application Toolkit

Die ursprünglichen Spezifikationen im GSM-System sahen die GSM-Karte lediglich als ein vom Mobiltelefon unabhängiges Medium zur Identifizierung des Benutzers mittels PIN und als Authentisierungstoken im Sinne der Abrechnungssicherheit. Im Laufe der Jahre trat aber immer das Bedürfnis in den Vordergrund, die GSM-Karte noch für weitere Funktionen insbesondere für Mehrwertdienste zu nutzen. So ist ein Mobiltelefon auch ein adäquates Medium zur Abfrage des Kontostands oder zum Empfang von interessanten Mitteilungen wie Fußballergebnisse oder Tageshoroskope. Es war jedoch mit der bestehenden Funktionalität der GSM-Karte nicht möglich, diese Mehrwertdienste (*value added service* – VAS) technisch zu realisieren. Die Reaktion darauf war die Erstellung der GSM-Spezifikation 11.14, SIM Application Toolkit (SAT), welche 1996 in der ersten Version von ETSI veröffentlicht wurde.

Das SIM Application Toolkit ermöglicht der SIM direkten Zugriff auf Funktionen der Mobilstation wie Ansteuerung des Displays, Abfrage der Tastatur, Versenden von Kurznachrichten und einige weitere Funktionen, die im Zusammenhang mit einer Zusatzanwendung durchgeführt werden müssen. Letzten Endes ist das SIM Application Toolkit ein Baukasten, mit dem sich nahezu beliebige Anwendungen auf einer SIM realisieren lassen.

Tabelle 13.8 Die in GSM 11.14 festgelegten proaktiven Chipkarten-Kommandos der SIM für das SIM Application Toolkit. Man beachte, dass die aufgelisteten Kommandos in diesem Fall von der Chipkarte an das Terminal gesendet werden und nicht, wie üblich, vom Terminal an die Chipkarte. Die Verwendung bestimmter Kommandos ist abhängig von der Hardwareausstattung des Mobile Equipment.

Kommando	Kurzbeschreibung
Benutzerschnittstelle	
DISPLAY TEXT	Anzeige eines übergebenen Textes oder Icons auf dem Display der Mobilstation.
GET INKEY	Anzeige eines übergebenen Textes oder Icons auf dem Display des Mobile Equipments mit anschließender Abfrage eines Zeichens von der Tastatur.
GET INPUT	Anzeige eines übergebenen Textes oder Icons auf dem Display des Mobile Equipments mit anschließender Abfrage eines oder mehrerer Zeichen von der Tastatur.
LANGUAGE NOTIFICATION	Bekanntmachung an das Mobile Equipment über die vom SIM Application Toolkit in den Textfeldern verwendete Sprache.
PLAY TONE	Aufforderung an das Mobile Equipment, einen Ton auszugeben.
SELECT ITEM	Übergabe einer Auswahlliste an das Mobile Equipment mit der Aufforderung, dass der Benutzer einen Punkt auswählen soll.
SET UP IDLE MODE TEXT	Anzeige eines übergebenen Textes oder Icons zur Anzeige auf dem Display der Mobile Equipments, wenn dieses angeschaltet ist, aber nicht benutzt wird.
SET UP MENU	Übergabe einer Menüliste an das Mobile Equipment mit der Aufforderung, diese in die Menüstruktur des Mobile Equipment zu integrieren.

Tabelle 13.7 (Fortsetzung)

Kommando	Kurzbeschreibung
zweites Kartenterminal	
GET READER STATUS	Abfrage des Status eines zusätzlichen Kartenterminals in der Mobilstation.
PERFORM CARD APDU	Senden einer APDU an die Chipkarte, die sich in einem zusätzlichen Kartenterminal in der Mobilstation befindet.
POWER OFF CARD	Abschalten der Chipkarte, die sich in einem zusätzlichen Kartenterminal in der Mobilstation befindet.
POWER ON CARD	Anschalten der Chipkarte, die sich in einem zusätzlichen Kartenterminal in der Mobilstation befindet.
Netzwerkschnittstelle	
CLOSE CHANNEL	Aufforderung an das Mobile Equipment zum Schließen eines Datenkanals.
GET CHANNEL STATUS	Aufforderung an das Mobile Equipment, den Status eines Datenkanals zurückzugeben.
OPEN CHANNEL	Aufforderung an das Mobile Equipment zum Öffnen eines Datenkanals.
RECEIVE DATA	Aufforderung an das Mobile Equipment zum Empfang von Daten über einen geöffneten Datenkanal.
RUN AT COMMAND	Übergabe eines AT-Kommandos an das Mobile Equipment und dortiges Ausführen mit anschließender Übergabe des Ergebnisses an die SIM.
SEND DATA	Aufforderung an das Mobile Equipment zum Senden von Daten über einen geöffneten Datenkanal.
SEND DTMF	Senden eines DTMF Tons während einer laufenden Sprachverbindung.
SEND SHORT MESSAGE	Senden einer Kurznachricht.
SEND SS	Senden einer SS-Nachricht (<i>subplementary service</i>), welche eine Steuersequenz an das Netzwerk ist.
SEND USSD	Senden einer USSD-Nachricht (<i>unstructured supplementary services data</i>), welche zum Versand von beliebigen Daten benutzt werden kann.
SET UP CALL	Aufbau einer Verbindung.
Sonstige	
MORE TIME	Anfrage an das Mobile Equipment, der SAT-Anwendung mehr Zeit für die Abarbeitung zur Verfügung zu stellen.
POLL INTERVAL	Start der zyklischen Abfragen an die SIM und Festlegung des Intervalls (<i>polling</i>).
POLLING OFF	Beenden der zyklischen Abfragen (<i>polling</i>) an die SIM.
PROVIDE LOCAL INFORMATION	Anfrage an das Mobile Equipment, dem SIM aktuelle Ortsinformationen zur Verfügung zu stellen.

Tabelle 13.7 (Fortsetzung)

REFRESH	Information an das Mobile Equipment, dass sich Dateninhalte in der SIM geändert haben und das Mobile Equipment diese neu lesen soll.
SET UP EVENT LIST	Übergabe einer Ereignisliste an das Mobile Equipment mit der Aufforderung, die SIM zu informieren, falls ein diesbezügliches Ereignis eintritt.
TIMER MANAGEMENT	Start, Beenden und Konfiguration der acht möglichen Timer im Mobile Equipment, welche ein Ereignis erzeugen können.
LAUNCH BROWSER	Start eines vom Chipkarten-Betriebssystems unterstützten Mikrobrowsers.

Für das SIM Application Toolkit wurde es notwendig, eine Reihe neuer Kommandos festzulegen. Hier muss beachtet werden, dass es sich dabei um Kommandos von der SIM an das Mobile Equipment handelt, was etwas Umdenken erfordert. Der Datenteil der proaktiven Kommandos ist BER-TLV-codiert.¹³ Damit kann eine einfache Erweiterbarkeit bei Sicherstellung von Abwärtskompatibilität erreicht werden. Der größte Vorteil ist jedoch die enorme Flexibilität, die man mit TLV-codierten Daten gewinnt.

Für die SIM musste bei SIM Application Toolkit eine Möglichkeit geschaffen werden, die übliche Master-Slave-Zuordnung zwischen Terminal und Chipkarte zu umgehen, dabei durfte aber das Übertragungsprotokoll aus Kompatibilitätsgründen nicht verändert werden. Die Lösung dieses Problems ist verhältnismäßig einfach. Das Mobile Equipment sendet in festlegbaren regelmäßigen (*polling*) Abständen (beispielsweise alle 20 Sekunden) das Abfragekommando STATUS zur SIM und diese kann gegebenenfalls antworten, dass ein Kommando an das Mobile Equipment bereitsteht und von diesem abgeholt werden soll. Die einstellbare Zeitspanne für das Polling wird in der Praxis vom Mobile Equipment nicht allzu exakt eingehalten, was aber unkritisch ist. Diese Umgehung des Master-Slave-Prinzips bezeichnet man auch als Proaktivität der SIM und die dazugehörigen Kommandos als proaktive Kommandos.

Die Master-Slave-Beziehung zwischen Mobile Equipment und SIM wird auf diese Weise umgedreht. Somit ist es nun möglich, dass die Karte von sich aus die Mobiltelefon-tastatur abfragt, eigene Daten und Menüstrukturen auf dem Display des Mobiltelefons darstellt oder auch einen Piepston auslöst. Auch können über den Mechanismus der Kurznachrichten über die Luftschnittstelle Daten zwischen GSM-Karte mit einem Hintergrundsystem ausgetauscht werden. Beispielsweise lässt sich so ein News-Server regelmäßig abfragen, und das Ergebnis wird als E-Mail oder Kurznachricht auf dem Display des Mobiltelefons dargestellt.

Die diesen Mechanismus möglich machenden Kommandos sind FETCH, TERMINAL RESPONSE und ENVELOPE. Mit dem Kommando FETCH holt das Mobile Equipment ein Kommando von der SIM ab. Nach Abarbeitung dieses Kommandos übergibt das Mobile Equipment die dazugehörige Antwort mit TERMINAL RESPONSE an die SIM. Mit dem Kommando ENVELOPE ist es möglich, Daten an die SAT-Anwendung vom Mobile Equipment aus zu übergeben.

¹³ Siehe auch Abschnitt 4.1 „Strukturierung von Daten“.

Zusätzlich zur Proaktivität hat die SIM noch die Möglichkeit, dem Mobile Equipment bestimmte Ereignisse mitzuteilen, bei denen sie explizit per Kommando informiert werden muss.

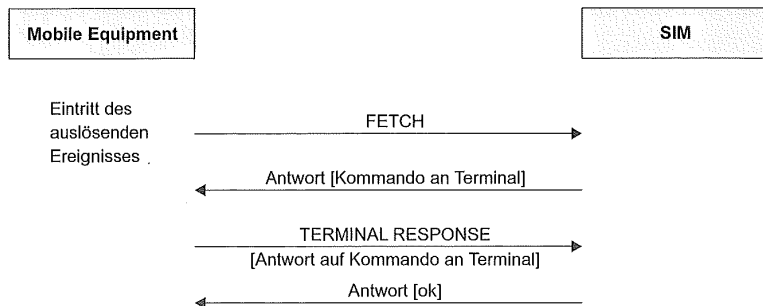


Bild 13.17 Der erweiterte Protokollablauf zwischen Mobile Equipment und SIM bei den proaktiven Kommandos des SIM Application Toolkit nach GSM 11.14. Die Antwort der Chipkarte an ein Kommando enthält im Datenteil ein Kommando an das Terminal. Dieses führt das Kommando aus und gibt die dazugehörige Antwort im Datenteil eines Kommandos an die Chipkarte zurück. Der Ablauf basiert auf den gesendeten APDUs und zeigt ausschließlich den Gutfall.

Es gibt mehrere Möglichkeiten, SAT-basierte Zusatzanwendungen auf der SIM anzustoßen. Die einfachste Art erfolgt über den Benutzer. Wählt dieser beispielsweise aus einem Menü im Mobiltelefon eine bestimmte Funktion und fußt diese auf einer Zusatzanwendung der SIM, so wird vom Mobile Equipment ein entsprechendes Kommando zur SIM gesendet. Den weiteren Ablauf kann dann das Programm der Zusatzanwendung in der SIM bestimmen. Aber auch bestimmte Ereignisse im Mobiltelefon wie Rufaufbau, Rufen- oder Zellwechsel können als Aufruf der SAT-basierten Anwendung in der SIM verwendet werden. Die einfachste Aufrufmethode einer SAT-Anwendung auf der SIM ist die zyklische Abfrage der SIM seitens des Mobile Equipments. Mit diesen drei grundlegenden Aufrufmethoden können in der Praxis alle relevanten SIM-basierten Zusatzanwendungen mit verhältnismäßig überschaubarem Aufwand realisiert werden.

Die eigentliche Funktionalität der Steuerung der Zusatzanwendung im SIM Application Toolkit wird über ausführbaren Programmcode erreicht. Dieser kann in einer beliebiger Programmiersprache, wie Assembler, C oder Java erstellt worden sein.

Der typische Ablauf bei einer SIM Application Toolkit-Anwendung sieht folgendermaßen aus: Nach der Anschaltsequenz der SIM werden unterschiedlichste Daten vom Mobile Equipment gelesen. Darunter auch die Datei EF_{Phase} , in der die von der SIM unterstützte Phase von GSM gespeichert ist. Ist dort die Codierung für Phase 2+ abgelegt, schließt das Terminal daraus, dass SIM Toolkit voll unterstützt wird. Das Terminal übergibt daraufhin mit dem Kommando **TERMINAL PROFILE** seine SIM Toolkit-relevanten Eigenschaften an die SIM. Damit ist die Initialisierung abgeschlossen und es können beliebige andere Kommandos im Rahmen der GSM-Anwendung gesendet werden, die nicht zum SIM Application Toolkit gehören.

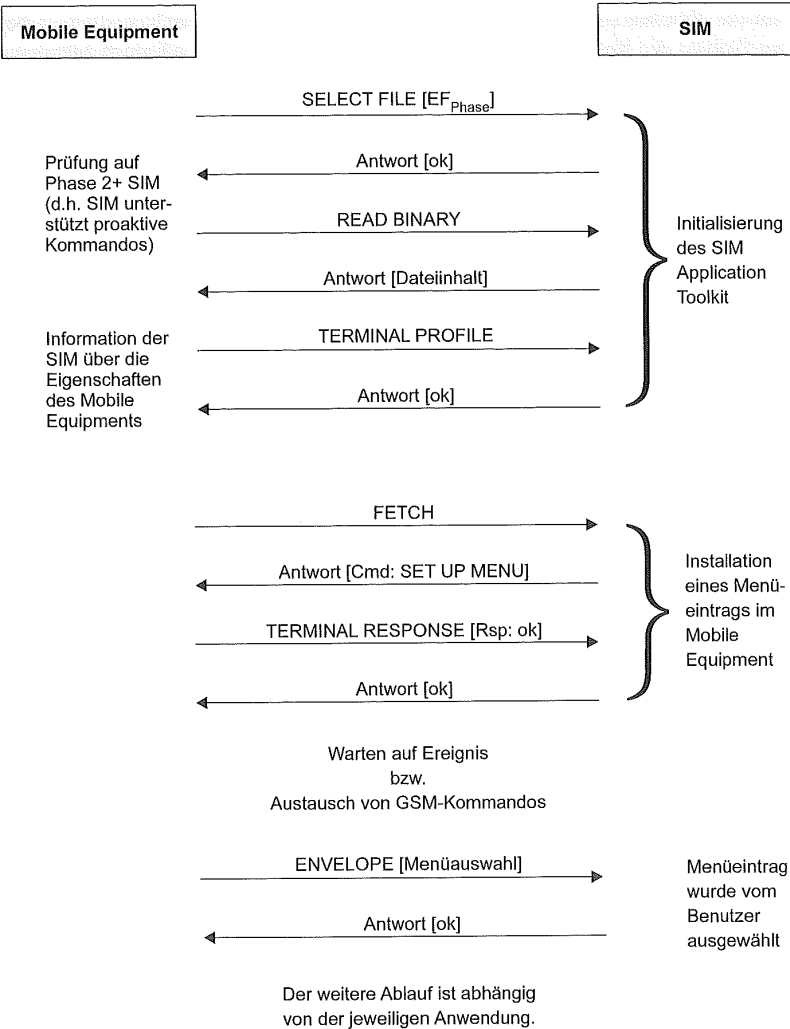


Bild 13.18 Typisches Beispiel der Anwendung des SIM Application Toolkit zur Installation eines zusätzlichen Menüeintrags im Mobile Equipment. Der Ablauf basiert auf den gesendeten APDUs und zeigt ausschließlich den Gutfall.

Der typische nächste Ablauf ist die Installation eines Auswahlmenüs im Mobil Equipment. Dazu werden die BER-TLV-codierte Daten für das Menü als Antwort auf ein von der SIM angefordertes und vom Mobil Equipment gesendetes **FETCH**-Kommando gegeben. Das Mobil Equipment integriert nun das neue Auswahlmenü in seinen Menübaum und quittiert dies mit einer Bestätigung im folgenden **TERMINAL RESPONSE** Kommando. Damit ist das Menü im Mobil Equipment installiert und aktiviert. Nun können wieder

rum die üblichen Kommandos im Rahmen von GSM ausgetauscht und abgearbeitet werden. Sobald der Benutzer des Mobiltelefons nun im neuen Menü einen Menüeintrag auswählt, wird ein ENVELOPE Kommando mit einer Information über den ausgewählten Eintrag zur SIM gesendet. Diese bestätigt den Empfang des Kommandos und kann nun im Rahmen der Anwendung und der Benutzerauswahl unterschiedlichste Abläufe starten. Beispielsweise könnte nun aufgrund der Auswahl von der Anwendung auf der SIM ein Börsenkurs abgefragt werden.

Zur Realisation dieser Funktion könnten nun die unterschiedlichsten Wege begangen werden. Eine einfache Vorgehensweise wäre das Versenden einer SMS an einen Server des Netzbetreibers mit der Frage nach dem aktuellen Börsenkurs einer bestimmten Firma. Der Server würde im Gutfall dann mittels einer Antwort-SMS die Anwendung auf der SIM über den Börsenkurs informieren und diese könnte dann beispielsweise den Benutzer des Mobiltelefons mit DISPLAY TEXT vom aktuellen Kurs in Kenntnis setzen.

Dies ist nur ein sehr einfaches Beispiel für die Möglichkeiten des SIM Application Toolkit, doch zeigt es schon deutlich, dass damit ein sehr mächtiges Werkzeug zur Erstellung von Zusatzanwendungen auf SIMs vorliegt, und zum anderen, dass es auch verhältnismäßig einfach ist. Die Schwierigkeiten beginnen dann, wenn beispielsweise Zusatzanwendungen auf der Grundlage von Funktionen des Mobile Equipments erstellt werden sollen, die vom SIM Application Toolkit nicht unterstützt werden. Ein weiterer (bekannter) Stolperstein bei SAT-basierten Zusatzanwendungen sind unterschiedliche Darstellungsformen auf dem Display oder schlichtweg Inkompatibilitäten oder Implementierungsfehler im Mobile Equipment. Dies alles sind aber Hürden, die mit etwas Anstrengung und einiger Erfahrung überwunden werden können. Zusammenfassend ist deshalb anzumerken, dass der SIM Application Toolkit immer noch die technisch ausgereifteste und sicherste Art der Verwirklichung von Zusatzanwendungen auf einem Mobiltelefon ist.

Mit dem SIM Application Toolkit wurde eine sehr mächtige und ohne Änderungen in das bisherige System integrierbare Schnittstelle für Zusatzanwendungen auf der SIM geschaffen.

Die Expertengruppe EP SCP (*ETSI project smart card platform*) definiert aufbauend auf dem SIM Application Toolkit eine generische Grundlage für alle Application Toolkits von Chipkarten im Mobilfunk. Der Name dieses Toolkits wird CAT (*card application toolkit*) lauten und die Grundlage für SAT (*SIM application toolkit*), USAT (*USIM application toolkit*) und UATK (*UIM application toolkit*) bilden.

Over The Air (OTA)-Kommunikation

Manchmal ist es nach der Ausgabe einer SIM erforderlich, eine direkte Verbindung vom Hintergrundsystem zur SIM aufzunehmen. Vor allem zur Verwaltung von bestehenden Anwendungen und zur Erzeugung neuer Zusatzanwendungen auf der SIM ist diese Kommunikation ein unbedingtes Muss. Deshalb sind in der GSM 03.48 Mechanismen geschaffen worden, um über die Luftschnittstelle (*air interface*) eine sichere Ende-zu-Ende-Kommunikation zwischen Hintergrundsystem und SIM aufzubauen.

Da diese Anforderung durch die ursprünglichen GSM-Spezifikationen nicht abgedeckt war und es fast unmöglich ist, Änderungen in einem derartig großen System durchzuführen, hat man für die Ende-zu-Ende Kommunikation mit der GSM-Karte einen Trick benutzt. Die im System vorgesehenen Kurznachrichten werden als Container für Nachrichten von und zur SIM verwendet. Damit mussten lediglich im Hintergrundsystem und in den

neu ausgegebenen Chipkarten Adaptionen gemacht werden. Alle Systeme dazwischen konnten unverändert beibehalten werden. OTA mittels Kurznachrichten als Trägerdienst ist jedoch mittlerweile nur einer von mehreren möglichen Trägern, wenn auch der mit der größten Verbreitung.

Die OTA-Kommunikation bietet eine verhältnismäßig große Bandbreite an Schutzmaßnahmen für die übertragenen Daten. So können als einfachste Absicherungsstufe die Daten mit einem CRC vor Übertragungsfehler geschützt werden. Im Bereich der kryptografischen Sicherung ist es möglich, die Daten auch mit einem Sendefolgezähler zu versehen und sie mit DES oder Triple-DES (2 Schlüssel oder 3 Schlüssel) zu verschlüsseln. Wenn nötig, kann auch ein MAC oder eine digitale Signatur über die zu übertragenden Daten errechnet werden.

Das Funktionsprinzip mit SMS als Trägerdienst ist dabei Folgendes: Möchte ein Hintergrundsystem beispielsweise ein Kommando (z. B.: READ BINARY) an eine bestimmte SIM senden, so erzeugt es eine Kurznachricht an die jeweilige Karte und bettet das Kommando unter Zuhilfenahme der notwendigen kryptografischen Schutzmaßnahmen darin ein. Sobald sich eine Mobilstation mit der bewussten SIM im System anmeldet, wird die Kurznachricht über den Signalisierungskanal übermittelt. Ein Verbindungsaufbau für Sprache über einen Verkehrskanal ist dazu nicht erforderlich. Das Mobile Equipment erkennt anhand der Codierung der Kurznachricht nach GSM 03.40, dass es sich um SIM-spezifische Daten handelt, und sendet sie mittels des SIM Application Toolkit-Kommandos ENVELOPE an die SIM. Die erhaltene Kurznachricht wird also nicht, wie sonst üblich, automatisch per UPDATE RECORD vom Mobile Equipment in der Datei EF_{SMS} gespeichert. Die SIM legt die via ENVELOPE erhaltene Kurznachricht in einem separaten Puffer ab. Sollte es sich um eine Kurznachricht handeln, die mit weiteren Kurznachrichten verkettet ist, muss die SIM als nächste Aufgabe die korrekte Reihenfolge wiederherstellen, da dies der SMS-Dienst bekanntermaßen nicht gewährleistet.

Anschließend interpretiert die SIM die erhaltene Nachricht, extrahiert daraus das oder die Kommandos und arbeitet sie ab. Als Antwort kann dann optional von der SIM eine Kurznachricht erzeugt werden. Diese wird dann als Antwort auf ein von der SIM angefordertes FETCH-Kommando dem Mobile Equipment übergeben und von dort über den Servicekanal auf dem üblichen Weg an das Hintergrundsystem weitergeleitet.

Mit diesem Kunstgriff ist es nun möglich, transparent zu allen dazwischenliegenden Systemkomponenten eine bidirektionale Ende-zu-Ende-Verbindung zwischen Hintergrundsystem und SIM aufzubauen. Damit kann die SIM genauso angesprochen werden, als ob sie sich in einem an einen PC angeschlossenen Terminal befinden würde. Dieser Kommunikationskanal kann nun beispielsweise im Rahmen eines Remote File Managements für Änderungen von bestehenden Daten in Dateien benutzt werden. Eine gängige Anwendung von OTA ist die Anpassung der Service-Kurzrufnummern in der SIM. Mit OTA lassen sich jedoch auch wesentlich komplexere Themen bearbeiten. So kann damit auch ausführbarer Programcode als Applet für Zusatzanwendungen auf Java Card-basierte SIMs nachgeladen werden. Die Möglichkeiten, die OTA einem Netzbetreiber bietet, sind immens. Leider entspricht GSM 03.48 in weiten Teilen keiner bitgenauen Spezifikation, sondern eher einer Norm, da sie eine große Bandbreite von zum Teil nicht vollständig spezifizierten Optionen anbietet, die vom jeweiligen Kartenhersteller individuell für seine SIMs maßgeschneidert eingesetzt werden können. Darunter leidet naturgemäß die Kompatibilität der SIMs verschiedener Hersteller. Dies muss deshalb im laufenden Netzbetrieb

durch Chipkartenhersteller-spezifische Bibliotheken in den Hintergrundsystemen der Netzbetreiber ausgeglichen werden.

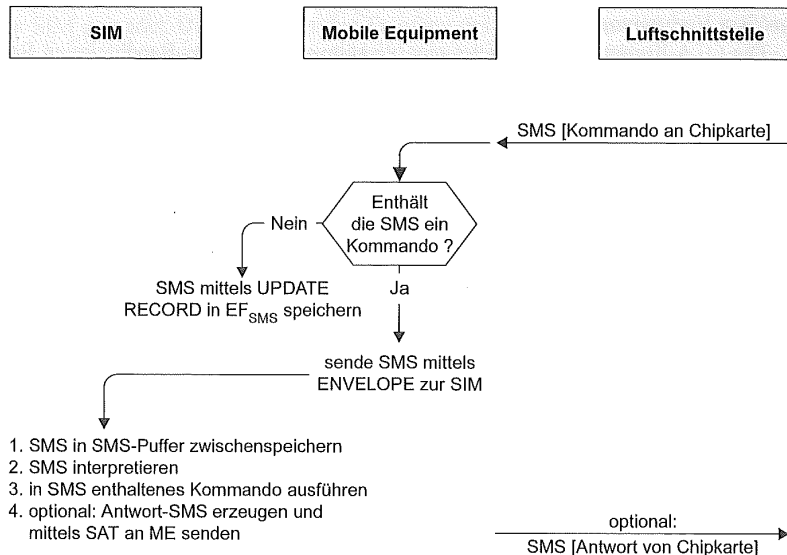


Bild 13.19 Der Ablauf beim Datenaustausch auf der Grundlage von Kurznachrichten zwischen Hintergrundsystem und GSM-Karte. Das Verfahren wird im Allgemeinen als OTA (*over the air*) bezeichnet.

Remote File Management (RFM)

Die von OTA zur Verfügung gestellten Mechanismen gestatten eine direkte Ende-zu-Ende-Kommunikation zwischen Hintergrundsystem und SIM. Dies ist die datenübertragungstechnische Grundlage für eine darauf aufbauende Fernverwaltung der Dateien auf der SIM, im GSM-Sprachgebrauch Remote File Management (RFM) genannt. Spezifiziert ist die trägerunabhängige grundsätzliche Funktionalität in der GSM 03.48, welche wiederum auf den Anforderungen der GSM 02.48 aufbaut.

Im Rahmen des Remote File Managements dürfen nur bestimmte SIM-Kommandos benutzt werden, mit denen sich jedoch durchaus eine erhebliche Bandbreite an Funktionalität erreichen lässt. Diese Kommandos sind eingeteilt in Kommandos, die Daten an die SIM senden (*input commands*) und Kommandos die Daten von der SIM anfordern (*output commands*). Seitens des Hintergrundsystems darf innerhalb einer OTA-Nachricht nicht nur ein einzelnes Kommando an die SIM gesendet werden, sondern auch eine Liste mehrerer Kommandos. Dabei ist jedoch die Einschränkung zu beachten, dass in einer Kommando-liste nur das letzte Kommando Daten von der SIM anfordern darf. Der Grund dieser in der Praxis problemlosen Einschränkung liegt vor allem in der erheblichen Vereinfachung der Software in der SIM für das Remote File Managements. Sollen mehrere Dateien oder Records gelesen werden, so sind wegen der vorgenannten Beschränkung mehrere OTA-Nachrichten an die SIM notwendig.

Tabelle 13.9 Aufstellung der in GSM 03.48 erlaubten Chipkarten-Kommandos an die SIM für das Remote File Management über die Luftschnittstelle.

Kommandos, die Daten an die SIM übergeben (input commands)	Kommandos, die Daten von der SIM anfordern (output commands)
SELECT UPDATE BINARY UPDATE RECORD SEEK INCREASE VERIFY CHV CHANGE CHV DISABLE CHV ENABLE CHV UNBLOCK CHV INVALIDATE REHABILITATE	READ BINARY READ RECORD GET RESPONSE

An einem typischen Beispiel aus der Praxis sei das Funktionsprinzip eines Remote File Managements mit SMS als Trägerdienst kurz verdeutlicht. Möchte ein Hintergrundsystem eine Kurzurufnummer im EF_{ADN} ändern, so könnte es dabei folgendermaßen vorgehen: In einer ersten OTA-Nachricht – beispielsweise mittels mehrerer aufeinander folgender Kurznachrichten übertragen – wird das EF_{ADN} mittels eines SELECT-Kommandos unter Angabe des Pfades im DF_{Telecom} ausgewählt. Als abschließendes Kommando in dieser OTA-Nachricht wird mit der dem Hintergrundsystem bekannten Recordnummer und READ RECORD die Service-Nummer ausgelesen und via OTA zurückgesendet. Ist die Service-Nummer nicht auf dem aktuellen Stand, wird mit einer weiteren OTA-Nachricht ein UPDATE RECORD an die SIM gesendet und der entsprechende Record mit der neuen Nummer überschrieben.

Vorsicht ist natürlich angebracht, wenn Dateien via Remote File Management geändert werden sollen, die für die laufende Sitzung von Bedeutung sind. Ein typisches Beispiel ist hier das EF_{SST} mit der SIM Service Table, einer Tabelle, die die möglichen und ggf. aktivierten Funktionen der SIM aufführt. Eine Änderung des Dateiinhalts kann dabei unter Umständen zu einem unvorhersehbaren Verhalten des Mobiltelefons führen oder im schlimmsten Fall die SIM unbrauchbar machen.

Die SIM beherbergt auch zwei EFs, bei denen Änderungen schlichtweg untersagt sind. Dies ist die Datei EF_{ICCID} mit der Identifikationsnummer der Chipkarte und die Datei EF_{Kc} mit dem Schlüssel zur Verschlüsselung der Daten auf der Luftschnittstelle zwischen Mobilstation und Feststation. Logisch gesehen ergibt es keinen Sinn, eine der beiden Dateien zu ändern, da die ICCID eine eindeutige Identifikationsnummer der Chipkarte ist und für den operativen Betrieb keine Rolle spielt. Der Schlüssel Kc wird ohnehin von der SIM pro Sitzung berechnet; es wäre unsinnig, ihn über RFM zu ändern. Falls man es bei einer laufenden Verbindung trotzdem tut, könnte die Verbindung zum Netzwerk möglicherweise unterbrochen werden, da vom Mobile Equipment der falsche Schlüssel für die Verschlüsselung der Daten auf der Luftschnittstelle verwendet würde.

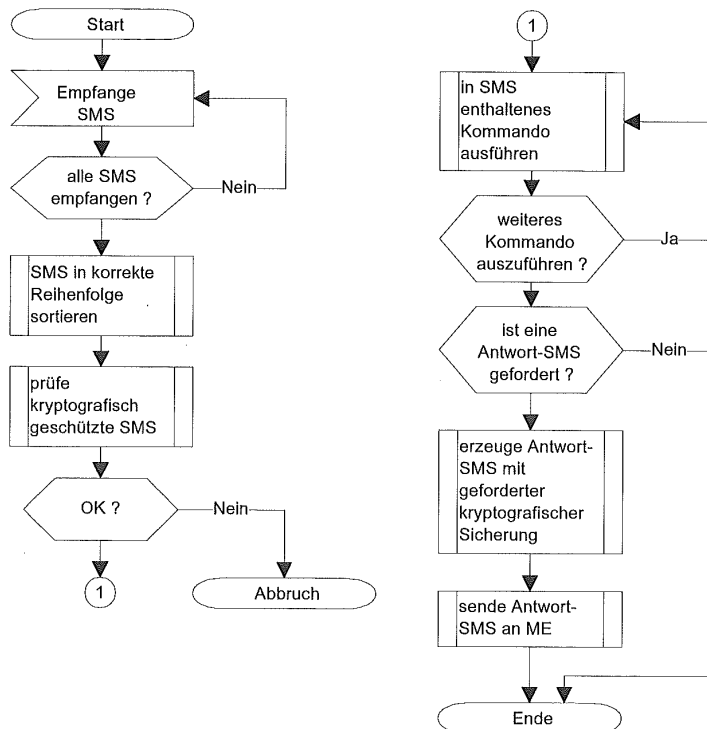


Bild 13.20 Flussdiagramm des grundlegenden Programmblaufs beim Remote File Management (RFM) über die Luftschnittstelle via OTA nach GSM 03.48. Das Remote File Management wird im Wesentlichen im rechten oberen Ast des Flussdiagramms durchgeführt, die restlichen Abläufe dienen zur Errichtung der Übertragungssicherheit nach GSM 03.48.

Remote Applet Management

Analog dem Remote File Management existiert in der GSM 03.48 auch ein Abschnitt über Remote Applet Management. Somit ist es möglich, über eine direkte Ende-zu-Ende-Kommunikation zwischen dem Hintergrundsystem und der SIM auf der Java Card aufbauende Anwendungen zu verwalten.

Als generelle Rahmenbedingung muss es sich bei der verwendeten Chipkarte um eine SIM nach GSM 03.19 handeln, welche im Wesentlichen auf den Java Card 2.1-Spezifikationen¹⁴ beruhen. Alle Verwaltungskommandos für Applets und Packages basieren auf der Open-Platform-Spezifikation, dem faktischen Industriestandard für diese Mechanismen.¹⁵

Zur Verwaltung von Anwendungen gehört das Laden (*loading*), Einrichten (*installation*), Entfernen (*removal*), Sperren (*locking*), Entsperren (*unlocking*) und die Informationsabfrage (*parameter retrieval*) von Java Applets auf der SIM. Desgleichen sind Mechanismen festgelegt, um Packages auf die SIM zu laden oder zu entfernen.

¹⁴ Siehe auch Abschnitt 5.14.1 „Java Card“.

¹⁵ Siehe auch Abschnitt 5.11 „Open Platform“.

Alle Abläufe und Mechanismen sind grundsätzlich unabhängig von einem bestimmten Trägerdienst, doch ist mittelfristig der Kurznachrichtendienst der universellste Weg, Applets und Packages per OTA auf einer SIM zu verwalten. Im Falle von SMS als Träger würde die Datenübertragung von und zur SIM völlig analog den beschriebenen Sequenzen bei OTS und Remote File Management ablaufen.

Tabelle 13.10 Aufstellung der in GSM 03.48 erlaubten Chipkarten-Kommandos an die SIM für das Remote Applet Management über die Luftschnittstelle. Die Kommandos entsprechen in Funktionalität und Codierung der Open Platform Spezifikation.

Kommandos, die Daten an die SIM übergeben (input commands)	Kommandos, die Daten von der SIM anfordern (output commands)
DELETE SET STATUS INSTALL LOAD PUT KEY	GET STATUS GET DATA

Dual-IMSI

Im geschäftlichen Bereich werden normalerweise die anfallenden Gesprächsgebühren der Firmenmobiletelefone von der jeweiligen Firma übernommen. Möchte nun der Benutzer eines solchen Mobiltelefons ein privates Gespräch führen, so müsste er dies später manuell mit seiner Firma abrechnen, was aber nur funktionieren würde, wenn er eine Telefonrechnung mit Einzelkostennachweis hat. In der Praxis wird er wohl sein Privatgespräch auf Firmenkosten führen, oder er müsste in Konsequenz zwei Geräte mit sich herumtragen, um auch private Gespräche führen zu können. Dass dies eigentlich niemandem zumutbar ist, muss hier nicht weiter ausgeführt werden. Dieses Szenario ist der Hauptgrund, weshalb es so genannte Dual-IMSIs gibt. Dabei ist auf der SIM eine zusätzliche Datei vorhanden, in der sich zwei IMSIs und je nach Realisierung auch zwei Ki befinden. Diese Datei ist im Übrigen nicht Bestandteil einer Norm. Eine Zusatzanwendung in der SIM auf der Grundlage des SIM Application Toolkits erzeugt nun einen neuen Mentieintrag im Mobile Equipment, mit dem der Benutzer auswählen kann, ob er geschäftlich oder privat telefonieren will. Je nach Wahl des Benutzers wird nun eine bestimmte IMSI in das EF_{IMSI} kopiert und optional auch ein anderer dazugehöriger Ki verwendet. Damit besitzt das Mobiltelefon zwei Identitäten im Sinne der IMSI und der Netzbetreiber kann zwei unterschiedliche Gesprächsabrechnungen erstellen. Damit sind geschäftliche und private Gespräche getrennt verrechenbar geworden.

Die Funktionalität kann noch weiter ausgebaut werden, indem beispielsweise bei der IMSI für geschäftliche Telefonate die Festrufnummern aktiviert werden, sodass der mögliche Rufnummernkreis auf bestimmte Nummern eingeschränkt ist. Damit hat der Benutzer eines solchen Mobiltelefons keine Möglichkeit mehr, private Gespräche zu führen, da nur firmenrelevante Rufnummern gewählt werden können. Wird hingegen die zweite IMSI über die Zusatzanwendung in der SIM aktiviert, so werden die Festrufnummern deaktiviert und die Gesprächsgebühren nicht mehr von der Firma getragen. Damit kann man mit einem einzigen Mobiltelefon sowohl geschäftlich als auch privat telefonieren, ohne dass es zu Problemen mit der Gebührenabrechnung kommt.

Es gibt noch eine weitere Anwendung von mehrfachen IMSIs, die technisch gesehen allerdings nicht sehr elegant ist. Sind auf einer SIM mehrere IMSIs verschiedener Netz-

betreiber gespeichert und können diese vom Benutzer über ein Menü ausgewählt werden, so lässt sich dies zum manuellen Roaming verwenden. Der Benutzer wählt je nach Netzwerk, in dem er sich gerade befindet, die jeweils dazugehörige IMSI aus und kann sich dann mit dieser in das vorhandene Netzwerk einbuchen. Vom jeweiligen Netzbetreiber erhält er dann eine entsprechende Telefonrechnung. Diese Art von Roaming mit mehrfachen IMSIs wird beispielsweise in weiten Teilen Indiens praktiziert, da es dort zwischen manchen Netzbetreibern keine regulären Roaming-Abkommen gibt.

Realisierung einer Homezone

Einige Netzbetreiber bieten personenbezogen und für lokal begrenzte Gebiete Sondertarife an. In der Regel ist dies die um den eigenen Wohnsitz festlegbare annähernd kreisförmige Zone, in der bei Benutzung des Mobiltelefons statt des teuren Mobiltarifs der günstigere Festnetztarif abgerechnet wird. Der Vorteil für den Benutzer dieses standortbezogenen Dienstes liegt vor allem darin, dass er keinen Festnetzanschluss mehr benötigt und seine Kurzrufnummern nicht zwischen Mobiltelefon und Festnetztelefon abgleichen muss.

Die Realisierung dieses Dienstes lässt sich geschickterweise mit einer Anwendung im SIM und den Funktionen des SIM Application Toolkit realisieren. Damit ist auch gewährleistet, dass alle Daten über die Homezone in der personenbezogenen SIM gespeichert sind und nicht etwa im Mobiltelefon.

Im GSM-System sendet jede Feststation über die Luftschnittstelle laufend eine eindeutige Kennung aus, die sich aus der LAI (*location area information*) und einem CI (*cell identity*) zusammensetzt. Ein möglicher Lösungsansatz wäre nun, diese Information vom Mobiltelefon an die SIM weiterzureichen und dort mit einem oder mehreren gespeicherten Werten zu vergleichen. Diese Werte würden vorzugsweise in einer Datei abgelegt, sodass sie zu beliebigen Zeitpunkten über RFM (*remote file management*) geändert oder angepasst werden können. Nachteilig bei dieser Lösung ist der verhältnismäßig große Speicherbedarf in der SIM, da grundsätzlich auch Gegenden mit sehr hoher Dichte an Feststationen berücksichtigt werden müssen, was dann zu einer sehr umfangreichen Liste von LAIs und CIs führen würde.

Ein ähnlicher Lösungsansatz wäre es, die *timing advance*-Information der Luftschnittstelle zu verwenden. Damit könnte man in Bereichen hoher Zelldichte mittels Kreuzpeilung den aktuellen Standort einer Mobilstation auf deutlich unter 10 m genau feststellen, was für die Realisierung einer Homezone mehr als ausreichend ist.

In der Praxis wird jedoch oft einer anderen Lösung der Vorzug gegeben.¹⁶ Dabei senden alle Feststationen eines Netzbetreibers über den Signalisierungskanal via Cell Broadcast Service periodisch ihre Ortskoordinaten aus. Auf der SIM befindet sich ein EF mit Referenzwerten, welche vom Mobile Equipment gelesen und dann mit den empfangenen Ortskoordinaten verglichen werden. Stellt das Mobile Equipment fest, dass sich das Mobiltelefon innerhalb der Homezone befindet, wird ein entsprechendes Symbol (z. B. ein kleines Häuschen) auf dem Display angezeigt. Da das Hintergrundsystem den Ort des Mobiltelefons kennt, kann es ein- und ausgehende Gespräche auf einen günstigeren Tarif umschalten. Die Informationen über die Koordinaten der Homezone sind in einem EF der SIM abgelegt, deshalb lassen sich diese auch problemlos über RFM (*remote file management*) ändern. Somit lässt sich die Homezone auch bequem per Fernwartung einrichten

¹⁶ c't 1999, Heft 18, „Telefon-Zellen“, Harald Bögeholz, Dusan Zivadinovic.

oder an einen anderen Ort verlegen. Nachteilig an dieser Lösung ist die Tatsache, dass sie spezielle Software im Mobile Equipment erfordert und nicht durch eine Zusatzanwendung in der SIM mit SIM Application Toolkit Funktionen verwirklicht werden kann.

Funktionsweise des SIM-Lock

SIM-Lock oder SIM-Sperre ist die Bezeichnung für das Verfahren, ein Mobile Equipment fest mit einer bestimmten SIM oder einer bestimmten Gruppe von SIMs zu verbinden. Die SIM-Lock-Funktion wird benutzt um vom Netzbetreiber subventionierte Mobiltelefone für eine bestimmte Zeit an eine bestimmte SIM und deren Bezahlmodalität zu binden. Die normative Grundlage zum SIM-Lock bildet die Spezifikation GSM 02.22. Das Funktionsprinzip eines SIM-Lock beruht immer darauf, dass Daten, die sowohl im Mobile Equipment als auch in der SIM gespeichert sind, von einer der beiden Komponenten beim Anschalten verglichen werden und bei positivem Ausgang des Vergleichs die Telefonfunktionalität freigeschaltet wird.

Zur technischen Realisierung des SIM-Locks gibt es zwei praxisrelevante Varianten. Bei der häufig benutzten Variante eins liest das Mobiltelefon aus der SIM bestimmte Daten und vergleicht diese mit im Mobiltelefon gespeicherten Daten. In der Regel sind dies die so genannten Group Identifiers, die in den Dateien EF_{GID1} (Group Identifier Level 1) bzw. EF_{GID2} (Group Identifier Level 2) gespeichert sind. Mit diesen Group Identifiers lassen sich Klassen von SIMs festlegen, die dann für eine klassenbasierte Paarung von bestimmten SIMs zu bestimmten (subventionierten) Mobile Equipments benutzt werden. Der Vorteil dieser Variante ist, dass SIM und Mobile Equipments nicht individuell verheiratet werden müssen. Alternativ zu den Group Identifiers wird manchmal auch noch die IMSI aus dem EF_{IMSI} oder andere statische SIM-spezifische Daten, die in EFs gespeichert sind, als Referenzwert für die Paarbildung benutzt.

Variante zwei, welche jedoch in der Praxis selten verwendet wird, funktioniert, indem die SIM mittels SIM Application Toolkit eindeutige Daten aus dem Mobile Equipment liest und diese mit gespeicherten Daten vergleicht. Passen die Daten zueinander, kann das Mobiltelefon benutzt werden, indem es über die SIM für das anstehende Gespräch freigeschaltet wird.

Es ist in der Regel möglich, entweder über die Luftschnittstelle oder durch Eingabe eines geheimen Schlüssels am Mobiltelefon, die SIM-Lock Funktion abzuschalten, sodass anschließend auch andere SIMs in dem SIM-Lock geschützten Mobile Equipment verwendet werden können. Die Referenz auf das jeweilige Mobile Equipment ist dann in aller Regel die eindeutige Geräteummer IMEI.

Funktionsweise von Prepaid-Lösungen

Der Anteil von Prepaid-SIMs bewegt sich je nach Land in Größenordnungen von 30 % bis hin zu 80 %. Die dafür hauptsächlich ausschlaggebenden Gründe sind einerseits die bessere Kostenkontrolle und andererseits die entfallenden Grundgebühren.

Das Funktionsprinzip der Realisierung eines Systems für vorbezahlte SIM gestaltet sich in der Regel folgendermaßen: Auf einem kartenförmigen Voucher, oft mit ID-1 Abmessungen, doch geringerer Dicke, befindet sich unter einem als Siegel dienenden Rubbelfeld eine beispielsweise 13-stellige Zahl. Will der Benutzer nun sein Mobiltelefon „aufladen“, kauft er sich einen Voucher und kann dessen Integrität durch die Unversehrtheit des Rubbelfeldes feststellen. Nach dem Abrubbeln des Zahlenfeldes muss der Benutzer nun die sichtbare Zahl in einem besonderem Menü seines Mobiltelefons eingeben. Dieser Referenzwert wird

direkt an das Hintergrundsystem weitergereicht, wo er über eine Datenbank mit den Referenzwerten der ausgegebenen Voucher verglichen wird. Ist der Vergleich positiv und wurde er zum ersten Mal durchgeführt, so wird der mit dem Referenzwert verbundene Ladebetrag der jeweiligen SIM gutgeschrieben. Hier teilen sich nun die möglichen Realisierungswege.

Die in den GSM-Spezifikationen ursprünglich vorgesehene Lösung war ein Einheitenzähler in einer Datei (EF_{ACM} – *Accumulated call meter*) deren vom Mobile Equipment mit *advice of charge* laufend aktualisiertes Wert von der SIM mit einem Maximalwert in der Datei (EF_{ACMmax} – *Accumulated call meter maximum value*) verglichen wird. Erreicht der aktuelle Wert den Maximalwert, unterbindet das Mobiltelefon weitere Telefonate so lange, bis der aktuelle Wert wieder zurückgesetzt worden ist. Dies könnte beispielsweise über Remote File Management geschehen. In der Praxis wird dieses technisch durchaus praktikable System jedoch nicht benutzt, da die Kommunikation zwischen Mobile Equipment und SIM nicht sicher ist und deshalb verhältnismäßig einfach manipuliert werden könnte.

In der Praxis wird die Verwaltung von vorbezahlten SIMs über ein zentrales System abgewickelt, wobei es zwei Lösungsansätze gibt: Ansatz eins verzichtet vollständig auf zusätzliche Informationen auf der SIM und wird zur Gänze im Hintergrundsystem abgewickelt. Der Nachteil dabei ist, dass die Rechner im Hintergrundsystem echtzeitfähig sein müssen, was die Anschaffungskosten in die Höhe treibt. Bei Lösungsansatz zwei muss sich auf der SIM eine entsprechende Zusatzanwendung befinden, dafür ist bei den Hintergrundsystemen keine generelle Echtzeitfähigkeit mehr notwendig. Nachteilig dabei ist, dass im Falle eines abtelefontierten Guthabens der Verbindungsabbruch unter Umständen zeitlich etwas verzögert erfolgt.

Am Beispiel der in den Bildern 13.21 und 13.22 dargestellten Komponenten und Abläufe wird im Folgenden ein typisches Hintergrundsystem ohne zwingend notwendige Echtzeitfähigkeit für Prepaid-SIMs erläutert. Für die Unterstützung vorbezahlter SIMs werden in ein bestehendes GSM-System einige zusätzliche Instanzen integriert. Dies ist im Beispiel das Anrufmanagement als Zusatzelement der Mobilvermittlungsstelle MSC, das in Echtzeit Rufe weiterleiten oder unterbinden kann und eine Schnittstelle zum eigentlichen Prepaid-System unterhält. Dieses ist die zentrale Instanz mit der Aufgabe, das Anrufmanagement und das eigentliche Abrechnungssystem zu koordinieren. Im Abrechnungssystem wird mittels einer Datenbank die Verwaltung der SIM-spezifischen Konten mit den Gesprächsguthaben durchgeführt. Die SIM enthält bei diesem System nur einige spezielle Kommandos sowie entsprechende Daten.

Wenn beim Mobiltelefon ein Anruf eintrifft, wird nun als Erstes vom Anrufmanagement das Prepaid-System darüber informiert, dass an ein bestimmtes Mobiltelefon mit einer bestimmten SIM ein Ruf erfolgen soll. Das Prepaid-System lässt nun als nächsten Schritt vom Abrechnungssystem die maximale Gesprächsdauer auf Basis des noch vorhandenen Gesprächsguthabens für die bewusste SIM errechnen und übergibt die Information dem Anrufmanagement. Dieses leitet bei genügend großem Guthaben den Ruf weiter und würde ihn seinerseits unterbrechen, wenn er die maximale Gesprächsdauer erreicht. Nach Beendigung des Gesprächs teilt das Anrufmanagement dem Prepaid-System die Gesprächsdauer mit, was dieses nutzt, um über das Abrechnungssystem den Saldo des Guthabens zu aktualisieren. Der aktualisierte Saldo kann dann auf dem Display des Mobiltelefons auch entsprechend dargestellt werden.

Bei einem vom Mobiltelefon abgehenden Ruf findet im Prinzip ein analoger Ablauf statt. Zuerst wird über USSD-Anfrage über das Prepaid- und Abrechnungssystem die maximal mögliche Gesprächsdauer errechnet und diese dem Anrufmanagement übergeben. Sol-

Ite während des Gesprächs der Maximalwert erreicht werden, wird das Telefonat unterbrochen. Ist dies nicht der Fall, wird nach Beendigung des Telefonats das Gesprächssaldo auf den aktuellen Stand gebracht und in der Datenbank entsprechend eingetragen. Optional kann natürlich auch der noch vorhandene Betrag auf dem Mobiltelefon dargestellt werden.

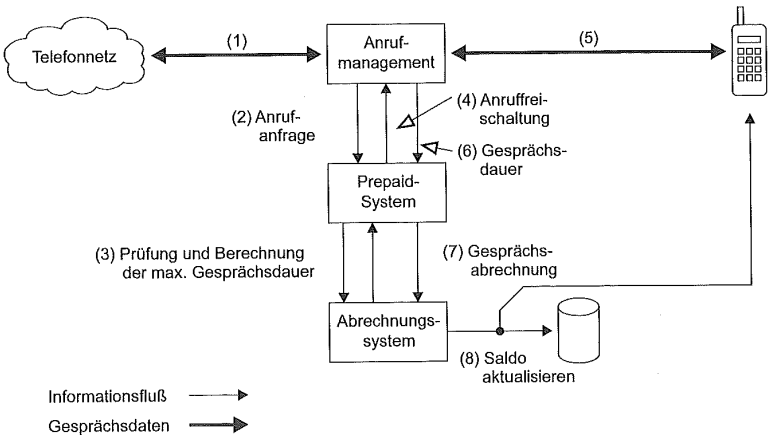


Bild 13.21 Grundlegender Systemaufbau eines Systems für vorbezahlte SIMs am Beispiel der SICAP Prepaid-Roaming-Lösung. Die Grafik zeigt den Ablauf bei einem zum Mobiltelefon geleiteten Anruf, die in Klammern gesetzten Zahlen die Reihenfolge des Ablaufs. Das Anrufmanagement ist Teil des GSM-Hintergrundsystems, beispielsweise eine Komponente der Mobilvermittlungsstelle MSC.

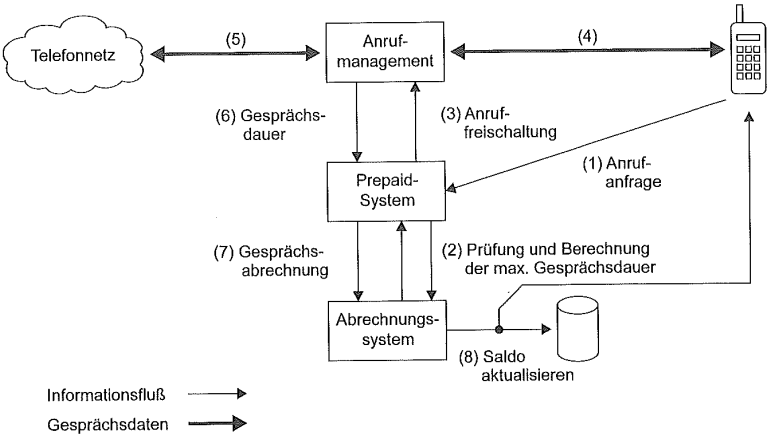


Bild 13.22 Grundlegender Systemaufbau eines Systems für vorbezahlte SIMs am Beispiel der SICAP Prepaid-Roaming-Lösung. Die Grafik zeigt den Ablauf bei einem vom Mobiltelefon abgehenden Anruf, die in Klammern gesetzten Zahlen die Reihenfolge des Ablaufs. Das Anrufmanagement ist Teil des GSM-Hintergrundsystems, beispielsweise eine Komponente der Mobilvermittlungsstelle MSC.

13.2.5 GPRS (*General Paket Radio Service*)

GPRS (*General Paket Radio Service*) ist eine von ETSI standardisierte Erweiterung des ursprünglichen GSM-Systems um einen paketorientierten Datendienst mit höherer Datenübertragungsrate, spezifiziert in GSM 01.60 (Requirements specification of GPRS) und GSM 02.60 (Service description; Stage 1). Diese kann dem aktuell notwendigen Kapazitätsbedarf dynamisch angepasst werden, sodass immer nur die aktuell benötigte Kapazität belegt wird. Durch Bündelung der acht vorhandenen Zeitschlitzte zu je 14400 Bit/s ist eine maximale Übertragungsrate von 115,2 kBit/s möglich. Ein Mobiltelefon mit GPRS-Technologie ist hinsichtlich der Datenübertragung ständig im Netz eingebucht und somit zu jedem Zeitpunkt für Datenübertragungen verfügbar, ohne dass dazu eigens eine Verbindung aufgebaut werden muss. Aus diesem Grund eignet sich GPRS auch sehr gut für diskontinuierliche Datenübertragungen. GPRS ist auch die Grundlage für IP-basierte (*Internet protocol*) Dienste im Mobilfunk.

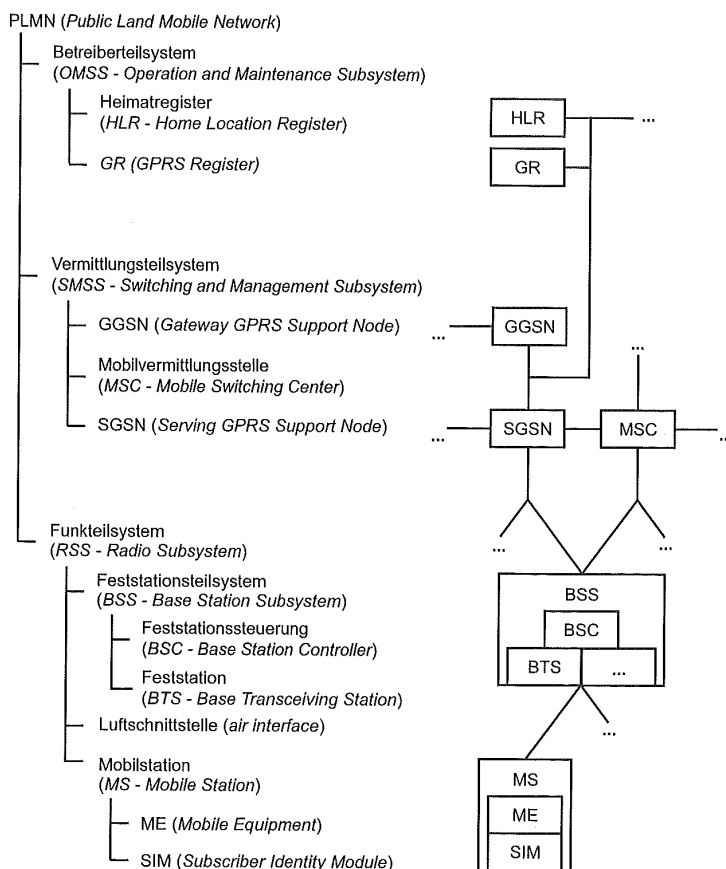


Bild 13.23 Die Teilarchitektur des GSM-Netzes eines einzelnen Netzbetreibers mit einem darin überlagerten GPRS-Netz, nach den GSM Spezifikationen GSM 01.60 und GSM 02.60.

Hinsichtlich der Systemarchitektur nutzt GPRS als Fundament ein GSM-System, das um einige neue Komponenten ergänzt wird. Als Analogon zur Mobilvermittlungsstelle MSC fungiert der Serving GPRS Support Node SGSN, welcher auf der Ebene des MSC den Austausch der Datenpakete in Richtung Mobile Equipment koordiniert. Dem SGSN übergeordnet ist ein Gateway Support Node GGSN, dessen Hauptaufgabe die Schnittstelle zu anderen packetorientierten Datendiensten wie X.25 oder IP ist. Der GGSN transformiert in beide Richtungen die GPRS-spezifischen Datenpakete in die entsprechenden anderen packetorientierten Dienste. Als zentrale Instanz, analog dem HLR, werden im GPRS-Register GR alle Informationen über einen bestimmten GPRS-Teilnehmer verwaltet.

13.2.6 Zukünftige Entwicklung

Die Anwendung GSM hat für Chipkarten den internationalen Durchbruch bedeutet und ist immer noch „der“ Standard für Chipkarten und Chipkarten-Betriebssysteme. Gegenüber den neuesten Entwicklungen auf dem Chipkarten-Sektor mögen manche Kommandos und Mechanismen des GSM-Bereichs veraltet erscheinen, doch war und ist dies der Wegbereiter für große und auch internationale Chipkarten-Anwendungen. Letztendlich können alle folgenden Anwendungen von den Erfahrungen und den Problemen dieser Anwendung nur lernen und profitieren. In vielerlei Hinsicht stellt GSM mit den Spezifikationen 11.11 und 11.14 die Grundlage aller neuen und aufwendigeren Chipkarten-Anwendungen dar.

Die Entwicklung der mobilen Endgeräte schließt neben der reinen Telefonfunktion immer mehr die Funktionen eines PDAs (*personal digital assistant*) ein. Da es zudem verhältnismäßig schwierig ist, die Software in einem Mobiltelefon von außen zu manipulieren, bietet es sich auch als vertrauenswürdiges Gerät (*trusted device*) an. Die hierdurch entstehenden Auswirkungen kann man bei vielen Servicefunktionen und hardwaremäßig erweiterten Mobiltelefonen beobachten. So besitzen Mobiltelefone Infrarotschnittstellen nach der IrDA-Spezifikation und auch entsprechende Bluetooth-Schnittstellen sowie immer größere und leistungsfähigere Displays.

Damit besteht seitens der Technik die Möglichkeit, an einer entsprechend erweiterten Kasse mit Mobiltelefon und elektronischer Geldbörse zu bezahlen. Falls eine PIN-Eingabe notwendig sein sollte, kann diese in Zukunft auf der relativ manipulationssicheren Mobiltelefonastatur erfolgen und nicht auf einem unbekannten Terminal. Über die Infrarot- oder Bluetooth-Schnittstelle lassen sich die entsprechenden Daten austauschen, und es ist nicht einmal nötig, eine kostenpflichtige Telefonverbindung aufzubauen. Die Möglichkeiten, die sich hierbei eröffnen, sind extrem vielfältig und lassen sich zum gegenwärtigen Zeitpunkt nur andeutungsweise absehen.

Dual Slot Mobiltelefone haben sich aus vielerlei Gründen nicht durchgesetzt. Ausschlaggebend waren wohl weniger technische Gründe wie Baugröße des Endgerätes als geschäftspolitische der Netzbetreiber. Diese zeigten bislang wenig Interesse daran, Anwendungen auf Chipkarten von dritten in ihren hoch subventionierten Endgeräten zum Zuge kommen zu lassen. Die Entwicklungsrichtung konzentriert sich derzeit auf Zusatzanwendungen auf der SIM. Die breite Einführung der digitalen Signaturanwendung im Rahmen der WIM, die im Gegensatz zu ihrem Namen nicht nur für WAP verwendet werden kann, schafft zumindest die technischen Voraussetzungen für die gesamte Palette an mobilen Geschäftsabwicklungen.

Die auf der SIM implementierten Mikrobrowser werden mit Sicherheit auch für die nächsten Jahre die Grundlage für sichere datenbasierte Anwendungen bilden, wobei es hier unvermeidlich zu einem Verdrängungswettbewerb mit GSM-fähigen Java Cards kommen dürfte. Diese werden zumindest im ersten Ansatz eher im Bereich von programcodebasierten Zusatzanwendungen ihre Hauptanwendung finden.

MExE (*mobile station execution environment*) ist ein Framework zur Integration netzbetreiberdefinierter Abläufe und ausführbaren Programmcodes in die Mobilstation. Die Stufe 1 von MExE legt die Integration von WAP-Browser für die Auszeichnungssprache WML in Mobile Equipments fest. Die darauf aufbauende Stufe 2 erweitert diese Funktionen um eine Java Virtual Machine (JVM). Damit können Java-Programme in Mobiltelefonen geladen und ausgeführt werden und es besteht die Möglichkeit, Zusatzanwendungen direkt im Mobiltelefon und nicht (wie zurzeit üblich) auf der SIM zu realisieren.

CAMEL (*customized applications for mobile enhanced logic*) ist ein neues mögliches Charakteristikum des GSM-Netzes zur Erweiterung der Funktionalität in Richtung so genannter intelligenter Netze (IN – *Intelligent Network*). Mit CAMEL können beispielsweise Rufnummern während des Rufaufbaus vom Netzwerk geändert werden. Damit lassen sich dann – deutlich einfacher als heute – Anwendungen wie weltweites Roaming mit vorbezahlten Karten oder weltweit verfügbare einheitliche Servicenummern realisieren.

Auch ein etabliertes Mobilfunksystem wie GSM muss weiterentwickelt werden, um neue Anforderungen und zusätzliche Kundenwünsche erfüllen zu können. Dies erfolgt momentan in kleinen Schritten und hat zu den Änderungen und Ergänzungen hinsichtlich proaktiver SIM-Kommandos, OTA, WIM, Mikrobrowser, RFM sowie zu erweitertem Leistungsumfang wie HSCSD, GPRS und EDGE geführt. Irgendwann wird es jedoch notwendig sein, einen großen Entwicklungsschritt zurückzulegen, um alle Ergänzungen, Veränderungen und Sonderfälle wieder in ein in sich geschlossenes neues System zu überführen. Das System wird UMTS (*Universal Mobile Telecommunication System*) sein, voraussichtlich viele Jahre parallel zu GSM existieren und möglicherweise irgendwann das GSM-System ablösen.

13.3 Das UMTS-System

Im Jahr 1998 wurde von den fünf Normungsinstituten ANSI T1 (USA), ARIB (Japan), ETSI (Europa), TTA (Korea) und TTC (Japan) das Projekt 3GPP (*third generation partnership project*) ins Leben gerufen. Die Aufgabe des Projekts war es, von den GSM-Spezifikationen als Grundlage ausgehend, die Folgegeneration für ein weltweites IMT-2000 konformes Mobilfunksystem zu spezifizieren. Dieses Mobilfunksystem wird weltweit üblicherweise auch als ein 3G-System (dritte Generation) bezeichnet, ist jedoch in Europa vor allem unter dem Namen UMTS (*Universal Mobile Telecommunication System*) ein Begriff geworden.¹⁷ Die breite Öffentlichkeit hat dieses System anfangs weniger aufgrund der neuen möglichen Anwendungen beachtet, als wegen der riesigen Lizenzgebühren für das notwendige Frequenzspektrum. Alleine in Deutschland wurden dafür von den

¹⁷ In diesem Buch wird durchgehend der Begriff UMTS benutzt, da er im Gegensatz zur Bezeichnung 3G dieses Mobilfunksystem eindeutig beschreibt. So ist beispielsweise auch das CDMA-2000 Mobilfunksystem ein 3G-System, allerdings mit optionaler Chipkarte, dem R-UIM (*removable user identity module*), welche viele Unterschiede zum USIM von UMTS aufweist.

Netzbetreibern bei der Versteigerung ungefähr 50 Milliarden Euro ausgegeben, in Europa waren es in der Summe 112 Milliarden Euro. Der Aufbau der Netze alleine in Deutschland wird nochmals 30 Milliarden Euro verschlingen.

Nach verhältnismäßig kurzer Zeit gingen bereits Ende 2001 die ersten UMTS-Netzwerke in Japan in Betrieb. UMTS wurde vor u. a. allem durch einige Länder wie beispielsweise Japan vorangetrieben, in denen aufgrund der hohen Teilnehmerdichte ein weiterer Ausbau der vorhandenen Mobilfunksysteme nicht mehr sinnvoll war.

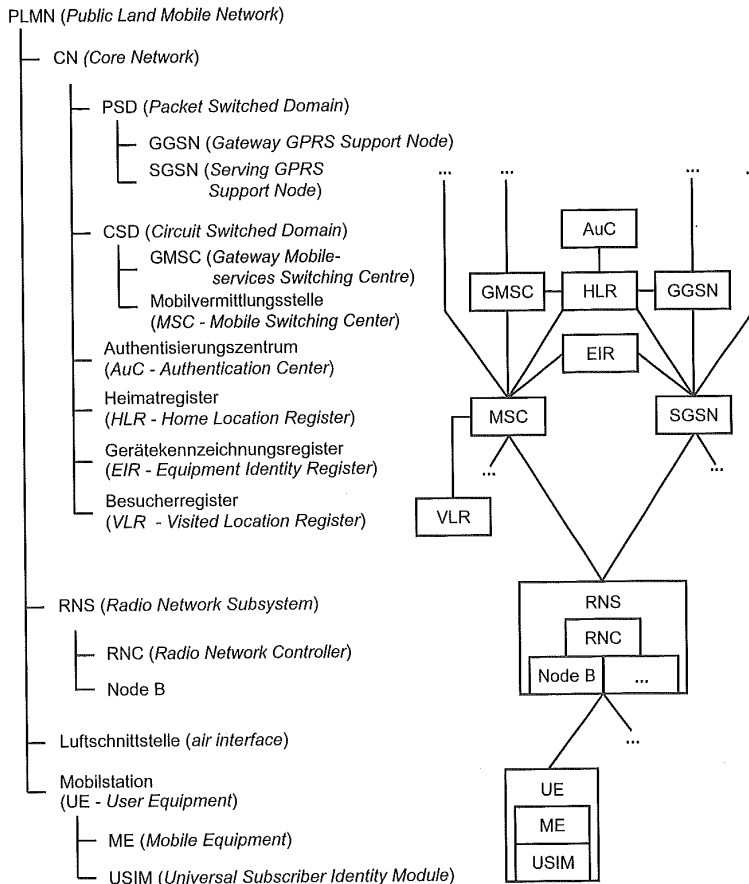


Bild 13.24 Die grundlegende Architektur und wichtigsten Bezeichnungen für Komponenten eines typischen Mobilfunknetzes nach UMTS Standard TS 123.002.

Im Folgenden seien einige wesentliche Unterschiede zwischen UMTS und GSM aufgeführt. Hinsichtlich der Luftschnittstelle UTRAN (*UMTS Radio Access Network*) wird bei UMTS zwischen Fest- und Mobilstation das Vielfachzugriffsverfahren CDMA (*Code Division Multiple Access*) benutzt. Als Übertragungsfrequenz wird gemäß IMTS-2000 der

Bereich von 2000 MHz (Wellenlänge ca. 15 cm) verwendet. Die Architektur ähnelt sehr der von GSM, mit dem Hauptunterschied, dass bei UMTS die Systemkomponenten über IP miteinander verbunden sind.

Der aus Sicht der Chipkarte größte Unterschied zwischen GSM und UMTS ist jedoch, dass bei UMTS ein völlig neu definiertes Sicherheitsmodul Verwendung findet, das USIM (*universal subscriber identity module*). Es benutzt als normative Grundlage die ISO/IEC 7816 Normenreihe und gewährleistet so erstmals in der Welt der Mobilfunk-Chipkarten eine große Kompatibilität zu anderen gemäß dieser Norm festgelegten Chipkarten, wie beispielsweise EMV 2000 konformen Chipkarten des Zahlungsverkehrs.

Tabelle 13.11 Aufstellung der wichtigsten Normen für die USIM

TS 21.111	USIM and IC card requirements
TS 31.102	Characteristics of the USIM Application
TS 31.110	Numbering system for telecommunication IC card applications
TS 31.111	USIM Application Toolkit (USAT)
TS 31.121	USIM Application Test Specification
TS 31.122	USIM Conformance Test Specification
TS 102.221	Physical and Logical Characteristics
TS 102.222	Administrative Commands

An dieser Stelle sei zum Verständnis der USIM-Funktionen generell empfohlen, im Vorfeld den Abschnitt über die SIM zu studieren, da sich beide Chipkarten-Anwendungen sehr ähneln und im Folgenden vor allem auf die Unterschiede eingegangen wird. Man sieht dabei auch sehr deutlich, dass das UMTS-System aus dem GSM-System hervorgegangen ist und sehr viele der dort bewährten Prinzipien und Mechanismen in UMTS übernommen wurden.

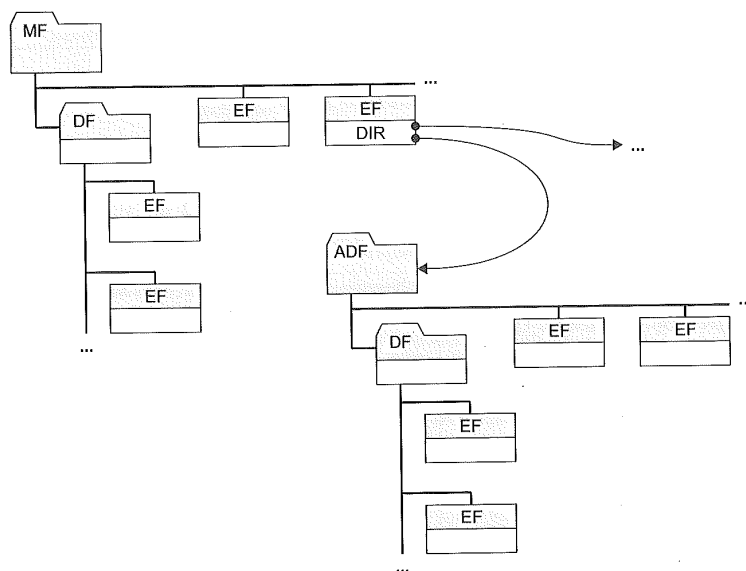
USIM ist die übliche Bezeichnung für die Chipkarten-Anwendung für UMTS, die auf einer UICC residiert. In der Praxis wird jedoch der Begriff USIM nicht nur für die Anwendung, sondern auch für die UMTS-Chipkarte verwendet, obwohl dies eigentlich nicht ganz korrekt ist. Die USIM ist primär der Träger der Identität des Teilnehmers und hat als Hauptaufgabe die Sicherstellung der Echtheit der Mobilstation gegenüber dem Netzwerk und umgekehrt.

Die Spezifikation der USIM fundiert auf der Grundlagenspezifikation für Telekommunikationschipkarten TS 102.221, welche eine UICC (*universal integrated circuit card*) in ihren physikalischen und logischen Parametern charakterisiert. Aufbauend auf dieser Spezifikation einer allgemein verwendbaren Chipkarte für die Telekommunikation, sind in der Anforderungsspezifikation TS 21.111 die Grundlagen für die USIM-Anwendung festgelegt, die wiederum *en detail* in TS 31.102 beschrieben ist. Ergänzend dazu enthält die TS 31.111 die ausführliche Beschreibung des USIM Application Toolkit (USAT). Die Kommandos für die Verwaltung der Anwendungen auf einer UICC sind in der TS 102.222 enthalten, welche mittlerweile auch im Umfeld der SIM als Quasi-Standard benutzt wird. Schlussendlich enthält die TS 31.122 eine Testspezifikation für Konformitätstests.

Tabelle 13.12 Tabelle der charakteristischen physischen, elektrischen und logischen Eigenschaften einer USIM-basierend auf der UICC

Eigenschaft	Bemerkung
Kartenformat	ID-1 oder ID-000
Versorgungsspannung	UICC: 1,8 V und/oder 3 V und/oder 5 V in der Praxis besitzen USIMs typischerweise die folgenden Spannungsbereiche: (1,8 V und 3 V) oder (3 V und 5 V)
Übertragungsprotokolle	PPS (obligatorisch) T = 0 (obligatorisch) T = 1 (optional)
logische Kanäle	bis zu 4
Kommandos	siehe Tabelle 13.13 mit Kommandoüberblick
Zugriffsbedingungen auf Dateien	nach ISO/IEC 7816-9

Ein Chipkarten-Betriebssystem für UICC hat sich in allen wesentlichen Eigenschaften an der ISO/IEC 7816-Normenreihe zu orientieren, es ist deshalb vollständig multiapplikationsfähig. Dies gilt insbesondere für die Kommandos, die Dateiverwaltung und die Zugriffsbedingungen auf Dateien. Diese sind gemäß ISO/IEC 7816-9 regelbasiert und

**Bild 13.25** Die Zusammenhänge zwischen MF, DFs und ADFs in der UICC bzw. USIM. Ein ADF kann mittels des im EF_{DIR} hinterlegten AID selektiert werden und beinhaltet alle Dateien einer Anwendung.

werden über ein EF_{ARR} (*EF access rule reference*) gesteuert. Seitens der Kommandos und der Dateiverwaltung gibt es dennoch viele Gemeinsamkeiten zur SIM. Eine Besonderheit weist die USIM bei den Dateitypen auf, dies ist das ADF (*application dedicated file*). Dies ist ein besonderer Typ von DF, der alle DFs und EFs einer bestimmten Anwendung enthält, jedoch nicht das MF als Wurzel hat. Ein ADF ist in diesem Sinne ähnlich wie das MF zu sehen, da es keinen übergeordnete Dateityp mehr hat. Selektiert wird ein ADF mithilfe eines AIDs, welcher im EF_{DIR} gespeichert ist.

Die USIM stellt auf einer UICC nichts anderes als eine von vielen möglichen Anwendungen in einem eigenen ADF dar. Aufgrund der Multiapplikationsfähigkeit einer UICC ist es auch ohne besondere Mühsal möglich, neben der USIM eine SIM zu realisieren und damit eine Chipkarte zu schaffen, die sowohl bei GSM als auch bei UMTS funktionsfähig ist. Dies wird wohl auch absehbare Zeit sogar die übliche Konstellation sein, da sich mit einer Chipkarte für zwei unterschiedliche Mobilfunktechnologien die Logistikaufwände der Netzbetreiber reduzieren lassen.

Die Dateien einer USIM-Anwendung finden sich größtenteils in gleicher oder zumindest ähnlicher Form auch auf einer SIM wieder. Eine wesentliche Änderung ist lediglich bei der Speicherung der Rufnummern eingetreten. Hier wurde ein verhältnismäßig undurchsichtiges Konzept mit über Zeiger verbundenen optionalen und obligatorischen Dateien festgelegt, in dem unterschiedlichste Daten zu einer Rufnummer bzw. einem Teilnehmer gespeichert werden können.

Tabelle 13.13 Die in TS 102.221 festgelegten Chipkarten-Kommandos für die USIM

Kommando	Kurzbeschreibung
Kategorie Sicherheit	
CHANGE PIN	Ändern der PIN
DISABLE PIN	Abschalten der PIN-Abfrage
ENABLE PIN	Einschalten der PIN-Abfrage
UNBLOCK PIN	Zurücksetzen des abgelaufenen PIN-Fehlbedienungszählers
VERIFY PIN	Überprüfen der PIN
AUTHENTICATE	Authentisierung der USIM durch die äußere Welt
Kategorie Operationen auf Dateien	
INCREASE	Erhöhen eines Zählers in einer Datei
READ BINARY	Lesen aus einer Datei mit „transparenter“ Struktur
READ RECORD	Lesen aus einer Datei mit Record-orientierter Struktur
SEARCH RECORD	Suchen eines Textes in einer Datei mit Record-orientierter Struktur
SELECT	Auswahl einer Datei
STATUS	Lesen von verschiedenen Informationen zur aktuell selektierten Datei
UPDATE BINARY	Schreiben in eine Datei mit „transparenter“ Struktur

Tabelle 13.13 (Fortsetzung)

Kommando	Kurzbeschreibung
UPDATE RECORD	Schreiben in eine Datei mit Record-orientierter Struktur
DEACTIVATE FILE	Reversibles Blocken einer Datei
ACTIVATE FILE	Entblocken einer Datei
Kategorie SIM Application Toolkit	
ENVELOPE	Übergabe von Daten an eine Zusatzanwendung der USIM im Rahmen des USIM Application Toolkit
FETCH	Abholung eines USIM Application Toolkit Kommandos von der USIM an das Mobile Equipment
TERMINAL PROFILE	Aufstellung aller Funktionalitäten des Mobile Equipment hinsichtlich des USIM Application Toolkit
TERMINAL RESPONSE	Übergabe der Antwort des Mobile Equipment auf ein vorangegangenes USIM Application Toolkit-Kommandos der USIM
Kategorie Sonstige	
GET RESPONSE	T = 0 spezifisches Kommando zur Anforderung von Daten von der Chipkarte
MANAGE CHANNEL	Steuerung der logischen Kanäle

Tabelle 13.14 Aufstellung der obligatorischen anwendungsunabhängigen Dateien einer UICC, die sich damit auch auf einer USIM befinden müssen

MF.EF _{DIR}		Erweiterung für bevorzugte Sprache (ELP – extended language preference)
	Beschreibung	Diese Datei enthält Angabe über die auf der Chipkarte enthaltenen Anwendungen.
	Datei	FID = '2F00'; Struktur: linear fixed, Anzahl der Records: ≥1, Dateigröße: x Byte; Zugriffe: READ: immer; UPDATE: ADM
	Dateiinhalt	Jeder Record enthält ein BER-TLV-codiertes Datenobjekt nach ISO/IEC 7816-5 zur Beschreibung einer Anwendung auf der Chipkarte mit der obligatorischen Angabe eines AIDs.
MF.EF _{ICCID}		Identifikationsnummer der Chipkarte (ICCID – ICC identification)
	Beschreibung	Diese Datei enthält eine eindeutige Identifikationsnummer der Chipkarte.
	Datei	FID = '2FE2'; Struktur: transparent, Dateigröße: 10 Byte Zugriffe: READ: immer; UPDATE: nie
MF.EF _{PL}		Bevorzugte Sprache (PL – preferred language)
	Beschreibung	Diese Datei enthält eine Liste der bevorzugten Sprachen für die Benutzerschnittstelle.
	Datei	FID = '2F05'; Struktur: transparent, Dateigröße: 2*x Byte; Zugriffe: READ: immer; UPDATE: PIN
MF.EF _{ARR}		Zugriffsregeln (ARR – access rule reference)
	Beschreibung	Diese Datei enthält eine Liste der Zugriffsregeln für die Dateien direkt unter dem MF.
	Datei	FID = '2F06'; Struktur: linear fixed, Dateigröße: x Byte; Zugriffe: READ: immer; UPDATE: ADM
	Dateiinhalt	Jeder Record enthält eine Zugriffsregel nach ISO/IEC 7816-9.

Tabelle 13.15 Aufstellung der obligatorischen Dateien einer USIM, die noch durch die obligatorischen anwendungsunabhängigen Dateien einer UICC ergänzt werden. Die Codierung der Datenelemente sowie decodierte erläuterte Beispiele zu den Dateien finden sich im Kapitel über die SIM.

MF	Wurzelverzeichnis
	Beschreibung Dies ist das Ursprungsverzeichnis für die gesamte UICC.
	Datei FID = '3F00'
MF.DF_{GSM}	Verzeichnis GSM
	Beschreibung In diesem Verzeichnis sind alle Dateien zusammengefasst, die spezifisch für das GSM-Netzwerk sind.
	Datei FID = '7F20'
MF.DF_{TELECOM}	Verzeichnis Telekom
	Beschreibung In diesem Verzeichnis sind alle Dateien zusammengefasst, die spezifisch für die Dienste sind.
	Datei FID = '7F10'
MF.DF_{PHONEBOOK}	Verzeichnis Telefonbuch
	Beschreibung In diesem Verzeichnis sind alle Dateien zusammengefasst, die zum Telefonbuch gehören.
	Datei FID = '5F3A'
ADF_{USIM}	Anwendungsverzeichnis USIM
	Beschreibung In diesem Verzeichnis sind alle Dateien zusammengefasst, die zur Anwendung USIM gehören.
	Datei RID = RID = 'A0 00 00 00 87'
ADF_{USIM}.DF_{PHONEBOOK}	Verzeichnis Telefonbuch
	Beschreibung In diesem Verzeichnis sind alle Dateien zusammengefasst, die zum Telefonbuch gehören.
	Datei FID = '5F3A'
ADF_{USIM}.DF_{GSM-ACCESS}	Verzeichnis GSM
	Beschreibung In diesem Verzeichnis sind alle Dateien zusammengefasst, die zum Zugriff auf das GSM-Netzwerk notwendig sind.
	Datei FID = '5F3B'
ADF_{USIM}.DF_{MExE}	Verzeichnis MExE (<i>mobile station execution environment</i>)
	Beschreibung In diesem Verzeichnis sind alle Dateien zusammengefasst, die spezifisch für MExE sind.
	Datei FID = '5F3C'
ADF_{USIM}.EF_{IMSI}	internationale Teilnehmernummer (<i>IMSI – international mobile subscriber identity</i>)
	Beschreibung In dieser Datei ist die internationale Teilnehmernummer gespeichert.
	Datei FID = '6F07'; SFI = '07'; Struktur: transparent, Dateigröße: 9 Byte Zugriffe: READ: PIN; UPDATE: ADM

Tabelle 13.15 (Fortsetzung)

ADF_{USIM}.EF_{Keys}		Schlüssel
Beschreibung	Diese Datei enthält den Verschlüsselungsschlüssel CK (<i>ciphering key</i>), den Schlüssel für die Integritätsprüfung IK (<i>integrity key</i>) und den Schlüsselbezeichner KSI (<i>key set identifier</i>).	
Datei	FID = '6F08'; SFI = '08'; Struktur: transparent, Dateigröße: 33 Byte Zugriffe: READ: PIN; UPDATE: PIN	
ADF_{USIM}.EF_{KeysPS}		Schlüssel für paketorientierte Dienste (<i>packet switched domain</i>)
Beschreibung	Diese Datei enthält für die paketorientierten Dienste den Verschlüsselungsschlüssel CKPS (<i>ciphering key packet switched domain</i>), den Schlüssel für Integritätsprüfung IK (<i>integrity key packet switched domain</i>) und den Schlüsselbezeichner KSIPS (<i>key set identifier packet switched domain</i>).	
Datei	FID = '6F09'; SFI = '09'; Struktur: transparent, Dateigröße: 33 Byte Zugriffe: READ: PIN; UPDATE: PIN	
ADF_{USIM}.EF_{HPLMN}		Zeitintervall für die Suche nach dem Heimatnetz (<i>HPLMN – home public land mobile network search period</i>)
Beschreibung	In dieser Datei ist ein Zeitintervall für die Suche nach dem Heimatnetz gespeichert.	
Datei	FID = '6F31'; SFI = '12'; Struktur: transparent, 1 Byte Zugriffe: READ: PIN; UPDATE: ADM	
ADF_{USIM}.EF_{UST}		Dienststeerkennungen (<i>UST – USIM service table</i>)
Beschreibung	In dieser Datei sind in einer Tabelle zusätzlich zum Sprachdienst mögliche und freigeschaltete Dienste angegeben.	
Datei	FID = '6F38'; SFI = '12'; Struktur: transparent, Dateigröße: ≥ 1 Byte Zugriffe: READ: PIN; UPDATE: ADM	
ADF_{USIM}.EF_{ACC}		Zugriffskontrollklasse (<i>ACC – access control class</i>)
Beschreibung	In dieser Datei sind Daten über die Netzzugriffsversuche und Zugriffsprioritäten gespeichert.	
Datei	FID = '6F78'; SFI = '12'; Struktur: transparent, 1 Byte Zugriffe: READ: PIN; UPDATE: ADM	
ADF_{USIM}.EF_{FPLMN}		verbotene Netzbetreiber (<i>FPLMN – forbidden public land mobile network</i>)
Beschreibung	In dieser Datei ist eine Liste der verbotenen Netzbetreiber gespeichert.	
Datei	FID = '6F7B'; SFI = '0D'; Struktur: transparent, x*3 Bytes Zugriffe: READ: PIN; UPDATE: PIN	
ADF_{USIM}.EF_{LOCI}		Ortsinformation (<i>LOCI – location information</i>)
Beschreibung	In dieser Datei werden Informationen über den aktuellen Ort des Mobiltelefons gespeichert.	
Datei	FID = '6F7E'; SFI = '0B'; Struktur: transparent, Dateigröße: 11 Byte Zugriffe: READ: PIN; UPDATE: PIN	

Tabelle 13.15 (Fortsetzung)

ADF _{USIM} .EF _{AD}		Verwaltungsdaten (AD – administrative data)
	Beschreibung	In dieser Datei werden Verwaltungsinformationen über die USIM gespeichert.
	Datei	FID = '6FAD'; SFI = '03'; Struktur: transparent, Dateigröße: 4+x Byte Zugriffe: READ: immer; UPDATE: ADM
ADF _{USIM} .EF _{ECC}		Verwaltungsdaten (AD – administrative data)
	Beschreibung	In dieser Datei werden Notrufnummern gespeichert.
	Datei	FID = '6FB7'; SFI = '01'; Struktur: linear fixed, Dateigröße: 4+x Byte Zugriffe: READ: immer; UPDATE: ADM
ADF _{USIM} .EF _{PSLOC1}		Ortsinformation für paketorientierte Dienste (PSLOC1 – packet switched location information)
	Beschreibung	In dieser Datei werden für die paketorientierten Dienste Informationen über den aktuellen Ort des Mobiltelefons gespeichert.
	Datei	FID = '6F73'; SFI = '0C'; Struktur: transparent, Dateigröße: 14 Byte Zugriffe: READ: PIN; UPDATE: PIN
ADF _{USIM} .EF _{START-HFN}		Initialisierungswerte für die Hyperframe-Nummer
	Beschreibung	In dieser Datei werden Informationen für die Datenübertragung auf der Luftschnittstelle gespeichert.
	Datei	FID = '6F7B'; SFI = '0F'; Struktur: transparent, Dateigröße: 6 Byte Zugriffe: READ: PIN; UPDATE: PIN
ADF _{USIM} .EF _{THRESHOLD}		Maximalwerte für die Hyperframe-Nummer
	Beschreibung	In dieser Datei werden Informationen für die Datenübertragung auf der Luftschnittstelle gespeichert.
	Datei	FID = '6F5C'; SFI = '10'; Struktur: transparent, Dateigröße: 3 Byte Zugriffe: READ: PIN; UPDATE: ADM
ADF _{USIM} .EF _{ARR}		Zugriffsregeln (ARR – access rule reference)
	Beschreibung	Diese Datei enthält eine Liste der Zugriffsregeln für die Dateien in dem ADF.
	Datei	FID = '6F06'; SFI = '17'; Struktur: linear fixed, Dateigröße: x Byte; Zugriffe: READ: immer; UPDATE: ADM
MF.EF _{NETPAR}		Netzwerkparameter (NETPAR – network parameters)
	Beschreibung	In dieser Datei werden Informationen über physikalische Parameter der Datenübertragung auf der Luftschnittstelle gespeichert.
	Datei	FID = '2FC4'; Struktur: transparent, Dateigröße: ≥ 46 Byte; Zugriffe: PIN: immer; UPDATE: PIN

Völlig analog zu CHV 1 und CHV 2 bei einer SIM existieren bei einer USIM PIN 1 und PIN 2. Die Zugriffsbedingungen der Dateien sind so festgelegt, dass der Kartenbenutzer PIN 1 kennt und der Kartenbesitzer PIN 2. Damit lässt sich ein Großteil der Funktionalität des Endgeräts und der möglichen Dienste flexibel steuern.

Zur Authentisierung der USIM im UMTS-System wird eine Reihe von kryptografischen Funktionen namens f1, f2, f3, f4, f5 (*function 1 ... 5*) benutzt. Diese dienen zur Authenti-

sierung von Netzwerk und USIM als auch zum Aufbau einer kryptografisch gesicherten Datenübertragung auf der Luftschnittstelle. Der Kern dieser *Security Functions* ist ein symmetrischer Kryptoalgorithmus, der mit zusätzlich verknüpften Eingangswerten parametrisiert werden kann. Als Beispielalgorithmus ist seitens der USIM-Spezifikation für f1 bis f5 der MILENAGE-Algorithmus vorgeschlagen, der in seinem Kern auf den AES basiert.

Ganz analog zum SIM Application Toolkit (SAT) der SIM wurde bei der USIM ein zu SAT nahezu identisches USIM Application Toolkit (USAT) spezifiziert. Dieses wird ebenfalls völlig analog zu SAT für die Realisierung von Mehrwertdiensten auf der USIM benutzt. Seitens ETSI wurde für die USIM auch ein Mikrobrowser namens USAT-Interpreter spezifiziert, der aber nicht, wie es der Name nahe legt, auch auf SIMs implementiert werden kann.

13.4 Mikrobrowser

Einen bedeutenden Anteil am Siegeszug des World Wide Web kann mit Sicherheit den Browsern zugesprochen werden. Damit wurde es möglich, ohne die rechnerindividuelle Installation von Programmen auf einem entfernten Server gespeicherte Inhalte in Form von Hypertextdokumenten zu betrachten, zwischen ihnen zu navigieren und in die Hypertextdokumente eingebetteten Programmcode ohne unerwünschte Seiteneffekte auf dem Client-Rechner auszuführen.

Einfach aufgebaute, wenig Speicher und Rechenleistung benötigende Browser werden oft auch als Mikrobrowser bezeichnet. In der Regel können diese keinen in Hypertextdokumente eingebetteten Programmcode ausführen und sind ansonsten hinsichtlich Speicherplatzverbrauch und notwendiger Rechenleistung an das Zielsystem angepasst. Mikrobrowser können als Form einer Erweiterung eines Chipkarten-Betriebssystems gesehen werden. Man kategorisiert dies auch als SIM Mikrobrowser. Die Alternative dazu ist die Integration in die Software des Mobiltelefons, auch als ME (*mobile equipment*) Mikrobrowser bezeichnet. Das wohl bekannteste Beispiel ist der Browser für WAP, welcher vollständig im Mobiltelefon ausgeführt wird und nur optional auf eine Chipkartenanwendung namens WIM¹⁸ (*wireless identification module*) zwecks Benutzung von Kryptofunktionen zugreift.

Mikrobrowser auf der SIM wurden erstmalig im Jahr 1998 von der Firma Across Wireless, heute Smarttrust [Smarttrust], im GSM-Umfeld bekannt gemacht. Sie können einen auf die Belange von SIMs optimierten WML-Dialekt interpretieren und verfügen auch über die Möglichkeit, durch nachladbare Plug-ins zusätzliche Funktionen – wie beispielsweise die Erstellung von digitalen Signaturen mit RSA-Algorithmus – nachzuladen. Die Größe des Programmcodes eines typischen Mikrobrowsers beträgt ca. 15 bis 25 kByte, zusätzlich sind noch 1 bis 2 kByte für Arbeitspuffer im RAM oder EEPROM notwendig.

Im Jahr 1999 wurde von den Firmen Gemplus, Giesecke und Devrient, ORGA und Schlumberger das Konsortium SIM Alliance [SIM Alliance] gegründet, um Dienstleistungen, die für WAP entwickelt wurden, auch auf nicht WAP-fähigen Mobiltelefonen zu ermöglichen. Dies war zu diesem Zeitpunkt noch sehr interessant, da die Durchdringung von WAP-fähigen Mobiltelefonen gering war. Zur Realisierung muss die SIM über einen SIM-

¹⁸ Siehe auch Abschnitt 13.5 „Die WIM“.

Alliance-fähigen Mikrobrowser verfügen (auch S@T-Browser genannt) und das Mobiltelefon lediglich die Phase 2+ von GSM unterstützen. Damit ist die SIM fähig, das Mobiltelefon über SIM Application Toolkit so weit zu steuern, dass ein Teil von WML-Inhalten und deren Funktionalität abgebildet werden kann.

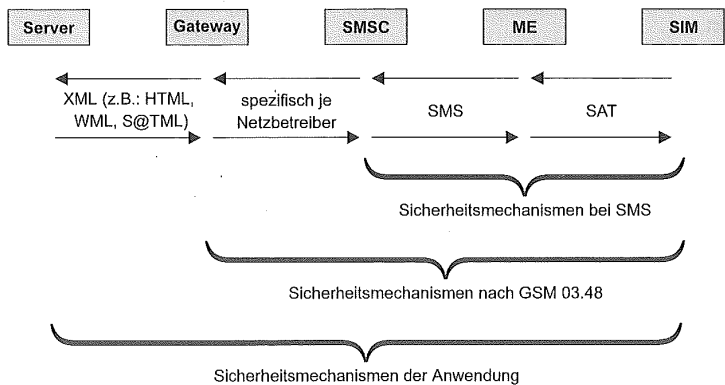


Bild 13.26 Prinzipieller Überblick der Datentübertragung und der unterschiedlichen und hierarchisch abgestuften Sicherheitsmechanismen bei Mikrobrowser-Anwendungen.

Ein großer Vorteil der Mikrobrowser auf der SIM gegenüber ihrem Äquivalent im Mobile Equipment ist die sichere Umgebung ihrer Ausführung. Es ist durchaus denkbar, einen WAP-Browser, welcher im Mobile Equipment abläuft, in vielen seiner Funktionen zu manipulieren, da sowohl Prozessor als auch Speicher Manipulationen gegenüber zugänglich sind. Ein SIM Mikrobrowser ist jedoch vollständig in einer SIM integriert und kann somit von allen chipkartenspezifischen Schutzmaßnahmen profitieren. Mit einem SIM Browser kann auch ohne Abstriche an Sicherheit eine Ende-zu-Ende-Sicherheit auf Anwendungsebene zwischen der Anwendung auf dem Anwendungsserver und der SIM realisiert werden. Dies ist mit einem WAP-Browser so nicht möglich.

Tabelle 14.16 Beispiel für eine einfache WML-Anwendung. Das WML-Dokument befindet sich auf einem Server und wird mit einem Konverter in einen 20 Byte langen WML-Bytecode umgewandelt, der dann mit einer SMS zum Mikrobrowser auf der SIM übertragen wird. Dieser interpretiert den WML-Bytecode und gibt dann mit dem SIM Application Toolkit Kommando DISPLAY TEXT die Schrift „Hello World“ auf dem Display des Mobile Equipments aus.

<pre><?xml version="1.0" encoding="ISO-8859-1"?> <wml> <card> Hello World </card> </wml></pre>	Angabe der WML-Version und der Zeichencodierung. Beginn des WML-Dokuments Beginn einer neuen <i>card</i> Auf dem Display auszugebender Text Ende der <i>card</i> Ende des WML-Dokuments
<pre>'00 12 02 0C 48 65 6C 6C 6F 20 57 6F 72 6C 64 20 06 00 06 00'</pre>	In WML-Bytecode umgesetztes WML-Dokument

Sowohl der Smarttrust-Browser als auch der SIM Alliance-Browser sind firmen- bzw. konsortienspezifische Mikrobrowser und stammen nicht von einer Normungsinstanz. Sie sind auch untereinander nicht kompatibel. Zum Ende des Jahres 2001 hat jedoch ETSI die erste Version der Spezifikationen für den so genannten USAT (*USIM application toolkit*) Interpreter veröffentlicht, welches das Ergebnis einer Mitte 2000 gegründeten Arbeitsgruppe zu diesem Themenkomplex ist. Im Gegensatz zu seinem Namen kann der USAT Interpreter im Übrigen sehr wohl auch auf SIMs und nicht nur USIMs realisiert werden. Es ist wahrscheinlich, dass der USAT Interpreter mittelfristig den Smarttrust-Browser und den SIM Alliance-Browser vollständig ablösen wird. Der Name Interpreter rührt auch von der neuen Funktionalität von Verzweigungen im interpretierten Bytecode her. Weiterhin existieren unterschiedliche Variablentypen, die vom zu interpretierenden Bytecode auch über Sitzungen hinweg als Zwischenspeicher verwendet werden können. Browser sind vom Ursprung her typische Vertreter des Pull-Prinzips, d.h. der Informationsweitergabe durch das Abholen von Information von einem übergeordneten System durch ein untergeordnetes System. Der USAT-Interpreter beinhaltet jedoch auch eine Push-Funktion, also die Informationsweitergabe durch das Versenden von Information von einem übergeordneten System an ein untergeordnetes. Dies ist für bestimmte Anwendungen sehr interessant, da damit der Mobilfunkbenutzer bezüglich bestimmter Dinge explizit vom Anwendungsserver informiert werden kann.

Tabelle 13.17 Aufstellung der wichtigsten Normen für den USAT Interpreter

TS 22.112	USAT Interpreter – Stage 1
TS 31.112	USAT Interpreter Architecture Description; Stage 2
TS 31.113	USAT Interpreter Byte Codes
TS 31.114	USAT Interpreter Protocol and Administration

Die Verwendung von Mikrobrowsern auf einer Chipkarte setzen in der Regel ein Online-System wie bei GSM oder UMTS voraus. Die darzustellenden bzw. auszuführenden Inhalte befinden sich auf einem Anwendungsserver (*application server*) des Inhaltsanbieters (*content provider*). Dieser Server ist über übliche Internet-Protokolle und kryptografische Kommunikationsabsicherung (SSL, TLS) mit einem Gateway-Server verbunden, der zum Anwendungsserver hin die entsprechende Kommunikationssicherheit gewährleistet. Der Gateway-Server ist seinerseits im Falle der Kommunikation zur SIM mittels Kurznachrichten an ein SMSC (*short message service center*) angebunden. Die an die SIM zu sendenden Daten werden nun in die entsprechenden Datenpakete nach GSM 03.48 umformatiert und zugleich auch mit der festgelegten kryptografischen Kommunikationssicherheit versehen. Anschließend werden die Datenpakete mittels Kurznachrichten über das GSM-System transparent an die SIM gesendet. Diese erkennt, dass es sich um Daten nach GSM 03.48 handelt, entsprechend decodiert und notfalls aus mehreren Kurznachrichten zusammengesetzt und im Gutfall an den SIM-Mikrobrowser weitergereicht. Dieser interpretiert die Nachricht und führt die entsprechenden SIM Application Toolkit Kommandos aus. Anschließend wird optional in umgekehrter Reihenfolge eine Nachricht zum Anwendungsserver erzeugt und abgeschickt.

Es gibt Fälle, in denen Mikrobrowser ihre Nachrichten nicht als Online-Inhalt von einem über das Netzwerk angeschlossenen Anwendungsserver erhalten, sondern der XML-konforme Bytecode in einer Datei auf der SIM abgelegt ist. Diese Offline-Inhalte sind eine elegante Variante, um häufig benutzte Anwendungen für einen schnellen Zugriff lokal abzulegen und nebenbei die Netzlast zu reduzieren. Der daraus resultierende Nachteil besteht jedoch darin, dass diese lokal gespeicherten Anwendungen vom Hintergrundsystem explizit aktualisiert werden müssen. Gerade dies ist aber der Hauptvorteil bei zentral auf einem Server gespeicherten Anwendungen.

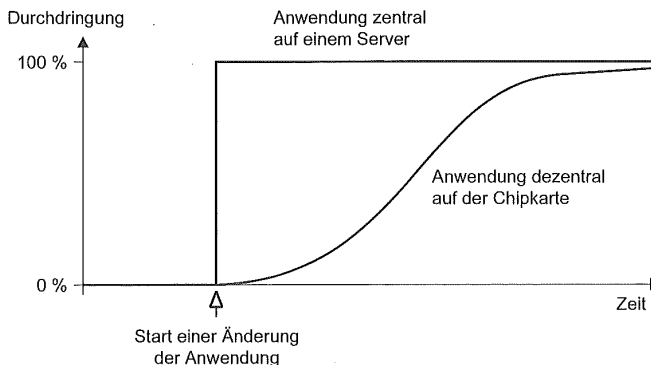


Bild 13.27 Verlauf der Durchdringung einer neuen Anwendung im Vergleich zwischen einer zentralen Speicherung auf einem Server und einer dezentralen Speicherung auf der SIM. Abhängig vom Mobilfunknetz und der jeweiligen Anwendung kann es bis zu einer Durchdringung von 90 % mehrere Wochen dauern.

13.5 Die WIM (Wireless Identification Module)

Im Jahr 2000 wurde weltweit der breiten Bevölkerung das Wort WAP massiv in den Massenmedien als die lang ersehnte neue Internet-Technologie für mobile Endgeräte angepriesen. Aus vielschichtigen Gründen, die von fehlenden Inhalten über nicht verfügbare Endgeräte, niedrige Datenübertragungsraten bis hin zur enttäuschenden Darstellung der Inhalte auf zum Teil textbasierten schwarz/weiß Displays ging, konnte WAP bei den Endkunden nicht Fuß fassen. Mittlerweile wird WAP oft als Synonym für milliardenteure sinnlose Investitionen in eine am Endkunden vorbeigehende Technik genannt. Es könnte jedoch durchaus sein, dass nach der Phase der beinahe schrankenlosen Enttäuschung früher oder später der Realismus obsiegt und WAP als einfache Möglichkeit gesehen und benutzt wird, unterwegs weitgehend textbasierte Inhalte zu betrachten.

Der Begriff WAP (*wireless application protocol*) ist eigentlich der Name für eine Reihe von Spezifikationen zur Realisierung der Anbindung von mobilen Endgeräten (Mobiltelefone, PDAs, ...) über ein drahtloses Netzwerk (GSM, CDMA, TDMA) an einen Server zum unmittelbaren Informationsaustausch. Die heute übliche Anwendung von WAP ist die Realisierung von Internetdiensten auf einem Mobiltelefon, weitgehend unabhängig vom jeweiligen Mobilfunkstandard. Die Bezeichnung Wireless Application Protocol steht je

doch neben der im üblichen Sprachgebrauch gemeinten Technologie auch für das Protokoll zwischen Endgerät und Hintergrundsystem.

Das international agierende Normungsgremium für WAP ist das seit Juni 1997 existierende und von den Firmen Phone.com, Ericsson, Motorola und Nokia gegründete WAP Forum [WAP] und setzt sich aus Vertretern von mittlerweile über 350 Firmen zusammen.

Seit Juni 2000 beinhalten die Spezifikationen für WAP auch ein Sicherheitsmodul im Mobilgerät namens WIM (*wireless identification module*). Erst damit kann in vernünftiger Art und Weise eine sichere Kommunikation mit den Anwendungen im Hintergrundsystem aufgebaut werden. Zwei mögliche Varianten einer WIM sind festgelegt. Die WIM als Anwendung auf einer eigenen Chipkarte oder die WIM als zusätzliche Anwendung auf einer SIM, USIM oder UICC. Da sich Dual-Slot-Mobiltelefone im Markt nicht durchgesetzt haben, wird sich die zweite Option, d. h. die WIM, als zusätzliche Anwendung auf einer vorhandenen Chipkarte etablieren.

Die Grundlage der Spezifikation für die WIM ist die PKCS #15 Spezifikation [RSA] für digitale Signaturanwendungen auf Chipkarten, welche mittlerweile auch in kompatibler Form auch von der ISO/IEC 7816-15 abgedeckt ist. Zum Verständnis des folgenden Textes wird zumindest überblickshaftes Wissen um PKCS #15-kompatible Signaturanwendungen vorausgesetzt. Bei Bedarf sollte deshalb der entsprechende Abschnitt „14.5 Signaturanwendung nach PKCS #15“ vorab gelesen werden.

Die Hauptaufgaben der WIM sind primär die kryptografischen Funktionen für die Sicherung der WAP-Datenübertragung bei der Übertragungsschicht WTLS (*wireless transport layer security*). Interessanterweise können die Funktionen der WIM jedoch nicht nur im Umfeld von WAP genutzt werden, sondern lassen sich auch im Rahmen des SIM Application Toolkits verwenden. Damit ist die WIM eine allgemein verwendbare Signaturanwendung auf Telekommunikationskarten. Deshalb befindet sie sich mittlerweile auch standardmäßig auf vielen SIMs bzw. USIMs.

Die WIM-Spezifikation enthält in abstrakter Form eine Reihe von Primitiven, welche grundlegende Funktionen beschreiben. Tabelle 13.18 enthält einen Überblick über diese generischen Funktionen, welche im WTLS-Protokoll benutzt werden. Diese Primitive sind in einem weiteren Schritt mit den entsprechenden Kommandos bzw. Kommandosequenzen verbunden.

Tabelle 13.18 Die in der WIM-Spezifikation festgelegten Primitive mit der Verbindung zu den entsprechenden Chipkarten-Kommandos

Primitiv	Chipkarten-Kommando
Gerätesteuerung (<i>device control</i>)	
Open Service	MANAGE CHANNEL
Close Service	MANAGE CHANNEL
Verifikation (<i>verification</i>)	
Perform Verification	VERIFY
Disable Verification Requirement	DISABLE VERIFICATION REQUIREMENT
Enable Verification Requirement	ENABLE VERIFICATION REQUIREMENT

Tabelle 13.18 (Fortsetzung)

Change Reference Data	CHANGE REFERENCE DATA
Unblock Reference Data	RESET RETRY COUNTER
Datenzugriff (data access)	
Open File	SELECT
Close File	---
Read Binary	READ BINARY
Update Binary	UPDATE BINARY
Kryptografie (cryptography)	
Compute Digital Signature	MANAGE SECURITY ENVIRONMENT PERFORM SECURITY OPERATION
Verify Signature	MANAGE SECURITY ENVIRONMENT PERFORM SECURITY OPERATION
Get Random	ASK RANDOM
Key Transport	MANAGE SECURITY ENVIRONMENT PERFORM SECURITY OPERATION
Key Agreement	MANAGE SECURITY ENVIRONMENT PERFORM SECURITY OPERATION
Derive Master Secret	MANAGE SECURITY ENVIRONMENT
PHash	MANAGE SECURITY ENVIRONMENT PERFORM SECURITY OPERATION
Decipher	MANAGE SECURITY ENVIRONMENT PERFORM SECURITY OPERATION
Ausnahmen (exception)	
Exception	dies kann jedes Kommando betreffen

Alle Dateien der WIM haben generell die Struktur transparent, um die Zugriffe zu vereinfachen und keinen unnötigen Aufwand zu verursachen. Informationstechnisch erschwert dies die Zugriffe nicht, da die Zeigerstruktur, die hinter PKCS #15 steht, einen verhältnismäßig übersichtlichen und erweiterbaren Zugriff auf die jeweiligen Datenobjekte gewährleistet. Alle Daten sind im ASN.1-Format beschrieben und mit DER (*distinguished encoding rules*) codiert.

Zusätzlich zu den üblichen PKCS #15-Dateien und Datenobjekten existiert bei der WIM noch ein *Peers* und ein *Sessions* Datenobjekt. Diese beiden Datenobjekte dienen zur Speicherung von spezifischen Verbindungsdaten, so dass bei einer folgenden Sitzung ein verkürzter Authentisierungs-Handshake möglich ist.

Hinsichtlich der physikalischen Eigenschaften muss sich die WIM an den einschlägigen GSM-Spezifikationen, wie beispielsweise GSM 11.11 orientieren. Sie muss das Übertragungsprotokoll T = 0 unterstützen; optional kann T = 1 zusätzlich von der WIM angeboten werden. Dies lässt sich über einen PPS bei Bedarf vom Terminal auswählen. Tabelle 13.19 enthält einen Überblick über die grundsätzlichen Eigenschaften einer WIM.

Tabelle 13.19 Tabelle einiger grundsätzlicher Eigenschaften einer WIM

Eigenschaft	Bemerkung
Kartengröße ID-000	optional
Kartengröße ID-1	optional
Kommandos	siehe Tabelle 13.20 mit Kommandoüberblick
logische Kanäle	optional
Übertragungsprotokoll T = 0	obligatorisch
Übertragungsprotokoll T = 1	optional
Versorgungsspannung 3 V	obligatorisch
Versorgungsspannung 5 V	optional

Die Kommandos, die für die Ausführung der Funktionen mit der WIM-Anwendung notwendig sind, orientieren sich gänzlich an der ISO/IEC 7816-4 und -8. Dies macht es verhältnismäßig einfach, eine WIM auf einer Chipkarte mit normkonformem Multiapplikations-Betriebssystem wie beispielsweise einer UICC zu realisieren.

Um die WIM-Anwendung aufzurufen, wird in der Praxis – sofern vom Betriebssystem unterstützt – ein neuer logischer Kanal zur Chipkarte geöffnet; als Erstes findet dabei die Anwendungsselektion mittels AID statt. Dabei hat der RID den Wert 'A0 00 00 00 63' und der dazugehörige PIX ist die ASCII-Codierung von "WAP-WIM" mit dem Wert '57 41 50 2D 57 49 4D'. Danach laufen die üblichen Kommandosequenzen für WTLS bzw. die Signaturanwendung ab.

Tabelle 13.20 Überblick der in der WIM-Spezifikation festgelegten Chipkarten-Kommandos

Kommando	Kurzbeschreibung
Logische Kanäle MANAGE CHANNEL	Steuerung der logischen Kanäle. Optionales Kommando nach ISO/IEC 7816-4.
Verifikation VERIFY ENABLE VERIFICATION REQUIREMENT DISABLE VERIFICATION REQUIREMENT CHANGE REFERENCE DATA RESET RETRY COUNTER	Überprüfung von übergebenen PIN-Daten. Kommando nach ISO/IEC 7816-4. Einschalten der PIN Abfrage. Optionales Kommando nach ISO/IEC 7816-8. Abschalten der PIN Abfrage. Optionales Kommando nach ISO/IEC 7816-8. Ändern der PIN. Kommando nach ISO/IEC 7816-8. Rücksetzen eines Fehlbedienungs Zählers. Kommando nach ISO/IEC 7816-8.
Datenspeicherung SELECT READ BINARY UPDATE BINARY	Auswahl einer Datei. Kommando nach ISO/IEC 7816-4. Lesen aus einer Datei mit „transparenter“ Struktur. Kommando nach ISO/IEC 7816-4. Schreiben in eine Datei mit „transparenter“ Struktur. Kommando nach ISO/IEC 7816-4.

Tabelle 13.20 (Fortsetzung)

Kryptografische Operationen MANAGE SECURITY ENVIRONMENT PERFORM SECURITY OPERATION ASK RANDOM	Ändern der Parameter für die Benutzung kryptografischer Algorithmen in der Chipkarte. Kommando nach ISO/IEC 7816-8. Ausführen eines kryptografischen Algorithmus in der Chipkarte. Kommando nach ISO/IEC 7816-8. Anforderung einer Zufallszahl von der Chipkarte. Kommando nach ISO/IEC 7816-4.
Diverses GET RESPONSE	T = 0 Kommando zur Anforderung von Daten von der Chipkarte. Kommando nach ISO/IEC 7816-4.

Tabelle 13.21 Beispiel für die typischen Dateien einer WIM19

MF.DF _{WIM}	WIM-Anwendung
Beschreibung	In diesem Verzeichnis sind alle Dateien zusammengefasst, die zur PKCS #15-konformen Anwendung WIM gehören.
Datei	AID = 'A0 00 00 00 63 57 41 50 2D 57 49 4D'
MF.DF _{WIM} .EF _{TokenInfo}	Verzeichnisdatei der privaten Schlüssel (PrKDF – private key directory file)
Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen privaten Schlüssel sowie entsprechende Zeiger auf diese Schlüssel.
MF.DF _{WIM} .EF _{ODF}	Objektverzeichnisdatei (ODF – object directory file)
Beschreibung	Diese Datei enthält ein Verzeichnis aller vorhandener Dateien mit Datenobjekten sowie entsprechende Zeiger auf diese Dateien.
MF.DF _{WIM} .EF _{AODF}	Verzeichnisdatei der Authentisierungsobjekte (AODF – authentication object directory file)
Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen Authentisierungsobjekte sowie entsprechende Zeiger auf diese Objekte.
MF.DF _{WIM} .EF _{PrKDF}	Verzeichnisdatei der privaten Schlüssel (PrKDF – private key directory file)
Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen privaten Schlüssel sowie entsprechende Zeiger auf diese Schlüssel in EF _{PrKDFData_1} und EF _{PrKDFData_2} .
MF.DF _{WIM} .EF _{PrKDFData_1}	Private Schlüssel für Authentisierung und Schlüsselaustausch
Beschreibung	Diese Datei enthält die privaten Schlüssel für Authentisierung und Schlüsselaustausch.
MF.DF _{WIM} .EF _{PrKDFData_2}	Private Schlüssel für digitale Signatur
Beschreibung	Diese Datei enthält die privaten Schlüssel für digitale Signatur.
MF.DF _{WIM} .EF _{CDF_1}	Verzeichnisdatei der Zertifikate für Benutzer (CDF – certificate directory file)
Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen Zertifikate für den Benutzer sowie entsprechende Zeiger auf diese Schlüssel EF _{CDFData_1} und EF _{CDFData_2} .

¹⁹ Siehe auch Abschnitt 14.5 „Signaturanwendung nach PKCS #15“.

Tabelle 13.21 (Fortsetzung)

MF.DF _{WIM} .EF _{CDFData_1}	Authentisierung und Schlüsselaustausch Zertifikate für Benutzer
Beschreibung	Diese Datei enthält die Zertifikate für Authentisierung und Schlüsselaustausch für den Benutzer.
MF.DF _{WIM} .EF _{CDFData_2}	Signatur Zertifikate für Benutzer
Beschreibung	Diese Datei enthält die Zertifikate für digitale Signatur für den Benutzer.
MF.DF _{WIM} .EF _{CDF_2}	Verzeichnisdatei der Zertifikate der Zertifizierungsinstanz (CDF – certificate directory file)
Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen Zertifikate der Zertifizierungsinstanz, welche vom Benutzer geändert werden können, sowie entsprechende Zeiger auf diese Zertifikate EF _{CDFData_3} .
MF.DF _{WIM} .EF _{CDFData_3}	Zertifikate der Zertifizierungsinstanz für Benutzer
Beschreibung	Diese Datei enthält ein Zertifikat der Zertifizierungsinstanz, welche vom Benutzer geändert werden können.
MF.DF _{WIM} .EF _{CDF_3}	Verzeichnisdatei der nur lese Zertifikate der Zertifizierungsinstanz (CDF – certificate directory file)
Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen Zertifikate der Zertifizierungsinstanz, welche nicht geändert werden können, sowie entsprechende Zeiger auf diese Zertifikate EF _{CDFData_4} .
MF.DF _{WIM} .EF _{CDFData_4}	Nur lese Zertifikate der Zertifizierungsinstanz
Beschreibung	Diese Datei enthält die Zertifikate der Zertifizierungsinstanz, welche nicht geändert werden können.
MF.DF _{WIM} .EF _{DODF}	Verzeichnisdatei der Datenobjekte (DODF – data object directory file)
Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen Hauptgeheimnisse (<i>master secrets</i>) für die WTLS-Sitzung sowie entsprechende Zeiger auf diese Objekte.
MF.DF _{WIM} .EF _{DODFData_1}	Datenobjekte
Beschreibung	Diese Datei enthält die Hauptgeheimnisse (<i>master secrets</i>) für die WTLS-Sitzung.
MF.DF _{WIM} .EF _{Peers}	Peers Datei
Beschreibung	Diese Datei enthält Daten für den Authentisierungs-Handshake von WTLS.
MF.DF _{WIM} .EF _{Sessions}	Sessions Datei
Beschreibung	Diese Datei enthält Daten für den Authentisierungs-Handshake von WTLS.
MF.DF _{WIM} .EF _{UnusedSpace}	Verzeichnisdatei des Freispeichers
Beschreibung	Diese Datei enthält ein Verzeichnis des in den EFs der WIM-Anwendung vorhandenen, doch nicht genutzten Speicherplatzes.

13.6 Öffentliches Kartentelefon in Deutschland

Ab dem Sommer 1989 wurde in Deutschland von der Telekom mit der bundesweiten Einführung von öffentlichen Kartentelefonen begonnen. Dem gingen mehrere Feldversuche mit Systemen verschiedener Hersteller voraus. Diese begannen schon im Juni 1983 in vier verschiedenen Gebieten mit unterschiedlicher Struktur (Ballungsräume, Städte und ländliche Gebiete) in Deutschland. In diesen Versuchen wurden ebenfalls unterschiedliche Kartentypen, wie Magnetstreifenkarte, Karte mit Hologramm und verschiedene Kartenmaterialien (Papier, Kunststoff) getestet.

Das Ergebnis dieser ersten Feldversuche war, dass Speicherkarten aus Kunststoff der am besten geeignete Kartentyp für Kartentelefone sind. Ausschlaggebend dafür war die damit erreichbare Sicherheit und Aufwärtskompatibilität gegenüber anderen Kartentypen. Im Dezember 1986 und im Juli 1987 startete man dann groß angelegte Feldversuche mit einem auf Speicherkarten basierenden System in 16 Großstädten, welche erfolgreich im Mai 1989 abgeschlossen wurden. Im Frühjahr 1999 gab es in Deutschland über 90 000 öffentliche Kartentelefone und über 300 Millionen verkaufte Guthabekarten seit Einführung des Systems. Weltweit waren es damals im Übrigen über 1,1 Milliarden Karten und 8,3 Millionen Kartentelefone.

Die in Deutschland von der Telekom installierten Kartentelefone können prinzipiell mit zwei verschiedenen Kartentypen betrieben werden. Der Erste ist die in sehr großen Stückzahlen produzierte und vertriebene Guthabekarte, die sich mittlerweile in fast jeder Geldbörse wieder findet. Sie ist, kaufmännisch gesehen, eine vom Kunden vorbezahle elektronische Geldbörse. Technisch gesehen ist sie eine Speicherkarte mit einem irreversiblen Zähler, einem Sicherheitsmerkmal und synchroner Datenübertragung.

Die andere Variante der Telefonkarte sind die wenig verbreiteten Buchungskarten. Ihrer Natur nach sind sie Kreditkarten, da der Benutzer seine Einheiten erst nach Inanspruchnahme der Leistung (d. h. nach dem Telefonieren) durch monatliche Abbuchung von seinem Konto bezahlen muss. Technisch realisiert werden die Buchungskarten durch Chipkarten mit Mikrocontroller. Die Datenübertragung zum Kartentelefon findet dabei mit dem blockorientierten asynchronen Datenübertragungsprotokoll T = 14 statt. Da diese Karte keine große Bedeutung hat, wird sie hier nicht weiter erwähnt.

Das Gesamtsystem der Kartentelefone ist dezentral mit mehreren aufeinander aufbauenden Ebenen von Computern angelegt. Die unteren Ebenen können im Falle eines Systemzusammenbruchs mehrere Tage völlig autark operieren, ohne dass es zu Einschränkungen im Telefonbetrieb kommt. Die beiden Teile des Systems der Telekom, mit denen der normale Benutzer üblicherweise Kontakt hat, also Telefonkarte und Kartentelefon, sind hier im Folgenden beschrieben.

Um die sehr harten Einsatzbedingungen möglichst lange zu erfüllen, hat das Kartentelefon ein sehr stabiles Metallgehäuse, in dem Aussparungen für die Bedien- und Anzeigeelemente, wie Tastatur und Display, vorhanden sind. Das Terminal ist von der restlichen Elektronikbaugruppe vollständig galvanisch getrennt und zusätzlich noch kurzschlussfest, um Vandalismus und Zerstörungen, die auf diese Weise versucht werden, abzuwehren.

Gesteuert wird das Kartentelefon von einem leistungsfähigen Mikrocontroller. Die Steuerungssoftware kann über Fernladung auf den neuesten Stand gebracht werden. Damit muss ein Software-Update nicht mehr durch einen Servicetechniker vor Ort durch Aus-

tausch von EPROMs durchgeführt werden. Der Steuerrechner des Kartentelefons kann auch während eines Gesprächs mit einem DOV-Modem (*data over voice – DOV*) direkt Informationen mit den übergeordneten Systemebenen austauschen. Dies ist vor allem bei den Buchungskarten notwendig, da hier die anfallenden Gebühren sofort zur Verrechnung an das Hintergrundsystem weitergereicht werden.

Die Kommunikation mit der Guthaben- oder Buchungskarte übernimmt ebenfalls der zentrale Steuerrechner des Kartentelefons. Im Falle einer Buchungskarte korrespondiert er über das von der Telekom spezifizierte und nur in Deutschland verwendete T=14-Protokoll mit der Karte, im anderen Fall über das jeweilige synchrone Protokoll für die Speicherkarte.

Die eingebauten Terminals können die Speicherkarte mit externen Programmierspannungen von 5 Volt bis 25,5 Volt in 255 Schritten versorgen. Da aber praktisch alle neuen Speicherkarten die Programmierspannung für das EEPROM intern aus der normalen Versorgungsspannung mit einem Spannungswandler erzeugen, ist dies technisch eigentlich nicht mehr notwendig. Der Grund dafür liegt in der geforderten Kompatibilität mit den älteren Generationen von Telefonkarten, deren enthaltene Einheiten nach wie vor gültig sind.

Wird eine Buchungskarte, also eine echte Chipkarte, verwendet, dann kann das Terminal die Taktfrequenz von 1,2 MHz bis 9,8 MHz wählen. Bei synchronen Karten ist es notwendig, den Takt in den Bereich von 20 kHz abzusenken, um eine funktionsfähige Kommunikation aufzubauen.

Die Terminals haben zum Schutz vor Dummy-Karten einen sehr massiv aufgebauten Shutter. Dieser würde mit einem Schlagmesser gegebenenfalls alle aus dem Einsteckschlitz herausführenden Kabel abtrennen. Damit verhindert man das Abhören und die Manipulation während der Kommunikation zwischen Terminal und Karte.

Die für die Telefonkarte eingesetzten Chiptypen haben sowohl ein vom Halbleiterhersteller maskenprogrammierbares ROM als auch EEPROM-Speicher für kartenindividuelle Daten und den Guthabenzähler.

Die Erzeugung der Programmierspannung für das EEPROM übernimmt eine auf dem Chip befindliche Ladungspumpe, sodass es nicht mehr notwendig ist, sie extern anzulegen. Somit genügt für die Versorgung des Chips das Anlegen der Betriebsspannung. Zum Schutz vor Betrug mit Nachahmungen hat der Chip ein Hardware-Sicherheitsmerkmal, dessen Funktion geheim ist. In Zukunft werden auch Speicherchips eingesetzt, die eine einseitige Authentisierung der Karte durch die äußere Welt ermöglichen.

Die neueren Telefonchips haben meist nur noch sechs Kontakte. Der Grund liegt darin, dass für die volle Funktionalität des Speicherchips nur fünf notwendig und somit auch alle weiteren Kontakte nicht belegt sind. Die Verwendung von acht Kontakten erhöht sogar noch die Fertigungskosten, da das Modul größer und somit teurer ist. Weiterhin dauert das Fräsen der Aussparung für das größere Modul länger, was wiederum den Durchsatz in der Maschine senkt und die Stückkosten erhöht. Deshalb haben praktisch alle neuen Telefonkarten nur mehr sechs Kontaktflächen. Die sechs oder acht auf die Kontaktfläche herausgeführten Leitungen des Chips haben die folgende Belegung:

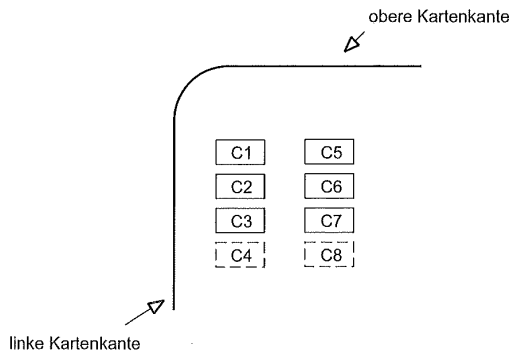


Bild 13.28 Die Belegung der Kontakte einer Telefonkarte. Die nicht aufgeführten Kontakte sind nicht belegt.

- C1 Betriebsspannung ($5\text{ V} \pm 5\%$)
- C2 Steuereingang
- C3 Takteingang (ca. 20 kHz)
- C5 Masse
- C7 Datenübertragung

Die Guthabekarte der Telekom enthält im Speicher des Chips die folgenden Daten:

Seriennummer

Die Seriennummer der Telefonkarten ist 7-stellig. Telefonkarten können deshalb bei Verdacht auf Betrug einzeln gesperrt werden. Gespeichert ist die Seriennummer in einem Teil des EEPROMs, das gegen Überschreiben oder Löschen gesperrt ist.

Herstellungsdatum

Während der Produktion der Chips werden in einen EEPROM-Speicherbereich sowohl das Herstellungsjahr wie auch der Herstellungsmonat irreversibel geschrieben. Dieses Datum bezieht sich nur auf den implantierten Chip, nicht auf die Karte selber.

Hersteller

Nachdem der Chip in den Kartenkörper implantiert ist, schreibt der Kartenhersteller einen individuellen Code in den Speicher der Karte. Dieser dient als Herstellermerkmal und kann ebenfalls nicht mehr geändert werden.

Anfangswert

In jeder Telefonkarte ist zusätzlich zum aktuellen Guthaben der Anfangswert der Karte gespeichert.

Restguthaben

Dies ist das einzige Datenelement in der Karte, das für die Funktionalität „vorbezahlte Telefonkarte“ funktionell notwendig ist. In einem fünfstufigen irreversiblen Oktalzähler im EEPROM werden vom Kartentelefon die einzelnen Einheiten nacheinander abgebucht. Der

Zähler besteht somit aus 5 Byte mit je 8 Bit, was einen maximalen Zählbereich von $8^5 = 32768$ ergibt. Während der Kartenfertigung setzt man den Zähler auf den gewünschten Wert herab, sodass die Karte den entsprechenden Zählerstand gemessen in Pfennigen aufweist. Danach kann während der Benutzung der Zähler weiter gegen null dekrementiert werden. Hat der Zähler irgendwann den Wert null erreicht, ist die Karte komplett abtelefoniert.

Transaktionsablauf

Steckt man nun eine Telefonkarte in ein Kartentelefon, schließt als Erstes der Shutter den Kartenschlitz, und die Kontakte des Terminals setzen sich parallel dazu auf die korrespondierenden Kontaktflächen der Karte. Danach fährt das Kartentelefon eine ISO-Anschaltsequenz und aktiviert so die Karte. Anschließend setzt das Terminal den Adresszeiger in der Speicherkarte auf null und liest die ersten 16 Bit aus. Verläuft dies positiv, dann ist die Telefonkarte funktionsfähig. Diese 16 Bit werden vom Terminal als eine Art ATR interpretiert, in dem diverse Betriebsparameter für die Karte enthalten sind. Das Terminal prüft nun anhand dieser Parameter, ob die eingesteckte Karte zu den erlaubten Kartenarten gehört und somit überhaupt in allen seinen Funktionen korrekt angesteuert werden kann. Die ebenfalls ausgelesene Kartennummer wird parallel dazu über das DOV-Modem in die nächste übergeordnete Ebene gesendet, wo sie mit einer Sperrliste verglichen wird. Ist dort kein Eintrag vorhanden, darf die Karte benutzt werden, und das Kartentelefon erhält eine entsprechende Nachricht.

Nun wird der Adresszeiger auf den Bereich des Gebührenzählers gesetzt, der noch vorhandene Guthabenbetrag aus dem EEPROM-Speicher ausgelesen und auf dem Display angezeigt. Nach Wahl der Telefonnummer durch den Benutzer und einem erfolgreichen Verbindungsaufbau erhält das Kartentelefon laufend über das DOV-Modem die abzubuchenden Gebühreneinheiten. Das Kartentelefon dekrementiert daraufhin jeweils den Zähler in der Speicherkarte und liest ihn danach sofort wieder aus. Auf diese Weise kann es kontrollieren, ob der Betrag auch korrekt abgebucht wurde. Ist dies nicht der Fall, wird die Verbindung sofort unterbrochen. Errechnet das Kartentelefon, dass das Guthaben in der Speicherkarte innerhalb der nächsten 20 Sekunden zu Ende geht, ertönt ein Signalton, und die Karte kann ohne Gesprächsabbruch gewechselt werden.²⁰

²⁰ Entwertungszyklus einer Speicherkarte, siehe auch Abschnitt 12.2 „Vorbezahlte Speicherkarten“.

14 Beispielhafte Anwendungen

14.1	Kontaktlose Speicherkarte für Flugverkehr	819
14.2	Krankenversichertenkarte	822
14.3	Elektronische Mautsysteme	827
14.4	Digitale Signatur	831
14.5	Signaturanwendung nach PKCS #15	841
14.6	Personalausweis FINEID	848
14.7	Tachosmart	848

Auf der Grundlage der vorangegangenen Kapitel, in denen alle relevanten Aspekte der Chipkarten-Technologie behandelt wurden, sind hier beispielhaft Anwendungen aufgeführt und in groben Zügen erklärt. An ihnen sieht man den Umfang und den Komplexitätsgrad großer Chipkarten-Anwendungen.

Ein Zweck dieses Kapitel ist es auch, Systeme darzustellen, in denen Chipkarten eingesetzt werden und in denen sie nur ein Bestandteil von vielen sind. Funktionalität, Benutzerfreundlichkeit und üblicherweise fast die gesamte Sicherheit dieser Systeme hängt an den verwendeten Chipkarten, doch ist es empfehlenswert, diese Systeme immer ganzheitlich zu betrachten, da sie nur zufrieden stellend funktionieren, wenn alle Komponenten reibungslos zusammenarbeiten.

14.1 Kontaktlose Speicherkarte für Flugverkehr

Das im Folgendem vorgestellte System unterscheidet sich aus mehreren Gründen von den üblichen Anwendungen von Chipkarten. Dies betrifft die zugrunde liegende Systemarchitektur sowie die Datenübertragung und Spannungsversorgung der Karte. Dabei handelt es sich um das System für „Fliegen ohne Ticket“ der deutschen Fluggesellschaft Lufthansa. Es basiert auf Speicherkarten, die kontaktlos arbeiten und somit nicht mehr in ein Terminal eingeführt werden müssen.¹ Der dahinterstehende Systemgedanke unterscheidet sich ebenfalls von allen hier dargestellten Chipkarten-Anwendungen. Man benutzt die Chipkarte nur als Identifikationsmittel für den Benutzer, alle Daten für die Anwendungen sind im Hintergrundsystem abgelegt.

Schon seit langem gibt Lufthansa Karten für Vielflieger und für ihr Bonussystem heraus. Diese Karten besaßen ursprünglich Hochprägung sowie einen Magnetstreifen für die automatische Bearbeitung. Zusätzlich konnten bestimmte Kartentypen noch mit einem Chip für die öffentlichen deutschen Kartentelefone und mit einer Kreditkartenfunktion ausgerüstet werden. Diese bestehende Familie von Karten sollte aufwärts kompatibel durch zusätzliche Anwendungen für den Bereich Boarding und Ticket ergänzt werden. Zusätzlich zur geforderten Kompatibilität war ein zweites Ziel, dass an den bestehenden Systemen so wenig wie möglich geändert werden durfte. Eine einfache Bedienbarkeit durch den Kunden musste ebenfalls möglich sein.

¹ Siehe auch Abschnitt 2.3.3 „Kontaktlose Chipkarten“.

In der maximalen Ausbaustufe ist das Ergebnis eine Multiapplication-Karte, mit Magnetstreifen, Hochprägung, Hologramm, Speicherchip mit Kontakten und einem kontaktlosen Speicherchip. Mit dieser Karte kann eine große Reihe unterschiedlicher Anwendungen bedient werden, ohne dass es mit den früher ausgegebenen Karten zu Problemen hinsichtlich der Kompatibilität kommt.

Vor der geplanten firmenweiten Einführung wurde ein Feldversuch auf der Strecke Frankfurt a. M. und Berlin durchgeführt. Man teilte 600 völlig unterschiedlichen Kunden die neuen Karten zu; zwischen Mai und Dezember 1995 wurden mehrere tausend Flüge auf dieser Strecke durchgeführt. Für die Benutzung der Karte stellte man den Reisenden entsprechende Automaten auf beiden Flughäfen zur Verfügung. Diese auf PCs basierten Geräte hatten einen berührungssensitiven Farbbildschirm sowie einen Drucker. An die PC-Steuerung der Automaten war außerdem ein Sender und Empfänger für die kontaktlose Chipkarte angeschlossen.

Anwendungen auf der Karte

Dem Kunden stehen auf der neuen Karte die unterschiedlichsten Anwendungen zur Verfügung. Hier und auch im Folgenden wird immer von der maximal möglichen Ausstattung der Karte ausgegangen. Es gibt natürlich auch einfachere Varianten, so beispielsweise eine Karte, die nur die Funktionen des kontaktlosen Speicherchips enthält. Mit diesem Chip ist ein selbstbedientes Einchecken am Automaten möglich, was natürlich zu einem entsprechend großen Durchsatz und zu weniger Wartezeiten bzw. schnellerer Abfertigung führt. Das Lufthansa Bonussystem ist im Rahmen des kontaktlosen Chips ebenfalls integriert, sodass keine zusätzliche Karte oder Dateneingabe notwendig ist.

An entsprechenden Automaten am Flughafen kann sich der Kunde einchecken und eine so genannte „Boarding Information“ ausdrucken lassen. Auf dieser sind, analog zum aktuellen Ticket, alle wesentlichen Informationen für den gebuchten Flug vorhanden. Gegebenenfalls, kann am Automaten der Sitzplatz im Flugzeug gewählt werden. Unter Angabe der auf die Karte geprägten Nummer können Flüge auch telefonisch gebucht werden. Es ist geplant, dies in Zukunft vollständig ohne Tickets aus Papier durchzuführen, da alle relevanten Informationen über die Chipkarte aus dem Hintergrundsystem abgerufen werden können. Dies schließt natürlich Flugbestätigungen per Telefon oder Fax nicht aus, die auch weiterhin wie bisher möglich sind.

Zusätzlich zu diesen Funktionen können auf die neue Karte noch die entsprechenden Daten auf den Magnetstreifen und das Hologramm für eine Kreditkarte aufgebracht werden. Auch besteht die Möglichkeit, einen Chip mit Kontakten an der ISO-Position zu implantieren, was man momentan dazu benutzt, um dort einen Telefonchip anzubringen.

Das Gesamtsystem

Alle Anwendungen, die auf dem kontaktlosen Speicherchip beruhen, sind seitens der Chipkarte sehr einfach aufgebaut. Die Karte wird nur zur Identifizierung benutzt, wobei man hier einen Speicherchip zum Einsatz bringt, der eine dynamische Authentisierung durch das Hintergrundsystem ermöglicht.² Nach der Authentisierung liest man alle Daten aus der Karte. Dies sind zurzeit Kundennummer, Kundenname und Kundenprofil. Mit diesen Daten

² Siehe auch Abschnitt 4.11.1 „Einseitige symmetrische Authentisierung“.

kann nun das Hintergrundsystem die Karte einer Person zuordnen, und Buchungsdaten, Guthaben im Bonussystem und alle anderen Anwendungen stehen zur Verfügung. Die Karte wird also nur als eine Art Schlüssel verwendet, die zu den verschiedenen Anwendungen gehörigen Daten und Mechanismen befinden sich stets im Hintergrundsystem. Dies hat natürlich in diesem Fall erhebliche Vorteile, da das Hintergrundsystem inklusive aller notwendigen Datenbanken, Programme und Schnittstellen schon längst vorhanden ist.

Ein weiterer Vorteil dieses Systems ist die Vorgehensweise bei verlorenen oder funktionsunfähigen Karten, eine Problematik, die in fast allen anderen Systemen mit Multiapplication-Karten bisher immer ausgespart wurde. Bei Anwendungen, die nach der Personalisierung in die Karte eingebracht wurden, muss der Karteninhaber mit seiner neu ausgestellten Karte alle Anwendungsbetreiber kontaktieren, damit diese ihre Anwendungen wieder auf seine Karte aufbringen. Durch den zentralistischen Systemaufbau gibt es diese Art von Problemen bei der Lufthansa-Chipkarte nicht. Der Kunde erhält bei Verlust bzw. Funktionsunfähigkeit eine neue Chipkarte, und die Alte wird anhand ihrer Nummer systemweit gesperrt. Dies stellt auch logistisch keinen großen Aufwand dar, da alle angeflogenen Flughäfen über das schon seit langem vorhandene Netzwerk von Lufthansa Zugriff auf die notwendigen Daten haben.

Die kontaktlose Karte

Die Chipkarte besitzt das international übliche ID-1 Format. Der innerhalb des Kartenkörpers eingebaute Speicherchip benutzt eine induktive Kopplung mit einer Spule für die Energie- und Datenübertragung. Mit dieser Technologie können in einer Entfernung von bis zu 10 cm vom Terminal Daten sowohl gelesen als auch geschrieben werden.³ Befindet sich die Chipkarte in einer normalen Geldbörse, ist es sogar möglich, den Datenaustausch mit dem Terminal durchzuführen, ohne die Karte aus der Geldbörse zu nehmen. Es reicht, die Geldbörse an das Terminal zu halten. Da sich Chip und Spule im Inneren der Karte befinden, lässt sich davon völlig unbeeinträchtigt das grafische Layout der Karte definieren. Außerdem kommt es aufgrund der kontaktlosen Technik natürlich zu keinen Abnutzungen an den Kontaktelementen des Chips, da diese schlichtweg nicht mehr vorhanden sind.

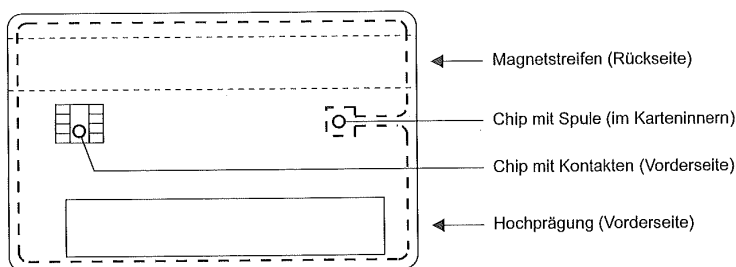


Bild 14.1 Physikalischer Aufbau der Chipkarte von Lufthansa in der maximalen Ausstattung

Die typische Transaktionszeit im vorgestellten System zwischen Terminal und Chipkarte beträgt ca. 100 ms und findet bei einer Frequenz von 13,56 MHz statt. Der verwen-

³ Siehe auch Abschnitt 3.6 „Kontaktlose Karten“.

dete Speicherchip (SLE 44R35) besitzt 1 kByte EEPROM und die Fähigkeit einer einseitigen dynamischen Authentisierung durch die äußere Welt. Da sich momentan nur 48 Byte Daten (Kundennummer, Kundenname, Kundenprofil) im Speicher befinden, können in Zukunft noch weitere Einsatzgebiete erschlossen werden, obwohl man bei diesem System für eine neue Anwendung bekanntermaßen keinen zusätzlichen Speicher in der Chipkarte benötigt.

Resümee

Der Feldversuch auf der Strecke Frankfurt a. M. – Berlin verlief sehr erfolgreich. Das System wurde jedoch in dieser Konstellation nicht weiter verfolgt.

Obwohl diese Chipkarten-Anwendung seitens des Systemaufbaus untypisch ist, bringt sie doch eine interessante neue Betrachtungsweise in die Chipkartenwelt. Da sich die Anwendungsdaten im Hintergrundsystem befinden, sind auch alle datenschutzrechtlichen Aspekte in dieses verlagert. Dies bedeutet, dass der Datenschutz, in welchem Land auch immer, keinen Einfluss mehr auf die Dateninhalte der Karte hat. Dadurch geht man auch den bekannten Speicherplatzproblemen bei mehreren Anwendungen auf einer Karte aus dem Weg. Das Problem, welcher Anwender welche Daten auf die Karte schreiben darf, besteht somit nicht mehr. Auch ist die Trennung der Anwendungen auf der Karte vollständig, es kann seitens des Systemdesigns keine Interferenzen zwischen den einzelnen Anwendungen geben. Zuletzt ist noch anzumerken, dass die für den Systembetreiber wertvollen Daten immer sicher in seinem Hintergrundsystem aufbewahrt sind und nicht auf der Chipkarte, auf der sie verloren gehen können.

Das neue System stellt außerdem ein Musterbeispiel für den nahtlosen Übergang von einer Kartentechnologie zur nächsten dar. Diese „sanfte“ Migration von einer Technologiestufe zur nächsten ist ein sehr gewichtiges Argument, da getätigte Investitionen nicht verloren sind und ein System nicht komplett neu aufgebaut werden muss.

14.2 Krankenversichertenkarte

In Deutschland haben bis Ende 1994 alle Mitglieder einer gesetzlichen Krankenkasse oder einer Ersatzkasse eine Krankenversichertenkarte (KVK) bekommen. Ab 1996 haben auch die privaten Krankenversicherungen damit begonnen, eigene und zu den gesetzlichen KV-Karten kompatible Chipkarten auszugeben. Somit weisen diese Karten mit über 72 Millionen Stück noch vor den Telefonkarten die höchste Durchdringung in der Gesamtbevölkerung Deutschlands auf.

Ursprünglich war nur die Einführung einer Magnetstreifenkarte geplant, doch entschied man sich in Hinblick auf künftige Weiterentwicklungen, Karten mit Chips zu verwenden. Die kostengünstigste Lösung waren Speicherkarten, da Karten mit implantiertem Mikrocontroller teurer wären. Das System ist jedoch so ausgelegt, dass in weiteren Ausbaustufen in einigen Jahren Chipkarten die Speicherkarten ablösen können. So ist ein bundesweites System entstanden, das die Grundlage für eine spätere Gesundheits-Chipkarte bilden könnte, wenn dies erforderlich wäre.

Die Krankenversichertenkarte für den Versicherten hat zwei grundsätzliche Funktionen. Zum einen dient sie als Ausweis dem behandelnden Arzt gegenüber. Sie ist insofern ein

Ersatz für den Krankenschein aus Papier geworden. Die zweite Funktion ist ein maschinenlesbarer Datenträger für den Praxiscomputer des Arztes. Im Normalfall ist das Terminal an einen PC in der Arztpraxis angeschlossen, der es auch ansteuert. Die Karte kann in dem Terminal ausgelesen, und die so erhaltenen Abrechnungsdaten können maschinell weiterverarbeitet werden. Falls der Arzt seine Praxis mit herkömmlichen Organisationsmethoden, d. h. ohne Computer, verwaltet, kann das Terminal auch direkt die Daten der Krankenversichertenkarte auf einen Drucker und somit auf Formulare übertragen.

Drei verschiedene Stellen können auf die Krankenversichertenkarte zugreifen: erstens die Arztpraxis, in der die Daten der Karte nur gelesen werden können. Schreiben von Daten ist nicht vorgesehen und wird durch die Software der Terminals verhindert. Zweitens die Krankenkasse selber, in der ebenfalls nur lesend auf den Speicher der Karte zugegriffen werden kann. Dort ist es dem Versicherten möglich, seine persönlichen Daten, die in der Karte gespeichert sind, auszulesen und zu kontrollieren. An speziellen Terminals bei den Krankenkassen können auch Daten in die Karte geschrieben werden. Dies wäre zum Beispiel bei einem Wohnungswechsel möglich, um so die neue Adresse in die Karte einzutragen.

Allerdings sind viele Krankenkassen schon dazu übergegangen, stattdessen dem Versicherten einfach eine neue Karte mit der Aufforderung, die Alte zu vernichten, zuzusenden. Der Grund dafür liegt in der wesentlich einfacheren und somit billigeren Logistik.

In der Anfangsphase des KVK-Projektes standen die verschiedensten Informationen über den Patienten zur Diskussion. Von der Blutgruppe bis zu Allergien sollten Informationen auf die Karte, die auf diese Weise auch eine Art Notfallpass geworden wäre. Nachdem alle datenschutzrechtlichen Bedenken zerstreut waren, blieben jedoch nur mehr die folgenden personenbezogenen Informationen im Chip der Karte.

Die in der Karte enthaltenen Daten sind im Wesentlichen auch außen auf dem Kartenkörper zu finden, sodass der Versicherte seine Daten genau kennt – jedoch nur insoweit, als sie unveränderlich und personenbezogen sind. Die Adresse befindet sich nur im Speicher der Karte, sodass bei einem Umzug theoretisch keine neue Karte ausgestellt werden müsste.

Die Kenntnis der in der Karte enthaltenen Daten war auch eine der Voraussetzungen für die Zulassung des gesamten Systems. Es dürfen in der Karte keine geheimen oder dem Versicherten nicht bekannte Daten vorhanden sein. Auch darf keine Möglichkeit bestehen, zu einem späteren Zeitpunkt unautorisiert zusätzliche Daten in den Chip zu schreiben. Um ein Schreiben von Daten schon im Vorfeld zu verhindern, erhalten weder die Arztpraxen noch die Krankenkassen Terminals, die eine solche Fähigkeit aufweisen. Lediglich einige Verwaltungsterminals bei den Kassen können Daten in die Karte schreiben. Dazu benötigt man keinen speziellen Schlüssel zur Authentisierung, da die Daten von einem technisch entsprechend ausgerüsteten Terminal ohne weiteres geschrieben werden können.

Rein äußerlich verhält sich die Krankenversichertenkarte so als ob nur eine einzige transparente Datei (EF) vorhanden wäre. Auf diese Datei kann mit einem Offset und einer Längenangabe lesend wahlfrei zugegriffen werden. Bestimmte Verwaltungsterminals können auch in den Speicher schreiben, doch ist dies aus Datenschutzgründen die Ausnahme.

Bei der Festlegung der Anordnung der Datenelemente war es von großer Wichtigkeit, künftige Erweiterungen oder Änderungen ohne Kompatibilitätsprobleme zu ermöglichen. Aufgrund dieser Forderung wurden alle personenbezogenen Daten der Krankenversicher-

Tabelle 14.1 Die Datenelemente und ihre TLV-Codierung in der deutschen Krankenversichertenkarte nach der „Technischen Spezifikation der Versichertenkarte“ von 1993

Datenelement	Länge	Kennzeichen (<i>tag</i>)
Bundesland	1 – 3 Byte	'8A'
Familienname des Versicherten	2 – 28 Byte	'87'
Geburtsdatum des Versicherten (Format: TTMMJJJJ)	8 Byte	'88'
Gültigkeitsdatum der Karte (Format: MMJJ)	4 Byte	'8D'
Name der Krankenkasse	2 – 28 Byte	'80'
Namenszusatz des Versicherten	1 – 15 Byte	'86'
Nummer der Krankenkasse	7 Byte	'81'
Nummer des Versicherten	6 – 12 Byte	'82'
Ortsname	2 – 22 Byte	'8C'
Postleitzahl	4 – 7 Byte	'8B'
Prüfsumme (XOR) über das gesamte Template	1 Byte	'8E'
Rechtskreis Ost/West	1 Byte	'90'
Status des Versicherten	4 Byte	'83'
Straßenname und Hausnummer	2 – 28 Byte	'89'
Titel des Versicherten	2 – 15 Byte	'84'
Template der Versichertendaten	70 – 212 Byte	'60'
Versichertenkartennummer (VKNR)	5 Byte	'8F'
Vorname des Versicherten	1 – 28 Byte	'85'

tenkarte in der Datenbeschreibungssprache ASN.1 strukturiert. Sie sind im Speicher der Karte in einer TLV-Struktur enthalten. Somit können in Zukunft weitere Datenobjekte hinzugefügt oder die Codierung vorhandener Daten geändert werden. Die zu verwendenden Kennzeichen (*tags*) schreiben eine Spezifikation vor, sodass die Datenelemente aller Krankenversichertenkarten auf der gleichen Grundlage strukturiert sind.

Die Krankenversichertenkarte ist keine Mikroprozessorkarte. Sie ist eine Speicherkarte, wie sie mit ähnlicher Hardware auch im Telefonkartenbereich seit Jahren eingesetzt ist. Der dabei verwendete EEPROM-Speicher muss eine Größe von minimal 256 Byte besitzen. Dies ist die Länge aller notwendigen Daten, die sich in der Krankenversichertenkarte befinden. Somit passen die Daten exakt in den Speicher und es ist physikalisch nicht möglich, zusätzliche Informationen unterzubringen, was den datenschutzrechtlichen Bestimmungen widersprechen würde.

Die zum Takt synchronen Übertragungsprotokolle sind vom jeweils eingesetzten Chip-typ abhängig. Jedes Terminal muss somit alle möglichen Protokolle komplett ausführen können. Der Kartenkörper kann in einem Spritzgussverfahren oder in Mehrlagentechnik gefertigt sein. Die Lebensdauer der Krankenversichertenkarte wurde auf 6 Jahre festgelegt. Nach Ablauf dieses Zeitraums erhält der Versicherte automatisch eine neue Karte. Dies führt dazu, dass pro Jahr ca. 15 Millionen neue Karten ausgegeben werden.

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0y	'A2'	'13'	'10'	'91'	'46'	'0B'	'81'	'15'	'44'	'45'	'47'	'2B'	'44'	'32'	'88'	'F1'
	ATR und Produktionsdaten															
1y	'9F'	'61'	'0B'	'4F'	'06'	'D2'	'76'	'00'	'00'	'01'	'01'	'53'	'01'	'07'	'60'	'81'
				T	L							T	L	V1	T	L1
2y	'88'	'80'	'18'	'56'	'65'	'72'	'65'	'69'	'6E'	'74'	'65'	'20'	'4B'	'56'	'20'	'20'
	L2	T	L	"V"	"e"	"r"	"e"	"i"	"n"	"t"	"e"	" "	"K"	"V"	" "	" "
3y	'20'	'20'	'20'	'20'	'34'	'30'	'33'	'33'	'35'	'33'	'31'	'81'	'07'	'30'	'30'	'30'
	" "	" "	" "	" "	"4"	"0"	"3"	"3"	"5"	"3"	"1"	T	L	"0"	"0"	"0"
4y	'30'	'30'	'30'	'30'	'8F'	'05'	'30'	'30'	'30'	'30'	'30'	'82'	'07'	'38'	'31'	'36'
	"0"	"0"	"0"	"0"	T	L	"0"	"0"	"0"	"0"	"0"	T	L	"8"	"1"	"6"
5y	'34'	'35'	'30'	'31'	'83'	'04'	'30'	'30'	'30'	'32'	'90'	'01'	'31'	'85'	'09'	'41'
	"4"	"5"	"0"	"1"	T	L	"0"	"0"	"0"	"2"	T	L	"1"	T	L	"A"
6y	'6C'	'65'	'78'	'61'	'6E'	'64'	'65'	'72'	'87'	'05'	'52'	'61'	'6E'	'6B'	'6C'	'88'
	"1"	"e"	"x"	"a"	"n"	"d"	"e"	"r"	T	L	"R"	"a"	"n"	"k"	"l"	T
7y	'08'	'30'	'31'	'30'	'31'	'31'	'39'	'36'	'35'	'89'	'15'	'50'	'72'	'69'	'6E'	'7A'
	L	"0"	"1"	"0"	"1"	"1"	"9"	"6"	"5"	T	L	"P"	"r"	"i"	"n"	"z"
8y	'72'	'65'	'67'	'65'	'6E'	'74'	'65'	'6E'	'73'	'74'	'72'	'2E'	'20'	'32'	'30'	'30'
	"r"	"e"	"g"	"e"	"n"	"t"	"e"	"n"	"s"	"t"	"r"	" "	" "	"2"	"0"	"0"
9y	'8B'	'05'	'38'	'30'	'30'	'30'	'30'	'8C'	'07'	'4D'	'7D'	'6E'	'63'	'68'	'65'	'6E'
	T	L	"8"	"0"	"0"	"0"	"0"	T	L	"M"	"ü"	"n"	"c"	"h"	"e"	"n"
Ay	'8D'	'04'	'30'	'31'	'30'	'37'	'8E'	'01'	'42'	'C0'	'54'	'20'	'20'	'20'	'20'	'20'
	T	L	"0"	"1"	"0"	"7"	T	L	XOR	T	L					
By	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'
	nicht genutzter Speicherbereich, gefüllt mit Leerzeichen															
Cy	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'
Dy	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'
Ey	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'
Fy	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'20'	'00'
	NU															

Bild 14.2 Beispielhafter Datensatz einer KV-Karte für Privatversicherte mit Decodierung aller enthaltenen Datenelemente. Die aufgeführten Daten gehören einer nicht existenten Person, und die XOR-Prüfsumme der Karte ist bewusst falsch. Die folgenden Abkürzungen wurden benutzt: T – Kennzeichen; L, L2 – Länge der nachfolgenden Daten; L1 – Kennzeichen innerhalb der Längencodierung, das folgende Byte L2 ist die Länge der nachfolgenden Daten, NU – nicht benutzt (*not used*) und V1 – Kennzeichen des Personalisierers.

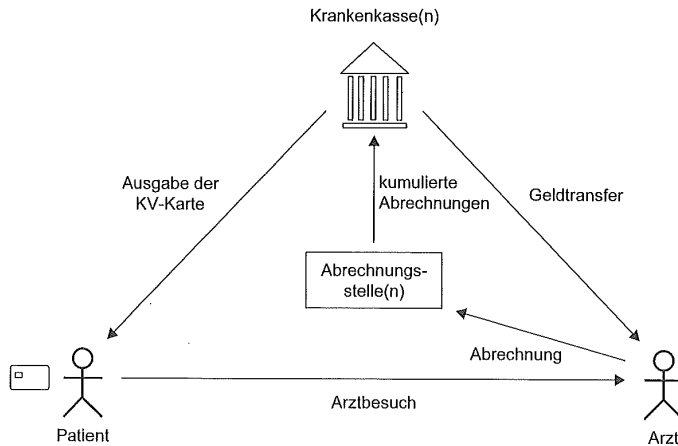


Bild 14.3 Grundlegender Systemaufbau bei der deutschen Krankenversichertenkarte.

Alle Terminals werden von einem optional angeschlossenen Rechner durch das T = 1 Übertragungsprotokoll nach ISO/IEC 7816-3 angesteuert. Dabei gibt es die Einschränkung, dass im Protokoll keine Datenverkettung (*chaining*) verwendet werden darf. Dies brächte der Anwendung keine zusätzliche Funktionalität, sondern nur einen erhöhten Speicherbedarf im Terminal. Im Übrigen ist dies ein typisches Beispiel dafür, dass in realen Anwendungen oft nur die notwendigen Teile von Normen verwendet werden und selten alle geforderten Funktionen einer Norm implementiert sind.

Es gibt nur drei mögliche Kommandos, die das Terminal ausführen kann: Das Erste ist ein Reset der Krankenversichertenkarte mit anschließendem Empfang oder Auslesen des ATRs. Er wird immer am Anfang einer Sitzung zum Anschalten der Krankenversichertenkarte verwendet. Das zweite Kommando ist ein READ BINARY in ISO-Codierung, mit dem über das Terminal selektiv oder gesammelt alle Daten ausgelesen werden. Als drittes Kommando ist ein WRITE BINARY ebenfalls nach ISO/IEC 7816-4, allerdings nur an Verwaltungsterminals verfügbar. Bei allen anderen Terminals ist dieses Kommando gesperrt. Bei den KV-Karten der privaten Krankenversicherungen verfügen die Karten über einen Schreibschutz, welcher über nur den Krankenversicherungen bekannte PINs realisiert ist. Die Karten der gesetzlichen Krankenversicherungen können bei Kenntnis der nötigen Kommandos frei geschrieben werden.

Somit ist die Funktion des Terminals im Falle einer direkten Computeranbindung im Wesentlichen nur eine Protokollumsetzung von T = 1 zu den von der Hardware abhängigen synchronen Protokollen der Krankenversichertenkarte. Man sieht hier jedoch deutlich, dass mit überschaubarem Aufwand eine Umstellung auf Mikroprozessorkarten möglich wäre. Das Terminal muss im Fall von Chipkarten lediglich die vom Praxiscomputer erhaltenen Daten transparent weiterreichen. Die daraufhin von der Chipkarte erhaltene Antwort kann dann ebenfalls transparent, ohne Aufbereitung durch das Terminal, zum ansteuernden Computer gesendet werden.

14.3 Elektronische Mautsysteme

In manchen Ländern ist es üblich, für die Benutzung von bestimmten Straßen Maut zu verlangen. Gegenüber einer pauschalen Gebührenerhebung durch den Verkauf einer Vignette ist die Erhebung von Maut benutzungsabhängig, d. h. es ist entsprechend der Benutzungshäufigkeit und der Art des Fahrzeugs ein bestimmter Betrag zu entrichten. Bislang war es *usus*, an Mautstellen bar zu bezahlen. Vereinzelt kommen schon seit einigen Jahren elektronische Systeme mit diversen Arten von Karten in Mautsystemen (*road pricing*) zum Einsatz. Von Nachteil ist aber bei allen bisherigen Systemen, dass sie den Verkehrsfluss erheblich beeinträchtigen, da entweder angehalten oder die Geschwindigkeit bis auf Schritt-Tempo gedrosselt werden muss. Auch haben die entsprechenden Mautstationen einen großen Platzbedarf.

Angesichts dieser Nachteile hat das deutsche Bundesministerium für Verkehr 1993 beschlossen, einen groß angelegten Feldversuch zum Thema automatische Gebührenerfassung (AGE) zu starten. Ausgewählt wurde dabei die Bundesautobahn A555 zwischen Köln und Bonn. Auf dieser Strecke wurden von Mai 1994 bis Juni 1995 verschiedene Systeme von zehn unterschiedlichen Firmen getestet.

Dabei gab es einige nennenswerte Rahmenbedingungen, die als Grundlage für die getesteten Systeme dienten. Der Verkehr sollte normal und ohne Behinderung durch die Gebührenerfassung fließen können. Ein normaler Verkehrsfluss kann aber in Deutschland eine Geschwindigkeit von bis zu 250 km/h bedeuten. Es sollte also nicht möglich sein, sich durch zu schnelles Fahren einer Gebührenerhebung zu entziehen. Außerdem durften keine Mauthäuschen oder Auffangkörbe zur Bezahlung verwendet werden, da man nur Systeme wollte, deren Platzbedarf an der Autobahn sich auf die Installation an Straßen- oder Schilderbrücken beschränkte. Auch mussten die verwendeten Systeme nicht nur eine Fahrspur (*single lane*), sondern mehrere parallele Fahrspuren (*multi lane*) unterstützen, da man aus baulichen Gründen keine Auffächerung in einzelne Spuren wollte, da dies wiederum den Verkehrsfluss stark beeinträchtigt hätte.

Im Laufe des Projekts bildete sich noch eine Zusatzanforderung heraus, die ursprünglich in dieser Weise nicht vorgesehen war. Sie wurde in der Öffentlichkeit mehr und mehr zum K.o.-Kriterium für den gesamten Feldversuch. Es war die Forderung nach vollständiger Anonymität bei der Gebührenerfassung. Es darf damit keine Möglichkeit geben, Bewegungsprofile von Fahrzeugen zu erstellen oder die Fahrroute von bestimmten Fahrzeugen zu überwachen. Alle Systeme konnten eine Bezahlvariante vorweisen, bei der die Anonymität des Fahrzeugs aufrechterhalten wurde. Dies aber nur so lange, wie die Gebühren bezahlt wurden. Sobald aber eine Gebührenerhebung fehlschlug, wurde das Fahrzeug fotografisch erfasst und der Fahrzeughalter ermittelt. Dieser erhielt dann einen entsprechenden Bußgeldbescheid. Dies war natürlich nicht im Feldversuch der Fall, da dort nur mit „Spielgeld“ bezahlt wurde.

Fast alle Systeme zur automatischen Gebührenerhebung verwendeten als Träger der elektronischen Währungseinheiten eine Chipkarte. Dies ist auch der Grund, warum hier diese Thematik dargestellt ist, da sie in Zukunft durchaus Bedeutung im Chipkartenbereich bekommen kann.

Die Systeme besaßen ein im Fahrzeug montiertes Bordgerät, welches die Bezeichnung OBU (*on board unit*) oder IVU (*in vehicle unit*) hat, sowie u. U. weitere Zusatzgeräte. Diese Geräte, die über die Autobatterie mit Strom versorgt werden, besitzen im Fahrzeuginnern, an der OBU, ein Chipkartenterminal und ein Display mit einfacher Tastatur. Auch besteht eine Verbindung zur äußeren Welt, die je nach System uni- oder bidirektional sein kann. Die Verbindung wird meist durch Mikrowellen in dem von CEPT für diese Anwendung empfohlenen Frequenzbereich von 5,795 – 5,805 GHz aufgebaut. Alternativ dazu wurden zur Datenübertragung auch Funkwellen im 400 – 500 MHz-Bereich und IR-Übertragung eingesetzt. Letzteres hat natürlich den großen Nachteil, dass die Übertragung stark witterungsabhängig ist. Man denke nur an starken Schneefall oder Nebel. Bei den im Feldversuch beteiligten Systemen betrug der je nach Ausstattung unterschiedliche Abgabepreis der OBU in Großserienproduktion zwischen 50 und 150 Euro.

Die Kontrollstationen wurden entlang der Autobahn an Straßen- oder Schilderbrücken errichtet. Damit sind keine baulichen Eingriffe in die Straßendecke erforderlich. Die verwendeten Chipkarten enthielten keine aufwändigen elektronischen Geldbörsen, sondern nur sehr einfache, aber schnelle Abbuchungskommandos. Dies war einerseits darauf zurückzuführen, dass man kein echtes Geld verwendete, und andererseits auf die zur Verfügung stehende Transaktionszeit. Die Optimierung ging so weit, dass man zum Teil sogar den ATR der entsprechenden Chipkarten auf vier Bytes verkürzte, damit noch genügend Zeit zum eigentlichen Abbuchen blieb. Betrachtet man die Rahmenbedingung des uneingeschränkten Verkehrsflusses, wird dies schnell verständlich. Bei Tempo 250 km/h legt ein Fahrzeug in einer Sekunde 70 m zurück. Die Kontrollstationen hatten im 5,8 GHz-Frequenzbereich eine Kommunikationsreichweite von ≈ 5 m. Daraus resultiert eine minimale Verweildauer im Bereich der Gebührenerfassungsstelle von ≈ 70 ms. In dieser Zeit müssen seitens der Chipkarte folgende Abläufe stattfinden:

- Reset der Chipkarte und Aussenden des ATRs (≈ 10 ms)
- eine DES-Verschlüsselung zur Authentisierung der Chipkarte (≈ 12 ms)
- ein EEPROM-Schreibzugriff zur Speicherung des neuen Guthabens ($\approx 2 \cdot 3,5$ ms)
- Datenübertragung von und zur Chipkarte (≈ 30 ms)

Zusätzlich ist noch Zeit für die Datenübertragung zwischen OBU und der elektronischen Mautstation notwendig. Man kann sich ausrechnen, dass die zur Verfügung stehende Zeit sehr knapp bemessen ist. Die benutzten Chipkarten wurden grundsätzlich mit dem maximal zulässigen Takt (meist 5 MHz) versorgt. Die Übertragungsgeschwindigkeiten zwischen OBU und der Mautstation waren mit bis zu 1 MBit/s wesentlich höher als zwischen OBU und Chipkarte, sodass diese Übertragungszeit von geringem Einfluss ist.

Im Folgenden sind, stellvertretend für die insgesamt zehn am Feldversuch beteiligten Systeme, drei typische Vertreter dargestellt. Einige der eingesetzten waren bis auf kleinere technische Details nahezu identisch, deshalb sind die drei im Folgenden beschriebenen auch nicht namentlich aufgeführt. Sie sind sozusagen typische Vertreter ihrer Art.

System 1

Der nachfolgend vorgestellte Teilnehmer am Feldversuch hat seitens der Infrastruktur den klassischen Aufbau. Zwei kurz hintereinander über die Autobahn führende Fahrbahn- oder

Schilderbrücken wurden zur Montage der notwendigen Geräte benutzt. Das System ist natürlich so konzipiert, dass ein Fahrspurwechsel zwischen den beiden Brücken auf die Erfassung keinen Einfluss hat. Es gibt zwei verschiedene Arten der Bezahlung: Im Postpaid-Modus werden die aufgelaufenen Gebühren nach Art von Kreditkarten über ein Bankkonto eingezogen. Dabei ist die Anonymität des Fahrzeugs sehr schwierig zu gewährleisten. Im Prepaid-Modus werden vorausbezahlte Chipkarten verwendet, bei denen bei jeder Gebührenerfassung der entsprechende Betrag abgebucht wird.

Technisch gesehen, wird an der ersten Brücke die OBU mit der Chipkarte aktiviert. Anschließend erfolgt die Gebührenerhebung und eine grobe Vermessung des Fahrzeugs. Dadurch kann die Fahrzeugklasse, also LKW oder PKW, festgestellt werden. Da der Mautbetrag eine unterschiedliche Höhe für jede Fahrzeugklasse aufweist, ist es notwendig, eine Kontrolle der in der Chipkarte vermerkten Fahrzeugklasse durchzuführen. Dazu erstellt man beim Unterfahren der Brücke ein Höhenprofil des Fahrzeugs, welches für eine sichere Klassifizierung ausreicht. Innerhalb der Kommunikationszone an der unmittelbar folgenden zweiten Brücke wird dann wiederum eine Verbindung via OBU zur Chipkarte aufgebaut. Die elektronische Mautstation prüft dort, ob die initiierte Abbuchung erfolgreich durchgeführt werden konnte. War dies nicht der Fall, wird das Fahrzeug fotografiert. Anhand der so erfassten Nummernschilder lässt sich mittlerweile vollautomatisch eine eindeutige Identifizierung durchführen. Anschließend erhält der Fahrzeughalter ein entsprechendes Bußgeld.

Das System lässt sich noch beliebig mit unterschiedlichen Verfahren erweitern. So kann beim Bezahlen die Chipkarte noch mit einer Sperrliste verglichen werden, um dadurch gestohlene Karten von der Benutzung auszuschließen. Es wurde auch diskutiert, ob die Erfassungskameras in bestimmten Sonderfällen (z. B. Banküberfall mit Flucht im PKW) alle durchfahrenden Fahrzeuge erfassen sollen, um auf diese Weise Hinweise über Fluchtwege zu erhalten. Vorteile bietet dieses System durch die einfach aufgebauten und somit preisgünstigen OBUs. Allerdings müssen dem gegenüber die elektronischen Mautstationen umso aufwändiger sein.

System 2

Das Zweite hier vorgestellte System basiert nicht auf einer straßenseitig installierten Infrastruktur. Es verwendet im Prinzip virtuelle Mautstationen, die nur als Datenbasis in der OBU des Fahrzeugs sind. Diese Daten beinhalten alle Koordinaten der vorhandenen Mautstationen sowie die entsprechenden Preise für die zu befahrende Autobahnteilstrecke. Sobald nun ein Fahrzeug in ein solches Straßenstück einfährt, wird wiederum von einer Chipkarte der zu zahlende Betrag abgebucht. Dieser Vorgang ist selbstverständlich längst nicht so zeitkritisch wie die in System 1 beschriebene Abbuchung.

Die OBU kennt durch ein angeschlossenes GPS-Modul zu jedem Zeitpunkt die Koordinaten, an denen sich das Fahrzeug befindet. GPS heißt global positioning system und ist ein weltweites System zur Ermittlung der Ortsposition. Es wurde ab 1973 in den USA im Auftrag des Verteidigungsministeriums für militärische Zwecke entwickelt, war 1993 vollständig einsatzfähig und besteht aus 24 Satelliten, die in 20000 km Höhe codierte Radiosignale abstrahlen. Diese Signale enthalten die Sendezeit, die Satellitenposition und ein Satellitenkennzeichen. Mit entsprechenden Empfängern, welche nur noch die Größe einer Zigarettenschachtel haben, kann man die Signale im 1,6 GHz-Frequenzband empfangen

und die Ortsposition anhand der gesendeten Daten für zivile Anwendungen auf etwa 10 bis 20 Meter genau bestimmen. In einer verbesserten Version mit Differentialmessung (Differential-GPS) kann die Position auf einige wenige Meter ermittelt werden, wozu aber ein terrestrisch ausgestrahltes Zusatzsignal notwendig ist, welches momentan nur in Zentraleuropa empfangen werden kann. Für militärische Anwendungen ist die Genauigkeit der Positionsbestimmung allerdings weltweit im Bereich einiger Meter.

Natürlich ist auch bei diesem Mautsystem eine Überwachung notwendig. Diese wird aber stichprobenartig, ähnlich den heutigen Geschwindigkeitskontrollen, durchgeführt. Das System hat den großen Vorteil, dass keinerlei Installationen an den mautpflichtigen Straßen notwendig sind und die Mauterhebung an virtuellen Punkten stattfindet. Die Kontrolle ist aber schwieriger, und es können keine Chipkarten im Postpaid-Modus verwendet werden, da kein Datenaustausch zwischen OBU und der äußeren Welt stattfindet. Dies hat allerdings wiederum Vorteile hinsichtlich der Anonymität.

System 3

Das hier vorgestellte System 3 ist die technisch anspruchsvollste Lösung einer automatischen Gebührenerfassung für Fahrzeuge. Die Ausrüstung der OBU besteht aus einem GPS-Empfänger und einem GSM-Mobiltelefon. Mittels der GPS-Einheit wird die Position des Fahrzeugs festgestellt, und zur Übermittlung von abrechnungsspezifischen Daten benutzt die OBU das Mobiltelefon. Die OBU enthält wie die beiden obig beschriebenen Systeme wiederum ein Terminal mit Chipkarte, die zur Verrechnung der Gebühren herangezogen wird.

Die Mautstationen sind ähnlich dem vorgestellten System 2 nur virtuelle Punkte an Straßen, die anhand von Koordinaten an der Erdoberfläche festgelegt sind. Bei diesem System ist es dem Autobahnbetreiber auch möglich, durch die bidirektionale Kommunikation mit der OBU seine Tarife beliebig zu ändern. Diese können somit abhängig von Ort, Zeit, Fahrzeugklasse und Umweltbedingungen sein. Die Berechnung der anfallenden Gebühren wird in der OBU des Fahrzeugs durchgeführt.

Um nun eine gewisse Anonymität sicherzustellen, werden die von der in der OBU befindlichen Chipkarte abgebuchten Gebühren nicht unmittelbar an das Hintergrundsystem übermittelt, sondern bis zu einem bestimmten Schwellwert in der Chipkarte gesammelt. Ist diese Grenze erreicht, wird der Summenbetrag per Mobiltelefon zur Kontrolle bzw. Abrechnung übertragen. Aufgrund des zweiseitigen Datenaustausches ist es mit diesem System natürlich möglich, sowohl vorbezahlte elektronische Geldbörsen (*prepaid*) als auch Kreditkarten (*postpaid*) zu verwenden. Allerdings ist die Anonymität bei der nachträglichen Bezahlung nicht mehr zu gewährleisten. Bei vorbezahlten Chipkarten, die somit wie eine elektronische Geldbörse benutzt werden, kann der Autobahnbetreiber durch die Summenbildung der Beträge kein Bewegungsprofil des Fahrzeuges ermitteln. Unabhängig davon ist es natürlich dem GSM-Netzbetreiber möglich, über das ständig aktivierte Mobiltelefon laufend die Koordinaten des sich bewegendes Fahrzeugs zu ermitteln.

Auch bei diesem Systems ist es natürlich notwendig, eine Überwachung des fließenden Verkehrs zur Prüfung auf Schwarzfahrer vorzunehmen. Dies wird durch automatische Kameras an Brücken oder durch ortsveränderliche Prüfstellen durchgeführt. Die dazugehörigen Überwachungscomputer können via Mobiltelefon das Kennzeichen aus der OBU bzw. Chipkarte auslesen und mit dem am Fahrzeug befindlichen vergleichen. Sind beide

unterschiedlich oder konnte zur OBU keine Verbindung aufgebaut werden, so wird das entsprechende Fahrzeug fotografiert und ein Bußgeldverfahren eingeleitet.

Der Vorteil dieses System liegt vor allem darin, dass bis auf die Überwachung keine spezifische Infrastruktur aufgebaut werden muss. Allerdings ist dadurch der Preis der OBU sicherlich höher als bei den beiden vorangehend beschriebenen Systemen. Zwei unterschiedliche Bezahlungsarten (*prepaid/postpaid*) und eine vorhandene Datenübertragung zum Fahrzeug sind ebenfalls gute Argumente für dieses System.

14.4 Digitale Signatur

Die Anwendung von rechtsverbindlichen digitalen Signaturen erfordert zwei grundsätzliche Voraussetzungen: eine Mikrocontroller-Chipkarte mit einem leistungsfähigen numerischen Koprozessor und eindeutige gesetzlichen Rahmenbedingungen. Die Chipkarte wird zur sicheren Aufbewahrung der geheimen Signaturschlüssel und zur Erstellung der digitalen Signatur benutzt. Das Gesetz schafft für alle an dieser Chipkarten-Anwendung Beteiligten einen verbindlichen Rahmen.

Digitale Signaturen als abgeschlossene Lösung lassen sich natürlich auch völlig unabhängig vor einem gesetzlichen Hintergrund einsetzen. Um jedoch digitale Signaturen für den Unterschreibenden verbindlich zu machen, ist es in diesem Fall erforderlich, zwischen den beteiligten Parteien entsprechende Verträge abzuschließen. So ist beispielsweise die Verwendung von digitalen Signaturen im Business-to-business-Bereich schon seit einigen Jahren üblich. Wird jedoch eine offene Lösung angestrebt, die auch mit untereinander unbekannten Beteiligten funktioniert, sind entsprechende gesetzliche Rahmenbedingungen notwendig. Durch diesen gesetzlichen Rahmen kann die digitale Signatur einer regulären Unterschrift gleichgesetzt werden, und das Unterschriebene kann im Streitfall vor Gericht, ganz analog einer händischen Unterschrift, eingeklagt werden.

Die Anwendung von gesetzlich legitimierten und damit den händischen Unterschriften gleichgestellten digitalen Signaturen ist schier unbegrenzt. Im einfachsten Falle können elektronische Briefe und Bestellungen signiert werden, doch lässt sich im elektronischen Zahlungsverkehr beispielsweise auch die Einzugerlaubnis für eine Lastschrift unterzeichnen oder auch Überweisungsaufträge im Homebanking-Bereich. Den bedeutungsvollsten Einsatz von digitalen Signaturen stellen sicherlich Verträge aller Art dar.

Die normativen Rahmenbedingungen

Damit digitale Signaturen in einem offenen industriellen Umfeld interoperabel benutzt werden können, müssen die sicherheitstechnisch relevanten Komponenten verschiedene Normen erfüllen. So weit es die Signaturkarten betrifft, sind dies die internationalen Normen ISO/IEC 7816-4 für allgemeine Chipkarten-Kommandos und ISO/IEC 7816-8 für die Signatur-Kommandos. Für Kartenkörper, elektrische Eigenschaften, und Datenübertragung sind die entsprechenden Normen der ISO/IEC 7816-Reihe zu berücksichtigen. Die Authentisierungsverfahren zwischen Chipkarte und dem Rest der Welt deckt die ISO/IEC 9796-2 ab. Grundlegende Mechanismen und Vorgehensweisen für digitale Signaturen sind in der ISO/IEC 14 888 behandelt. Außerdem ist es üblich, dass die Zertifikate gemäß X.509 aufgebaut und codiert sind.

Mit einer digitalen Signaturkarte kann das Analogon einer echten Unterschrift erzeugt werden. Dies bedeutet, dass abhängig vom unterschriebenen Dokument unter Umständen

große Geld- oder Sachwerte im Spiel sein können. Deshalb müssen die sicherheitsrelevanten Komponenten im Umfeld einer digitalen Signaturanwendung evaluiert sein. Das dazu probate Mittel ist der Kriterienkatalog nach ITSEC oder Common Criteria (CC).

Die gesetzlichen Rahmenbedingungen in Deutschland

Damit digitale Signaturen als allgemein gültige, verbindliche und auch einklagbare Unterschriften anerkannt werden, ist es unumgänglich, die dazu notwendigen Rahmenbedingungen gesetzlich zu regeln. Der amerikanische Bundesstaat Utah hat 1995 als Erster ein diesbezügliches Gesetz verabschiedet. Dieses hatte für viele andere Länder Vorbildcharakter, so auch für das deutsche Signaturgesetz.

Die wesentlichen Forderungen des deutschen Signaturgesetzes sind im Folgendem aufgeführt:

- Der Betrieb einer Zertifizierungsstelle (d. h. eines Trustcenters) muss von der zuständigen Behörde genehmigt werden.
- Personen, die ein Zertifikat beantragen, müssen von der Zertifizierungsstelle zuverlässig identifiziert werden.
- Das Signaturschlüssel-Zertifikat darf anstelle des Namens auch ein Pseudonym des Inhabers beinhalten.
- Auf Ersuchen von bestimmten Behörden kann das Pseudonym des Signaturschlüssel-Inhabers aufgedeckt werden.
- Die Zertifizierungsstelle muss die Zertifikate für die öffentlichen Signaturschlüssel so zur Verfügung stellen, dass sie online nachprüfbar sind.
- Ein Zertifikat ist unter bestimmten Voraussetzungen, z. B. wenn der Signaturschlüssel-Inhaber dies verlangt, zu sperren.
- Die technischen Komponenten für Erzeugung und Überprüfung von digitalen Signaturen müssen die Daten, auf die sich die digitale Signatur bezieht, eindeutig darstellen.
- Durch entsprechende Vereinbarungen können auch digitale Signaturen aus anderen Ländern in Deutschland anerkannt werden.

Ein Signaturschlüssel-Zertifikat, üblicherweise nach X.509, muss nach dem Signaturgesetz die nachstehend aufgeführten Angaben enthalten:

- Namen oder Pseudonym des Signaturschlüsselinhabers.
- Den dem Signaturschlüsselinhaber zugeordneten öffentlichen Signaturschlüssel.
- Die Bezeichnung der für dieses Zertifikat benutzbaren Algorithmen (Hash-Algorithmus und Signatur-Algorithmus).
- Die laufende Nummer des Zertifikats.
- Beginn und Ende der Gültigkeit des Zertifikats.
- Den Namen der Zertifizierungsstelle.
- Angaben, ob der Signaturschlüssel nur für bestimmte Anwendungen benutzt werden darf.

Im deutschen Bundestag wurde am 13. Juni 1997 das „Gesetz zur Regelung der Rahmenbedingungen für Informations- und Kommunikationsdienste“ (IuKDG) beschlossen.

Es ist am 1. August 1997 in Kraft getreten und wurde in einer überarbeiteten Fassung am 22. Mai 2001 veröffentlicht. Dieses Gesetz enthält als Artikel 3 das Gesetz zur digitalen Signatur, in Kurzform Signaturgesetz (SigG) genannt. Der Artikel 3 teilt sich in 16 Paragraphen auf. Im §1, Abs. 1 ist der Zweck des Signaturgesetzes genannt. Er lautet: „Zweck des Gesetzes ist es, Rahmenbedingungen für digitale Signaturen zu schaffen, unter denen diese als sicher gelten und Fälschungen digitaler Signaturen oder Verfälschungen von signierten Daten zuverlässig festgestellt werden können.“ Diese Aussage zeigt bereits deutlich, dass keine spezielle technische Lösung festgelegt wurde, sondern nur die zur Anwendung von digitalen Signaturen notwendigen Rahmenbedingungen. Dies führt natürlich dazu, dass die gesetzlichen Anforderungen auf relativ abstraktem Niveau beschrieben sind.

Die Regelung und Beschreibung von digitalen Signaturanwendungen kann nicht alleine in einem Gesetz geregelt sein. Dazu ist eine Hierarchie von Anforderungen notwendig, deren oberste Instanz die gesetzliche Regelung – das Signaturgesetz – ist. Dieser folgen die dazugehörigen Ausführungsbestimmungen, in Deutschland Signaturverordnung (SigV) genannt. Auf der nachfolgenden Ebene, die technisch bereits sehr konkret die technischen Anforderungen zur digitalen Signatur beschreibt, befinden sich die Maßnahmenkataloge. Sie werden in Deutschland von der Regulierungsbehörde für Telekommunikation und Post herausgegeben. Diese drei Hierarchieebenen sind für alle Anwendungen von digitaler Signatur in Deutschland verbindlich, sofern die digitalen Signaturen einer regulären Unterschrift entsprechen sollen.

Auf der den Maßnahmenkatalogen nachfolgenden Ebene definieren die technischen Spezifikationen in eindeutig und in nicht interpretierbarer Form die konkreten Lösungen. Diese Spezifikationen sind abhängig vom jeweiligen Anwendungsbetreiber und müssen sich an den drei übergeordneten Hierarchieebenen orientieren. Benötigt jedoch ein Anwendungsbetreiber nicht den gesetzlichen Charakter der digitalen Signatur, so kann er selbstverständlich sein System aufbauen, wie es für seine Zwecke optimal erscheint. Man spricht dann von nicht gesetzeskonformen Lösungen, im Gegensatz zu den gesetzeskonformen Lösungen.

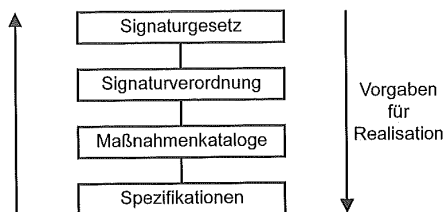


Bild 14.4 Der hierarchische Aufbau der Vorgaben für die Anwendung „digitale Signatur“ in Deutschland. Das System wurde top-down gestaltet, deshalb basieren die weiter unten stehenden Dokumente auf den jeweils oberhalb angeführten.

Die Verordnung zur digitalen Signatur (Signaturverordnung – SigV) beinhaltet die Ausführungsbestimmungen zum Signaturgesetz. Sie wurde von der Bundesregierung am 8. 10. 1997 beschlossen und ist zum 1. 11. 1997 in Kraft getreten. Sie enthält zu jedem der 16 Paragraphen des Signaturgesetzes detaillierte Beschreibungen, deren Abstraktionsgrad

deutlich geringer ist als derjenige des Gesetzes. Unter anderem sind in der Signaturverordnung folgende Aussagen enthalten:

- Der Antragsteller auf ein Zertifikat muss anhand seines Personalausweises, Reisepasses oder auf andere geeignete Weise identifiziert werden.
- Der Datenträger (in der Regel eine Chipkarte mit Signaturschlüssel) ist in persönlichem Gewahrsam zu halten, und die dazugehörigen Identifikationsdaten (z. B. PIN) sind geheim zu halten.
- Bei der Überprüfung von digitalen Signaturen muss überprüft werden, ob das Signaturschlüsselzertifikat zum Zeitpunkt der Signaturerstellung gültig war oder Beschränkungen der Benutzung enthält.
- Die Gültigkeitsdauer eines Zertifikats beträgt maximal fünf Jahre.
- Die Sperrung eines Zertifikats darf nicht mehr rückgängig gemacht werden.
- Es ist ein Maßnahmenkatalog mit geeigneten Sicherheitsmaßnahmen zu erstellen.
- Die Zertifizierungsstelle (d. h. das Trustcenter) muss von der zuständigen Behörde geprüft werden.
- Die Signaturschlüssel sind so zu erzeugen, dass ein Schlüssel mit an Sicherheit grenzender Wahrscheinlichkeit nur ein einziges Mal vorkommt.
- Der Signaturschlüssel darf nicht dupliziert werden können.
- Alle sicherheitstechnischen Veränderungen an technischen Komponenten müssen für den jeweiligen Benutzer erkennbar sein.
- Der Signaturschlüssel darf erst nach Identifikation des Inhabers durch Besitz und Wissen angewendet werden können.
- Bei der Anwendung des Signaturschlüssels darf dieser nicht preisgegeben werden.
- Im Bundesanzeiger wird eine Liste veröffentlicht, die die geeigneten Algorithmen zur Erzeugung von Signaturschlüsseln, zur Hash-Berechnung über die zu signierenden Daten und zur Erzeugung und Prüfung von digitalen Signaturen angibt. Zusätzlich wird ein Zeitrahmen angegeben, wie lange sie verwendet werden dürfen.
- Sollen mit einer Signatur versehene Daten länger gültig bleiben, als dies die zugelassenen Algorithmen erlauben, so sind diese Daten vor dem Ablauf des Algorithmus mit einer neuen digitalen Signatur inklusive Zeitstempel zu versehen.
- Die wesentlichen Komponenten einer signaturgesetzkonformen Anwendung sind nach ITSEC mit der Mechanismenstärke „hoch“ zu prüfen. Die Prüfstufe ist dabei von der jeweiligen Komponente und ihrem Einsatz abhängig. Die Tabelle 14.2 zeigt die Komponenten in Abhängigkeit von der Prüfstufe.

Die Maßnahmenkataloge zum Signaturgesetz und zur Signaturverordnung fixieren die technischen Anforderungen an dieses System und sind in ihrem Inhalt deutlich konkreter als das Signaturgesetz. Die technischen Spezifikationen basieren wiederum auf den Maßnahmenkatalogen und legen im Detail fest, wie die technische Realisierung zu erfolgen hat. Die in Deutschland für die digitale Signatur wichtigste Spezifikation ist die DIN-Spezifikation „Spezifikation der Schnittstelle zu Chipkarten mit digitaler Signatur-Anwendung/Funktion nach SigG und SigV“.

In der DIN-Spezifikation sind die folgenden Themenbereiche für Signaturkarten behandelt:

- Answer to Reset (ATR) und Protocol Type Selection (PTS)
- Übertragungsprotokolle (T = 0, T = 1)

- Dateien, Datenobjekte und Datenformate
- Authentisierung des Karteninhabers
- Berechnung und Überprüfung von Signaturen
- Protokollierung bei der Signaturerstellung
- Ablauf bei Erstellung und Überprüfung von Signaturen
- Zusammenarbeit mit Terminals und dazugehörige Kommandosequenzen

Tabelle 14.2 Gegenüberstellung der wesentlichen Unterschiede zwischen der ITSEC Prüfstufe E2 und E4 im Zusammenhang mit Signaturkarten

ITSEC Prüfstufe E2 hoch	ITSEC Prüfstufe E4 hoch
Komponenten zum Erfassen und Prüfen von Identifikationsdaten und zur Darstellung der zu signierenden Daten	Komponente zur Schlüsselerzeugung
Komponenten zur Darstellung der signierten Daten und zur Prüfung von Zertifikaten	Komponente zur Speicherung und Anwendung des Signaturschlüssels (i. d. R. Chipkarte)
Komponenten zur Erzeugung von Zeitstempeln	Komponenten zur Erzeugung und Prüfung von digitalen Signaturen, die geschäftsmäßig Dritten zur Benutzung angeboten werden

Der Systemaufbau

Die beiden wesentlichen Komponenten eines Systems für digitale Signatur sind einerseits das Trustcenter und andererseits die Signaturkarte. Diese beiden Instanzen stehen deshalb im Vordergrund, weil nur in ihnen die sicherheitstechnisch relevanten Abläufe stattfinden. Grundsätzlich kann dabei in einem System für digitale Signatur zwischen der Kartenausgabe und dem Signieren und Verifizieren unterschieden werden.

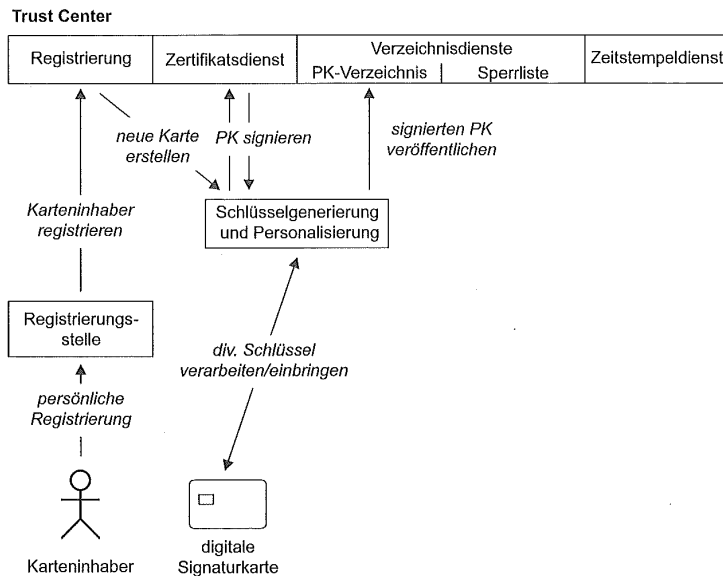


Bild 14.5 Die grundlegenden Abläufe bei der Kartenausgabe in der Anwendung „digitale Signatur“.

Kartenausgabe

Der zukünftige Benutzer einer digitalen Signaturkarte muss sich als Erstes bei einer Registrierungsstelle registrieren lassen. Dazu ist es erforderlich, dass er persönlich erscheint und einen anerkannten Identifikationsbeweis – wie seinen Personalausweis – vorlegt. Zu diesem Zeitpunkt kann der zukünftige Benutzer einer digitalen Signaturkarte wählen, ob er anstelle seines Namens ein Pseudonym benutzt. Die Registrierungsstelle gibt die erhaltenen und geprüften Informationen an die Registrierung des Trustcenters weiter, welche die Schlüsselgenerierung sowie Personalisierung einer Signaturkarte anstößt. Dieser Vorgang kann entweder im Trustcenter geschehen, von einer dritten Instanz oder auch direkt an der Registrierungsstelle durchgeführt werden. Bei diesem Prozess wird der öffentliche Schlüssel der neuen Chipkarte vom Zertifikatsdienst des Trustcenters signiert und damit als echter öffentlicher Schlüssel anerkannt. Der signierte öffentliche Schlüssel wird im Public-Key-Verzeichnis des Trustcenters anschließend als gültig eingetragen und steht damit jedem Systemteilnehmer für die Benutzung zur Verfügung. Nachdem dies abgeschlossen ist, erhält der neue Teilnehmer seine digitale Signaturkarte sowie einen PIN-Brief. Damit ist die Kartenausgabe abgeschlossen, und die Karte kann benutzt werden.

Signieren und Verifizieren von Dokumenten

Beim Signieren eines elektronischen Dokuments wird dieses zu einem Hash-Wert komprimiert und von der digitalen Signaturkarte mit dem geheimen Schlüssel signiert. Dieser Vorgang kann nur nach eindeutiger Identifizierung des Karteninhabers (d. h. des Signierers) gestartet werden, was momentan bedeutet, dass dieser seine PIN eingeben muss. In

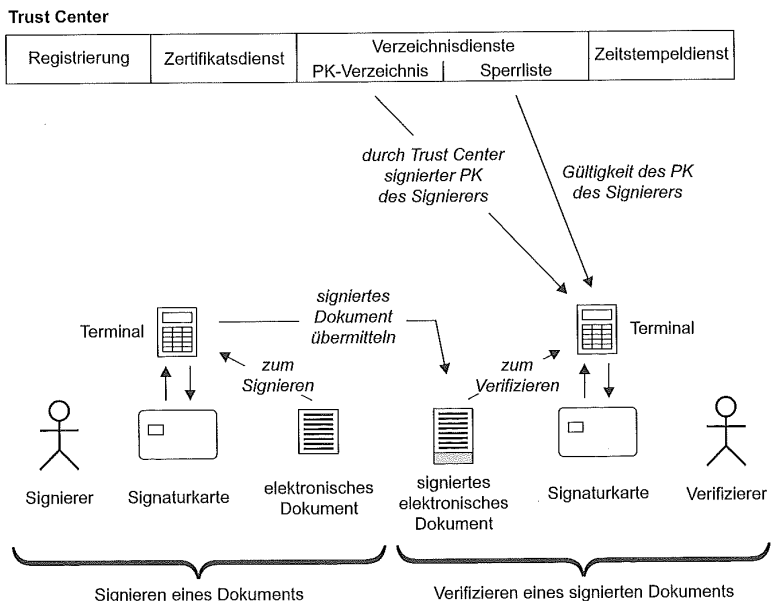


Bild 14.6 Die grundlegenden Abläufe beim Signieren und Verifizieren von signierten Dokumenten in der Anwendung „digitale Signatur“.

Zukunft könnte dies auch die Prüfung eines biometrischen Merkmals (z. B. Fingerabdruck) sein. Die beim Signieren benutzten Komponenten sind Signaturkarte, Terminal und im Regelfall ein PC.

Das digital signierte elektronische Dokument kann nun auf beliebigem Wege verschickt werden. Zur Prüfung dieses Dokuments ist wiederum ein Hash-Wert zu bilden. Bei Bedarf kann nun online der öffentliche Schlüssel des Signierers vom Public-Key-Verzeichnis des Trustcenters geholt werden. Mit diesem ist es nun möglich, zu überprüfen, ob die digitale Signatur über das elektronische Dokument echt ist. Um sicherzustellen, dass der geheime Schlüssel des Signierers nicht kompromittiert worden ist, muss bei der Signaturprüfung die Sperrliste (*revocation list*) des Trustcenters gefragt werden. Dies gibt dem Verifizierer die Sicherheit, dass der geheime Schlüssel des Signierers nicht gesperrt ist.

Benötigt der Verifizierer noch eine von Dritten überprüfbare Bestätigung dieser Sperrlistenabfrage, so kann er über den Zeitstempeldienst und in Verbindung mit den Verifikationsdaten eine für Dritte nachvollziehbare Bestätigung vom Trustcenter erhalten.

Die Systemstruktur und die dargestellten Abläufe bei der Ausgabe, beim Signieren und Verifizieren mit digitalen Signaturkarten sind Beispiele, die je nach Realisierung Abweichungen aufweisen können. Sie stellen jedoch einen realistischen, wenn auch vereinfachten Überblick über ein typisches digitales Signatursystem dar.

Das Trustcenter (TC)

Das Trustcenter ist neben der digitalen Signaturkarte das wesentliche Element in einem digitalen Signatursystem. Es hat sechs unterschiedliche Aufgaben: Registrierung von neuen Benutzern (*registry*), Schlüsselgenerierung mit Personalisierung von neuen digitalen Signaturkarten, Zertifikatsdienst (*certification service*), Verzeichnisdienst (*directory service*) für öffentliche Schlüssel (*public key directory service*), Verzeichnisdienst für Sperrlisten (*revocation list*) und Zeitstempeldienst (*time stamp service*).

Die Registrierung dient dazu, die persönlichen Daten und die Identität eines neuen Benutzers für digitale Signaturkarten zu erfassen. Im Regelfall ist zwischen Benutzer und Trustcenter eine Registrierungsstelle geschaltet, damit der Benutzer nicht persönlich beim Trustcenter erscheinen muss. Die Aufgaben Schlüsselgenerierung und Personalisierung müssen nicht zwangsläufig vom Trustcenter getätigt werden, sondern können beispielsweise auch bei einem Kartenhersteller geschehen. Die Schlüsselgenerierung kann sowohl außerhalb als auch innerhalb der Chipkarte stattfinden. Diese Funktion wird jedoch nur bei der Ausgabe von neuen Karten benötigt. Der Zertifikatsdienst eines Trustcenters wird benutzt, um öffentliche Schlüssel von digitalen Signaturkarten mit dem geheimen Schlüssel des Trustcenters zu unterschreiben und somit als echt anzuerkennen. Ein Trustcenter kann natürlich mehr als einen geheimen Schlüssel für seinen Zertifikatsdienst besitzen.

Die beiden Verzeichnisdienste für die öffentlichen Schlüssel und die Sperrlisten enthalten einerseits die vom Trustcenter signierten öffentlichen Schlüssel der digitalen Signaturkarten und andererseits die gesperrten Schlüssel von beispielsweise verloren gegangenen oder kompromittierten digitalen Signaturkarten. Diese Verzeichnisdienste müssen laut Signaturgesetz über allgemein verfügbare öffentliche Netze erreichbar sein.

Der Zeitstempeldienst muss nicht zwangsläufig in einem Trustcenter integriert sein, doch wird es wohl in den meisten Realisationen eines Trustcenters verfügbar sein. Dieser Dienst wird benutzt, um dem Trustcenter übergebene elektronische Informationen mit

aktuellem Datum und Uhrzeit zu versehen. Information und Zeitangabe werden dann mit dem geheimen Signaturschlüssel des Zeitstempeldiensts unterschrieben. Damit kann gegenüber Dritten bewiesen werden, dass die vom Zeitstempeldienst unterschriebene Information spätestens zu diesem Zeitpunkt vorhanden gewesen sein muss.

Die digitale Signaturkarte

Zur Aufbewahrung und Benutzung des geheimen Signaturschlüssels benötigt man eine sichere Hardware-Umgebung. Chipkarten stellen dafür eine ideale Lösung dar, da sie kleine Abmessungen haben, preisgünstig sind und einen hohen Schutz vor Auslesen und Beeinflussung von außen bieten. Man kann ohne Einschränkung behaupten, dass die digitale Signatur erst mit Chipkarten möglich geworden ist.

Die dabei verwendeten asymmetrischen Kryptoalgorithmen bedingen einen Mikrocontroller mit zusätzlicher numerischer Recheneinheit (NPU), damit Signaturen in akzeptabler Zeit erstellt und geprüft werden können. Theoretisch ließe ein signiertes Dokument auch außerhalb der Signaturkarte prüfen, da auch dort alle notwendigen Informationen vorliegen. Ob dies in der Praxis relevant wird, bleibt abzuwarten. Sicherheitstechnisch wäre es jedoch günstiger, die Prüfung in der Signaturkarte durchzuführen.

Die Schlüsselgenerierung von öffentlichen und geheimen Schlüsseln für eine digitale Signaturkarte kann entweder zentral oder dezentral erfolgen. In allen Fällen muss jedoch sichergestellt sein, dass sie in einer kryptographisch gesicherten Umgebung, wie z. B. einem Sicherheitsmodul oder einer Chipkarte, abläuft. Wichtig ist weiterhin, dass der generierte öffentliche Schlüssel mit dem geheimen Schlüssel des Trustcenters unterschrieben wird. Nur so ist sichergestellt, dass der geheime und der öffentliche Schlüssel der Signaturkarte vom Trustcenter als echt anerkannt sind. Ohne diese Anerkennung kann ein Dritter nicht prüfen, ob die digitale Signatur eines elektronischen Dokuments von einem echten Teilnehmer eines digitalen Signaturesystems stammt.

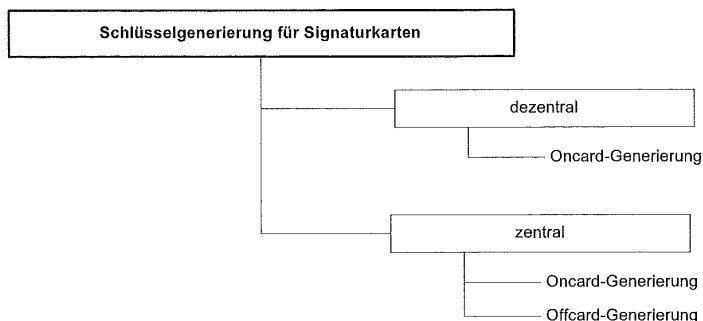


Bild 14.7 Die möglichen Varianten der Schlüsselgenerierung für Signaturkarten.

Die Schlüsselgenerierung lässt sich zentral oder dezentral durchführen. Bei der zentralen Schlüsselgenerierung werden alle Schlüsselpaare an einem Ort erzeugt und unmittelbar danach mit dem geheimen Schlüssel des Trustcenters unterschrieben. Sowohl Schlüsselgenerierung als auch Trustcentersignatur können oncard oder offcard erstellt werden. Die dezentrale Schlüsselgenerierung ist nur oncard möglich, da dies die Einzige kryptogra-

phisch gesicherte Umgebung in diesem Umfeld ist. Die Vor- und Nachteile der drei Verfahren sind in Tabelle 14.3 dargestellt. In der Praxis wird sich wahrscheinlich eine zentrale Oncard-Schlüsselgenerierung durchsetzen, da diese sehr sicher ist und auch dem bisherigen Herstellungsprozess von Chipkarten entspricht.

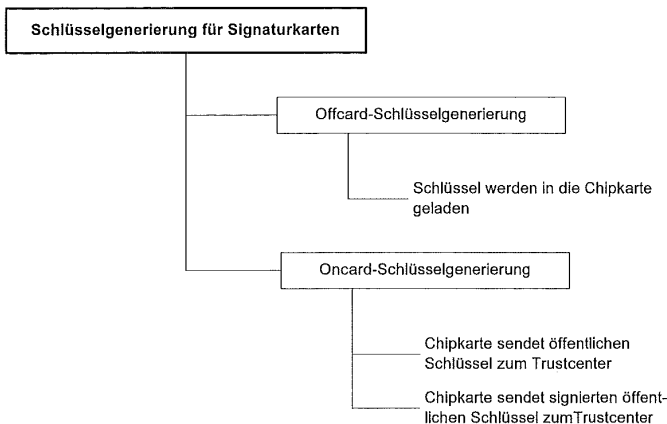


Bild 14.8 Die drei grundlegenden Möglichkeiten bei der Schlüsselgenerierung und Schlüsselverteilung zwischen Signaturkarte und Trustcenter.

Tabelle 14.3 Gegenüberstellung der verschiedenen Aspekte bei den drei benutzten Verfahren zur Schlüsselgenerierung. Die in der mittleren Spalte dargestellte Vorgehensweise wird in großen realen Anwendungen üblicherweise benutzt.

Verfahren	Offcard-Schlüsselgenerierung	Oncard-Schlüsselgenerierung mit Übergabe des öffentlichen Schlüssels zum Trustcenter zwecks Signierung	Oncard-Schlüsselgenerierung mit Übergabe des signierten öffentlichen Schlüssels zum Trustcenter
Vorteile	• schnelle Personalisierung • Nachproduktion von identischen Karten möglich	• der geheime Signaturschlüssel verlässt nie die Chipkarte	• der geheime Signaturschlüssel verlässt nie die Chipkarte
Indifferent	• Schlüsselerückgewinnung (key escrow) technisch problemlos möglich	• Schlüsselerückgewinnung (key escrow) schwierig	• Schlüsselerückgewinnung (key escrow) schwierig
Nachteile	• geheimer Schlüssel wird offcard erzeugt	• öffentlicher Schlüssel muss auf sicherem Weg zum Trustcenter gebracht werden • Nachproduktion von identischen Karten nicht möglich	• geheimer Schlüssel des Trustcenters in der Chipkarte • Nachproduktion von identischen Karten nicht möglich • zeitaufwendige Personalisierung

Das in Deutschland wichtigste Dokument zur Schnittstelle der digitalen Signaturkarte ist die bereits oben erwähnte DIN-Spezifikation. Sie beschreibt ausführlich die benötigten Kommandos, Dateien und Abläufe für eine digitale Signaturkarte.

Die Signaturanwendung befindet sich in einem DF direkt unter dem MF. Die DIN-Spezifikation sieht für diese Anwendungen 9 EFs vor, die alle für diese Anwendungen benötigten Daten enthalten. Alle verwendeten Datenelemente sind TLV-codiert. Die EFs und deren Dateninhalte sind in der Tabelle 14.4 im Überblick beschrieben.

Tabelle 14.4 Der auf die wesentlichen Dateien vereinfachte Dateibaum einer Signaturkarte nach deutschem Signaturgesetz.

Dateityp	FID	Struktur, Größe	Beschreibung
MF	'3F00'	---	Wurzelverzeichnis
MF.EF _{GDO}	'2F02'	transparent, x Byte	Seriennummer der Chipkarte, Name des Kartenbesitzers
DF _{Sig} G	undefiniert	---	DF digitale Signaturanwendung (AID = 'D2 76 00 00 66 01')
DF _{Sig} G.EF _{KEY}	---	undefiniert, x Byte	PINs, öffentliche/geheime Schlüssel
DF _{Sig} G.EF _{SSD}	'1F00'	transparent, x Byte	Informationen über Eigenschaften der Signaturkarte (security service description – SSD)
DF _{Sig} G.EF _{C.ICC.AUT}	'C000'	transparent, x Byte	Authentisierungszertifikat der Signaturkarte
DF _{Sig} G.EF _{C.CA.AUT}	'C008'	transparent, x Byte	Authentisierungszertifikat der Zertifizierungsstelle
DF _{Sig} G.EF _{DM}	'D000'	transparent, 8 Byte	Anzeigetext für das Terminal, damit der Benutzer dieses als echt erkennen kann (display message – DM)
DF _{Sig} G.EF _{C.CH.DS}	'C100'	transparent, x Byte	Signaturzertifikat der Kartenbesitzers
DF _{Sig} G.EF _{C.CA.DS}	'C108'	transparent, x Byte	Signaturzertifikat der Zertifizierungsstelle
DF _{Sig} G.EF _{PK.CA.DS}	'B000'	transparent, x Byte	öffentliche Schlüssel der Zertifizierungsstelle
DF _{Sig} G.EF _{PROT}	'A000'	cyclic, x Byte	Protokollierung der Signaturerstellung

Die für die Signaturkarte verwendeten Kommandos sind:

- APPEND RECORD
- EXTERNAL AUTHENTICATE
- GET CHALLENGE
- INTERNAL AUTHENTICATE
- MANAGE SECURITY ENVIRONMENT
- PERFORM SECURITY OPERATION
- READ BINARY
- READ RECORD
- SELECT FILE
- UPDATE BINARY
- Optional: MANAGE CHANNEL
- Optional für T = 0: GET RESPONSE

Bei der deutschen Signaturkarte sind die folgenden Hash-Funktionen und Signaturalgorithmen vorgesehen: SHA-1, RIPEMD-160, RSA, DSA und ECC. Da die Berechnung eines Hash-Wertes über große elektronische Dokumente in der Chipkarte zu lange dauern würde, besteht die Möglichkeit, den Hash-Wert im PC zu bilden und dann zur Signierung an die Karte zu übergeben. Alternativ dazu kann der Hash-Wert bis auf den letzten Block extern im PC berechnet werden. Anschließend wird der letzte Block mit dem entsprechenden Teil des Dokuments in der Chipkarte gerechnet und signiert. Die vollständige Berechnung eines Hash-Wertes in der Chipkarte ist ebenfalls möglich, wird jedoch oft aus bekannten Performance-Gründen nicht sinnvoll sein.

Resümee und Ausblick

Das Signaturgesetz hat in Deutschland den gesetzlich notwendigen Rahmen geschaffen, digitale Signaturen im Alltagsleben verlässlich zu benutzen. Die Einsatzgebiete werden wahrscheinlich weniger private Kaufverträge über Häuser oder Testamente sein als der tägliche Umgang mit E-Mail, Bezahlung und Bestellung über öffentliche – und damit per se – unsichere Netze.

Man kann sicherlich davon ausgehen, dass es neben den gesetzeskonformen Lösungen auch viele nicht gesetzeskonforme Lösungen geben wird, da diese weitaus kostengünstiger und einfacher zu realisieren sind. Die in Deutschland sehr hoch liegenden Anforderungen für die Sicherheit der eingesetzten Komponenten wird kostengünstige und schnelle Lösungen nicht begünstigen. Andererseits ist dieses Sicherheitsniveau notwendig, da eine elektronische Unterschrift der eigenhändigen Unterschrift gleichgestellt wurde. Käme es zu erfolgreichen Manipulationen, würde das ganze System „digitale Signatur“ in Unglaublichkeit gestürzt und müsste im schlimmsten Fall sogar außer Betrieb gehen.

Ein weiterer wichtiger Aspekt in diesem Umfeld ist das Verhalten von weltweit operierenden Organisationen, die wahrscheinlich nicht auf die Gesetze eines einzelnen Landes Rücksicht nehmen werden. Deswegen wäre es durchaus möglich, dass sich eine nicht gesetzeskonforme Lösung als internationaler Quasi-Standard etabliert.

Trotz dieser Risiken hat die deutsche Lösung für digitale Signaturen weltweiten Einfluss, sowohl auf Gesetzgebung als auch auf technische Realisierung von ähnlichen Projekten. Wird das Potenzial dieser Lösung genutzt, so wird es in Deutschland eine technisch hochwertige und vorbildliche Lösung für die Chipkarten-Anwendung „digitale Signatur“ geben.

14.5 Signaturanwendung nach PKCS #15

Im Jahr 1998 wurde von der Firma RSA Inc. [RSA] ein Papier namens „Cryptographic Token Information Format Standard“ vorgestellt. Diese Spezifikation umfasst die Beschreibung von Dateien mit dazugehörigem Datenformaten für ein kryptografisches Token. Bei dem Token handelt es sich vorzugsweise um eine Chipkarte mit Kryptokoprosessor. Als Ausgangsdokumente für PKCS #15 (*public key cryptography standards*) wurden die DC/SC- (*digital certificates on smart cards*) und SEIS-Spezifikationen (*secured electronic information in society*) für Chipkarten-Anwendungen mit digitaler Signatur herangezogen. Mittlerweile ist die PKCS #15 Spezifikation auch in kompatibler Form von der in-

ternationalen Chipkartennorm ISO/IEC 7816-15 abgedeckt. Im Folgenden wird jedoch auf die PKCS #15 Spezifikation referiert, da sich diese weltweit als Referenz etabliert hat.

Bei der Erarbeitung der PKCS #15 Spezifikation wurden verschiedene Anforderungskriterien beachtet. Die Spezifikation sollte Plattform-, Hersteller- und anwendungsneutral sein und sich auch konform zu den üblichen Normen verhalten. Außerdem musste sie modular und erweiterbar aufgebaut sein. Auch war gefordert, dass alle Daten der Anwendung mit einer ausreichenden Beschreibung versehen sind, so dass diese beim Zugriff als Referenz verwendet werden kann.

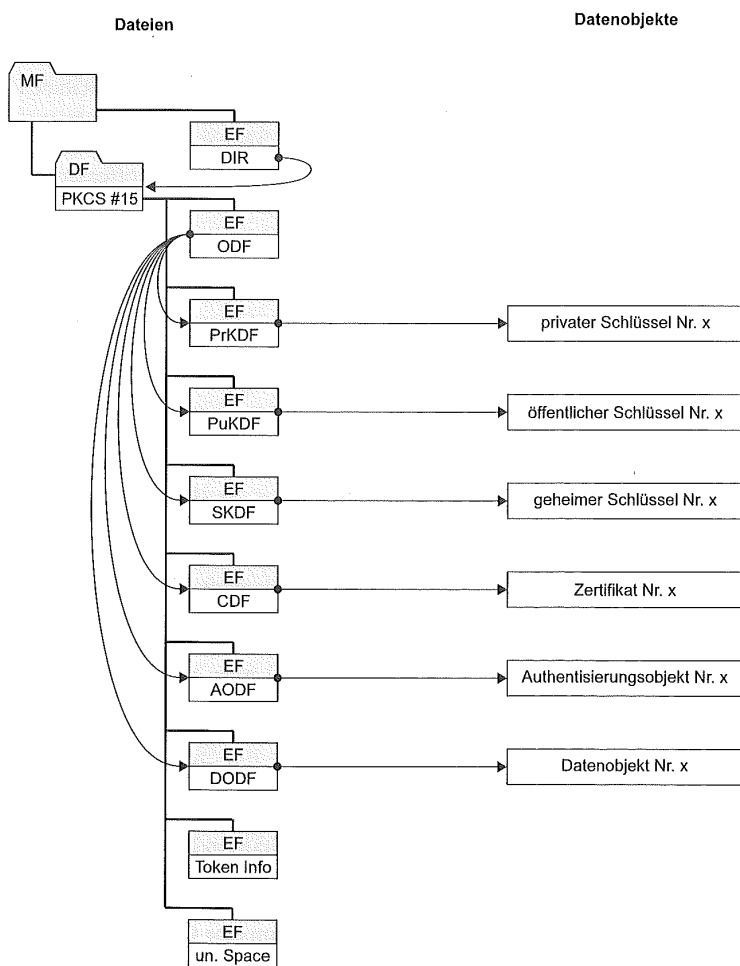


Bild 14.9 Überblick über Dateien und Datenreferenzen mittels Zeiger einer Signatur-Anwendung nach PKCS #15. Die Variable x steht für einen ganzzahligen positiven Zähler größer Null und kann für die unterschiedlichen Datenobjekte verschiedene Werte annehmen.

Als Grundlage für die Darstellung von Dateien wurde die ISO/IEC 7816-Normenreihe, für die Formatbeschreibung wurde ASN.1 gewählt. Die für die jeweilige Abarbeitung der Daten notwendigen Kommandos sind nicht Gegenstand der PKCS #15-Spezifikation. Aus diesen vorgenannten Gründen lässt sich PKCS #15 verhältnismäßig einfach auf Chipkarten mit typischen Multiapplikations-Betriebssystemen realisieren. Seitens des Konzepts kann jedoch jedes Sicherheitstoken für die PKCS #15-Funktionen benutzt werden.

Umgesetzt wurde PKCS #15 bisher bei einigen großen Chipkarten Anwendungen wie zum Beispiel beim finnischen chipkartengestützten Personalausweis FINEID⁴ und im Rahmen der WIM⁵ (*wireless identification module*), der Signaturanwendung auf SIMs und USIMs für WAP (*wireless application protocol*) und ähnliche Einsatzzwecke in der Telekommunikation. Innerhalb kurzer Zeit hat sich die PKCS #15 Spezifikation auf breiter Basis als internationaler Industriestandard für die Datenrepräsentation von Signaturanwendungen auf Chipkarten durchgesetzt.

Die gesamte PKCS #15-Anwendung befindet sich innerhalb eines DFs mit einem festgelegten AID. Der RID hat den Wert 'A0 00 00 00 63' und der dazugehörige PIX ist die ASCII-Codierung von "PKCS-15" und besteht demnach aus der Bytesequenz '50 4B 43 53 2D 31 35'. Je nach Realisierung kann die PIX jedoch auch einen anderen Wert aufweisen.

Als Referenz auf das DF mit der PKCS #15-Anwendung kann ein Eintrag in einem optionalen und durch ISO/IEC 7816 genormten EF_{DIR} direkt unter dem MF benutzt werden. Sollten mehrere PKCS #15 konforme Anwendungen auf der Chipkarte vorhanden sein, können diese über das EF_{DIR} eindeutig identifiziert und ausgewählt werden. Alle weiteren Dateien der Anwendung befinden sich unter dem DF_{PKCS #15}.

Das EF_{TokenInfo} innerhalb des DF_{PKCS #15} enthält generelle Informationen über die Chipkarte, welche die PKCS #15-Anwendung beherbergt. Dies sind unter anderem eine Seriennummer, Herstellkennung und unterstützte Kryptoalgorithmen.

Das zentrale und wichtigste EF der PKCS #15-Anwendung ist das EF_{ODF} (*object directory file*), da es ein Verzeichnis der Verzeichnisdateien enthält, auf die mittels Zeigern verwiesen wird, sodass sie eindeutig identifiziert werden können. Diese EF ist dann sozusagen eine Art Wurzelverzeichnis, von dem aus alle anderen EFs der Anwendung durch Zeiger adressiert werden können.

Die Verzeichnisdateien EF_{PrKDF}, EF_{PrKDF}, EF_{SKDF}, EF_{CDF} und EF_{AODF} verfügen über einen einheitlichen internen Aufbau mit einem Verzeichnis und einem pro Verzeichniseintrag dazugehörigen Zeiger zu den jeweiligen Daten. Diese Daten können in Dateien innerhalb des DF_{PKCS #15} abgelegt sein oder sich auch in Dateien eines anderen DFs der Chipkarte befinden. Die Zeigeradressierung lässt auch zu, auf Daten innerhalb eines EFs der Struktur transparent mit Offset und Länge zu verweisen. Es wäre also zumindest theoretisch denkbar, alle Daten in einem einzigen großen EF abzulegen, wenn dies auch aus Gründen der Datenorganisation kaum sinnvoll sein würde. Die Verknüpfung mittels Zeiger ist sogar dahingehend ausgelegt, dass sogar mit einer URL auf Daten extern der Chipkarte verwiesen werden kann. Eine charakteristische Anwendung dieses Mechanismus wäre die Hinterlegung eines Zertifikats zur Prüfung der Signatur bei einem Trustcenter, das somit nicht auf

⁴ Siehe auch 14.6 „Personalausweis FINEID“.

⁵ Siehe auch 13.5 „Die WIM“.

der Chipkarte gespeichert werden muss. Seitens der Chipkarte würde dabei nur mittels einer URL auf das Zertifikat verwiesen.

Die Verzeichnisdateien müssen mindestens einmal auf der Chipkarte vorhanden sein, es ist allerdings auch erlaubt, mehrere Dateien mit dem gleichen Funktionszweck auf einer Chipkarte zu erzeugen. Das EF_{ODF} dient dann als Verteiler auf die jeweiligen Dateien.

Die Verzeichnisdateien EF_{PrKDF} (*private key directory file*) und EF_{PuKDF} (*public key directory file*) enthalten ein Verzeichnis der privaten respektive öffentlichen Schlüssel auf der Chipkarte. Die beiden Dateien enthalten somit die Schlüssel für asymmetrische Kryptoprogramme, wie RSA, DAS oder elliptische Kurven. Im Verzeichnis der Dateien sind Zeiger auf die dazugehörigen Schlüssel vorhanden. Analog ist das EF_{SKDF} (*secret key directory file*) aufgebaut, in dem sich ein Verzeichnis mit Zeigern zu den geheimen Schlüsseln für symmetrische Kryptoprogramme auf der Chipkarte befindet. Im EF_{CDF} (*certificate directory file*) und EF_{AODF} (*authentication object directory file*) finden sich auch Verzeichnisse mit Zeigern zu Zertifikaten bzw. Authentisierungsobjekten. Typische verwendete Zertifikate sind X.509 Zertifikate; Authentisierungsobjekte sind beispielsweise PINs oder auch Daten für die biometrische Benutzeridentifizierung. Entsprechend dem EF_{AODF} verhält es sich mit dem EF_{DODF} (*data object directory file*), das ein Verzeichnis mit Zeigern auf Datenobjekte enthält.

Die optionale Datei $EF_{UnusedSpace}$ ist zur Verwaltung von nicht genutzten Speicher in bereits vorhandenen EFs der PKCS #15-Anwendung vorgesehen.

Tabelle 14.5 Aufstellung der typischen Dateien einer Signaturanwendung nach PKCS #15. Die Zugriffsbedingungen werden unterschieden in eine Nurlese-Chipkarte (*read-only card* – *R/O*) für nach der Kartenausgabe nicht mehr änderbare Hochsicherheitsanwendungen und eine Schreiblese-Chipkarte (*read-write card* – *R/W*) für normale Sicherheitsanwendungen.

$DF_{PKCS\#15}$	PKCS #15-Anwendung
Beschreibung	In diesem Verzeichnis sind alle Dateien zusammengefasst, die zur PKCS #15-Anwendung gehören.
Datei	AID = 'A0 00 00 00 63 50 4B 43 53 2D 31 35'
MF, $DF_{PKCS\#15}$, EF_{ODF}	Objektverzeichnisdatei (<i>ODF</i> – <i>object directory file</i>)
Beschreibung	Diese Datei enthält ein Verzeichnis aller vorhandener Dateien mit Datenobjekten sowie entsprechende Zeiger auf diese Dateien.
Datei	FID = '5031'; Zugriffe R/O: READ: immer; UPDATE: niemals; APPEND: EXTERNAL AUTHENTICATE / niemals Zugriffe R/W: READ: immer; UPDATE: EXTERNAL AUTHENTICATE / niemals; APPEND: EXTERNAL AUTHENTICATE / niemals
MF, $EF_{PKCS\#15}$, EF_{PrKDF}	Verzeichnisdatei der privaten Schlüssel (<i>PrKDF</i> – <i>private key directory file</i>)
Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen privaten Schlüssel sowie entsprechende Zeiger auf diese Schlüssel.
Datei	FID = vom Anwendungsherausgeber festzulegen Zugriffe R/O: READ: immer / PIN; UPDATE: niemals; APPEND: EXTERNAL AUTHENTICATE / niemals Zugriffe R/W: READ: immer / PIN; UPDATE: PIN; APPEND: PIN

Tabelle 14.5 (Fortsetzung)

MF.DF _{PKCS #15} .EF _{PuKDF}		Verzeichnisdatei der öffentlichen Schlüssel (<i>PuKDF – public key directory file</i>)
	Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen öffentlichen Schlüssel sowie entsprechende Zeiger auf diese Schlüssel.
	Datei	siehe EF _{PrKDF}
MF.DF _{PKCS #15} .EF _{SKDF}		Verzeichnisdatei der geheimen Schlüssel (<i>SKDF – secret key directory file</i>)
	Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen geheimen Schlüssel sowie entsprechende Zeiger auf diese Schlüssel.
	Datei	siehe EF _{PrKDF}
MF.DF _{PKCS #15} .EF _{CDF}		Verzeichnisdatei der Zertifikate (<i>CDF – certificate directory file</i>)
	Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen Zertifikate sowie entsprechende Zeiger auf diese Zertifikate.
	Datei	siehe EF _{PrKDF}
MF.DF _{PKCS #15} .EF _{AODF}		Verzeichnisdatei der Authentisierungsobjekte (<i>AODF – authentication object directory file</i>)
	Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen Authentisierungsobjekte sowie entsprechende Zeiger auf diese Objekte.
	Datei	FID = vom Anwendungsherausgeber festzulegen Zugriffe R/O: READ: immer; UPDATE: niemals; APPEND: niemals Zugriffe R/W: READ: immer; UPDATE: PIN / EXTERNAL AUTHENTICATE / niemals; APPEND: PIN / EXTERNAL AUTHENTICATE / niemals
MF.DF _{PKCS #15} .EF _{DODF}		Verzeichnisdatei der Datenobjekte (<i>DODF – data object directory file</i>)
	Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen Datenobjekte sowie entsprechende Zeiger auf diese Objekte.
	Datei	siehe EF _{PrKDF}
MF.DF _{PKCS #15} .EF _{TokenInfo}		Verzeichnisdatei der privaten Schlüssel (<i>PrKDF – private key directory file</i>)
	Beschreibung	Diese Datei enthält ein Verzeichnis der vorhandenen privaten Schlüssel sowie entsprechende Zeiger auf diese Schlüssel.
	Datei	FID = '5032' Zugriffe R/O: READ: immer; UPDATE: niemals; APPEND: niemals Zugriffe R/W: READ: immer; UPDATE: PIN / AUTHENTICATE / niemals; APPEND: niemals
MF.DF _{PKCS #15} .EF _{UnusedSpace}		Verzeichnisdatei des Freispeichers
	Beschreibung	Diese Datei enthält ein Verzeichnis des in den EFs der PKCS #15-Anwendung vorhandenen, jedoch nicht genutzten Speicherplatzes.
	Datei	FID = '5033' siehe EF _{PrKDF}

Die ASN.1 Beschreibung der möglichen Daten einer PKCS #15-Anwendung umfasst in der Originalspezifikation 14 Seiten. Wir haben es nicht als sinnvoll erachtet, diese hier Zeile für Zeile abzuschreiben, da das Originalpapier kostenlos via Internet auf [RSA] erhältlich ist und wir seitenlange Listings zur Erhöhung des Buchumfangs weder in diesem noch in anderen Büchern zugetan sind. Als Beispiel befindet sich jedoch in Tabelle 14.6 eine leicht komprimierte ASN.1 Beschreibung der Eigenschaften von PINs.

Tabelle 14.6 Beispiel für die ASN.1 Beschreibung der Eigenschaften der PIN nach PKCS #15 in leicht vereinfachter Form

PinAttributes ::= SEQUENCE {	Definition des Datenobjekts für die Eigenschaften der PIN
pinFlags PinFlags,	Die möglichen binären Eigenschaften der PIN sind im Datenobjekt PinFlags definiert.
pinType PinType,	Die Codierung der PIN ist im Datenobjekt PinType definiert.
minLength INTEGER,	Dies ist die Angabe der minimal erlaubten Länge der PIN.
storedLength INTEGER,	Dies ist die Angabe der Länge der gespeicherten PIN.
maxLength INTEGER OPTIONAL,	Dies ist die optionale Angabe der maximal erlaubten Länge der PIN.
padChar OCTET STRING OPTIONAL,	Dies ist die optionale Angabe eines Füllzeichens zum Padding der PIN.
lastPinChange GeneralizedTime OPTIONAL,	Mit dieser optionalen Zeitangabe wird die letzte Änderung der PIN protokolliert.
path Path OPTIONAL	Dies ist die optionale Angabe eines Pfades zum DF, für das die PIN verwendet wird.
}	
PinFlags ::= BIT STRING {	Definition der binären Eigenschaften der PIN
case-sensitive(0),	Angabe über notwendige Unterscheidung von Groß- und Kleinschreibung.
local(1),	Angabe, ob es sich um eine anwendungsspezifische, d. h. lokale, oder um eine kartenweite, d. h. globale PIN handelt.
change-disabled(2),	Angabe, ob die PIN geändert werden darf.
unblock-disabled(3),	Angabe, ob der Fehlbedienungszähler der PIN zurückgesetzt werden darf.
initialized(4),	Angabe, ob die PIN initialisiert ist.

Tabelle 14.6 (Fortsetzung)

<code>needs-padding(5),</code>	Angabe, ob die übergebene PIN mit einem Füllzeichen gepadded werden muss.
<code>unlockingPin(6),</code>	Angabe, ob die PIN zum Rücksetzen eines PIN Fehlbedienungs Zählers benutzt werden darf.
<code>soPin(7),</code>	Angabe, ob es sich um eine Security Officer PIN, gemäß PKCS#11, handelt.
<code>disable-allowed(8),</code>	Angabe ob die Abfrage zu dieser PIN abgeschaltet werden darf.
<code>integrity-protected(9),</code>	Angabe, ob die PIN nur über Secure Messaging (gesichert mit MAC) der Chipkarte mitgeteilt werden darf.
<code>confidentiality-protected(10),</code>	Angabe, ob die PIN nur über Secure Messaging (gesichert durch Verschlüsselung) der Chipkarte mitgeteilt werden darf.
<code>exchangeRefData(11)</code>	Angabe, ob beim Wechsel der PIN alter und neuer Wert der PIN übergeben werden müssen oder nur der neue Wert.
<code>}</code>	
PinType ::= ENUMERATED {	Definition der Wertedarstellung der PIN
<code> bcd, ascii-numeric, utf8, ...,</code>	Die PIN kann folgendermaßen codiert sein: BCD, ASCII, UTF8, halbnibble-BCD, ISO 9561-1
<code> half-nibble-bcd, iso9561-1}</code>	

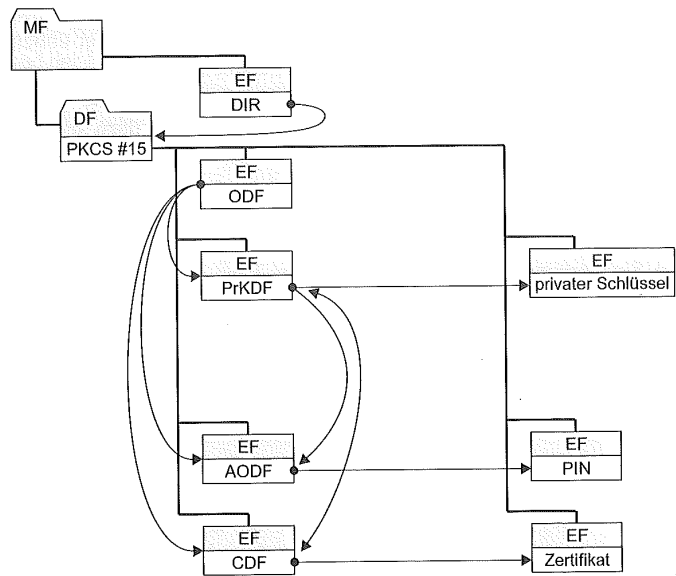


Bild 14.10 Beispiel für eine typische Zeigerstruktur nach PKCS #15 zum Schutz eines Signaturschlüssels durch eine vorherige PIN-Abfrage und einer Verbindung zu einem Zertifikat, das zur Überprüfung der mit dem Signaturschlüssel unterschriebenen Daten benutzt werden kann. Die Zeichnung enthält nur die für das prinzipielle Verständnis notwendigen Dateien.

Nach PKCS #15 ist es möglich, den Zugriff auf bestimmte Datenobjekte wie beispielsweise Signaturschlüssel, geheime Schlüssel oder Zertifikate, zu reglementieren. Dies wird entsprechend dem Konzept dieser Spezifikation wiederum über Zeiger realisiert, die vom zu schützenden Datenobjekt auf die entsprechende Verzeichnisseiten mit den Informationen über den zu verwendenden Authentisierungstyp verweisen. Beispielsweise lässt sich so auf einfache Weise ein Signaturschlüssel nur nach vorheriger PIN-Eingabe des Benutzers verwenden. In diesem Fall würde von der Information über den Signaturschlüssel im EF_{PrKDF} ein Zeiger auf einen Eintrag im EF_{AODF} mit den Informationen über die zu benutzende PIN weisen. In den jeweiligen Datensätzen des EF_{PrKDF} und EF_{AODF} sind dann Zeiger gespeichert, die auf die entsprechenden EFs mit den eigentlichen Daten verweisen. Um das Beispiel zu vervollständigen, würde sinnigerweise im EF_{PrKDF} ein Zeiger einen Eintrag im EF_{CDF} markieren, der Informationen über das zum Signaturschlüssel zugehörige Zertifikat enthält, sodass die Signatur von Dritten geprüft werden kann. In Bild 14.10 ist dieses Beispiel nochmals visualisiert. Dieses Konzept auf der Grundlage von Zeigern ist im ersten Augenblick etwas gewöhnungsbedürftig, jedoch sehr mächtig, flexibel und vor allem auch für zukünftige Ergänzungen offen.

14.6 Personalausweis FINEID

Im Dezember 1999 wurde in Finnland als erstem Land der Welt damit begonnen, elektronische Personalausweise auszugeben. Der Name des Systems ist FINEID (*Finnish Electronic Identification Card*). Emittiert wird ein Personalausweis mit Foto, Unterschrift und diversen anderen Informationen in ID-1 Größe. Die Karte enthält an der üblichen Position zusätzlich einen Chipkarten-Mikrocontroller mit kontaktbehafteter Datenübertragung.

Die FINEID-Chipkarte ist Teil einer Public-Key-Infrastruktur und basiert auf einer PKCS #15-konformen Signaturanwendung.⁶ Die Spezifikationen dazu sind öffentlich und via Internet kostenlos erhältlich [FINEID]. Die FINEID-Chipkarte ist jeweils drei Jahre gültig und kann auch als Reisedokument in den meisten europäischen Ländern verwendet werden. Dazu genügen die optisch von Menschen lesbaren Informationen auf der Chipkarte, der Mikrocontroller muss dazu nicht benutzt werden.

Durch die Signaturanwendung auf der Chipkarte kann sie jedoch auch zur Erstellung von rechtsgültigen digitalen Signaturen benutzt werden, was die gesamte Bandbreite an e-Commerce und e-Government für den Benutzer eröffnet.

14.7 Tachosmart

Nach einer Verordnung des Rates der Europäischen Gemeinschaften [EG 98] sind die Kontrollgeräte im Straßenverkehr, welche im normalen Sprachgebrauch schlicht und ergreifend Fahrtenschreiber genannt werden, mittelfristig durch elektronische Fahrtenschreiber mit Chipkarte zu ersetzen. Stark verallgemeinernd wird in der oben genannten Verordnung die Karte des Fahrers als Speicherkarte bezeichnet, da sie in der Lage sein muss, Daten zu speichern.

⁶ Siehe auch 14.5 „Signaturanwendung nach PKCS #15“.

Das Prinzip von Tachosmart ist Folgendes: Durch entsprechende Weg- und Geschwindigkeitssensoren misst der gegen Manipulationen gesicherte Fahrtenschreiber den zurückgelegten Weg sowie die Geschwindigkeit des Fahrzeugs. Des Weiteren enthält er eine ebenfalls gegen äußere Beeinflussung gesicherte Echtzeituhr sowie ein Terminal für Chipkarten. Der Fahrer des Fahrzeugs erhält eine auf ihn personalisierte Chipkarte, Fahrerkarte genannt, mit der er sich gegenüber dem Fahrtenschreiber ausweisen kann.

Der Fahrtenschreiber kann nun die fahrerbezogene Lenkzeit und auch die Geschwindigkeit überwachen und protokollieren. In der EU-Verordnung ist vorgesehen, dass die Daten von mindestens 28 Kalendertagen für den jeweiligen Fahrer in der Chipkarte gespeichert werden. Es ist erlaubt, bei Bedarf ältere Daten zu überschreiben. Der Fahrtenschreiber selber muss jedoch die Daten der letzten 365 Kalendertage in manipulationssicherer Form speichern.

Wie bei EU-Verordnungen üblich, werden die Details über die Gesetze der Mitgliedsstaaten und Spezifikationen der Hersteller von Fahrtenschreibern geregelt. Leider sind die technischen Dokumente vertraulich, sodass es nicht möglich ist, Details zu veröffentlichen. Voraussichtlich werden jedoch Chipkarten mit digitaler Signatur verwendet werden, die den Normen ISO/IEC 7816-4 und ISO/IEC 7816-8 entsprechen. Eine digitale Signatur ist notwendig, weil die im Fahrtenschreiber gespeicherten Daten der letzten 365 Tage gegen Veränderungen geschützt werden müssen.

Es existieren noch weitere Karten in diesem System. Die Prüfstellenkarte wird von bei den Behörden zugelassenen Stellen verwendet, um die Kalibrierung und Programmierung des Fahrtenschreibers durchzuführen. Die Kontrollkarte wird von Behörden und die Unternehmenskarte vom Eigentümer des Fahrzeugs benutzt, um auf die Daten des Fahrtenschreibers und der Fahrerkarten zuzugreifen. Um die doch ziemlich aufwändige Infrastruktur sicher, wirtschaftlich und europaweit zu betreiben, ist eine verhältnismäßig umfangreiche PKI (*public key infrastructure*) notwendig.

15 Design von Anwendungen

15.1	Allgemeine Hinweise und Kennzahlen	852
15.1.1	Mikrocontroller	852
15.1.2	Anwendungen	854
15.1.3	System	857
15.1.4	Normenkonformität	858
15.2	Formelsammlung zur Abschätzung von Ausführungszeiten	859
15.3	Zeitfunktionen typischer Chipkarten-Kommandos	867
15.4	Typische Ausführungszeiten von Kommandos	870
15.5	Hilfsmittel zur Anwendungsgenerierung	872
15.6	Analyse einer unbekannten Chipkarte	876
15.7	Vorgehensmodelle und Prozessreifegrad	879
15.7.1	Vorgehensmodelle (<i>life cycle models</i>)	883
15.7.2	Prozessreifegrad (<i>process maturity</i>)	890
15.8	Ablauf eines Chipkarten-Projekts	893
15.9	Beispiele für die Konzeption von Chipkarten-Anwendungen	894
15.9.1	Börse für Spielautomat	896
15.9.2	Zugangskontrolle	899
15.9.3	Prüfung auf Echtheit eines Terminals	902

Dieses Kapitel enthält im ersten Teil allgemeine Hinweise und Kennzahlen für den Einsatz von Chipkarten. Diese aus vielen Anwendungen zusammengetragenen Informationen sind direkt für die Konzeption einer Chipkarten-Anwendung verwendbar. Sie geben ebenso einen kurzen Überblick über den heutigen Stand der Technik.

Viele technische Angaben sind Werte, die stark abhängig von der verwendeten Hardware sind. Allerdings lässt sich damit trotzdem eine für die Praxis ausreichende Abschätzung für zu projektierende Anwendungen vornehmen. Dies ist unter anderem auch deshalb der Fall, weil die physikalischen und elektrischen Eigenschaften der auf dem Markt angebotenen Chipkarten-Mikrocontroller innerhalb gewisser Grenzen praktisch alle gleich sind. Die Hauptunterschiede liegen eigentlich nur in den angebotenen Speichergrößen.

Im zweiten Abschnitt werden überblickshaft Prinzipien für Werkzeuge vorgestellt, mit denen ohne Programmierung auf einfache und schnelle Art auch komplexe Anwendungen erstellt werden können. In den anschließenden Unterkapiteln sind anhand von drei Beispielen einige mögliche Anwendungen vorgestellt. Sie wurden systematisch aufgebaut, sodass die verschiedenen Mechanismen und Abläufe leicht nachvollziehbar sind.

Die aufgeführten Hinweise und Beispiele dieses Kapitels sind bewusst so allgemein und flexibel ausgelegt, dass sie als eine Art Design Pattern zur Gestaltung beliebiger Anwendungen benutzt werden können.¹

¹ Zusätzliche Gestaltungshinweise, vor allem hinsichtlich der Sicherheit, befinden sich im Abschnitt 8.2.4.2 „Angriffe auf der logischen Ebene“.

15.1 Allgemeine Hinweise und Kennzahlen

15.1.1 Mikrocontroller

Produktion

Muss für eine neue Anwendung ein spezielles Betriebssystem für Chipkarten programmiert werden, erfordert dies sehr viel Zeit. Ein Richtwert bezüglich der Dauer für die Konzeption, Programmierung und den Test eines völlig neuen Betriebssystems ist etwa 12 Monate. Wenn man Bibliotheksroutinen und schon vorhandene Programmteile verwenden kann, benötigt eine einzelne Person immer noch etwa 4 bis 6 Monate. Eine Parallelisierung der Tätigkeiten lässt sich aufgrund der sehr komplexen Programmierung in Assembler oder C nur eingeschränkt vornehmen. Dies führt dazu, dass mit der heutigen Softwaretechnologie selbst bei einem größtmöglichen Aufwand und Personaleinsatz die Erstellung eines neuen Chipkarten-Betriebssystems mindestens 2 bis 3 Monate dauert.

Die Entwicklung von Betriebssystemen, die ausführbaren Programmcode unterstützen, ist noch komplexer und aufwendiger. Als Richtwert kann hier auch bei günstigen Voraussetzungen von einer Entwicklungszeit von etwa 18 Monaten ausgegangen werden, sofern keine vorhandenen Konzepte oder Sourcecode genutzt werden können.

Nach Produktionsstart des Mikrocontrollers bei einem Halbleiterhersteller dauert es ungefähr 8 bis 12 Wochen, bis fertige Dice in mittleren Stückzahlen (d. h. einige 10000 Stück) verfügbar sind. Der Einbau der Dice in Module benötigt dann noch zusätzlich ein bis zwei Wochen. Somit beträgt die Durchlaufzeit bei der Herstellung der Mikrocontroller etwa 12 Wochen.

Lebensdauer

Für die Lebensdauer einer Chipkarte sind im Wesentlichen die Biegebeständigkeit des Kartenkörpers, die Korrosionsbeständigkeit der Kontakte des Moduls und die Anzahl der Schreib-/Löschzyklen des EEPROMs verantwortlich.

Die Lebensdauer des Kartenkörpers ist stark abhängig vom Einsatzgebiet. Bei typischen Telekommunikationskarten (SIM, USIM), die sich ständig im Mobiltelefon befinden und nie herausgenommen werden, gibt es seitens des Kartenkörpers praktisch keine Lebensdauerbeschränkung. Beim anderen Extrem – beispielsweise einem Betriebsausweis, der auch noch für Kantine und Zugangskontrolle verwendet wird – kann es schon zu einem Bruch des Kartenkörpers in zwei bis drei Jahren kommen. Das Mitführen der Karte in einer Geldbörse in der Gesäßtasche begünstigt dies noch.

Die zweite Beschränkung der Lebensdauer ist die maximale Anzahl der Steckzyklen. Die aus diesem Grund hartvergoldeten Kontakte können etwa 20000-mal kontaktiert werden. Danach ist die Oberfläche so stark verkratzt, dass daran anhaftender Schmutz und Fett eine sichere elektrische Kontaktierung verhindern. Zusätzlich kann aufgrund der dann weitgehend abgetragenen Goldschicht eine Oxidation eintreten, was sich auf die elektrische Leitfähigkeit der Kontakte ebenfalls nachteilig auswirkt. Die Lebensdauer der Kontakte und somit der Chipkarte ist natürlich auch sehr von der Form und den Kräften der Kontaktierungseinheit des Terminals abhängig. Sind diese optimal gestaltet, kann sich durchaus eine Erhöhung der Lebensdauer um den Faktor 2 bis 4 ergeben.

Die größte Beschränkung der Lebensdauer einer Chipkarte folgt aus der maximalen Anzahl der Schreib-/Löschzyklen des EEPROMs.² Die üblicherweise von den meisten Halbleiterherstellern garantierte Anzahl von Schreib-/Löschzyklen für jede Speicherseite des EEPROM beträgt minimal 100000. In der Praxis fällt das EEPROM bei Raumtemperatur und kleinen Abweichungen der Versorgungsspannung jedoch erst ab 500000 bis 1 Million Zyklen aus.

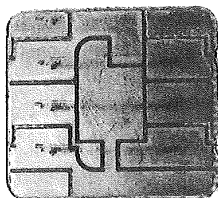


Bild 15.1 Ein Modul nach 50 000 Steckzyklen. Die dabei eingesetzten Kontakte gehörten zu einer für diese Art von Test konstruierten Prüfmaschine und waren sehr hochwertig. In der Praxis kann es, abhängig von der Kontaktierungseinheit, unter Umständen zu einem erheblich schnelleren Verschleiß der Kontaktflächen kommen.

Der Ausfall einer EEPROM-Seite ist dabei nicht schlagartig, sondern allmählich. Ein Anzeichen für den beginnenden Ausfall besteht darin, dass sich das EEPROM beim ersten Schreibversuch nicht mehr auf den gewünschten Wert setzen lässt und die geschriebenen Daten nach wenigen Stunden nicht mehr im Speicher vorhanden sind. Benutzt man eine solche Speicherseite weiter, lassen sich nach einigen weiteren tausend Zyklen überhaupt keine Daten mehr im EEPROM speichern, da es die Inhalte sofort wieder verliert. Betroffen ist dabei aber jeweils nur eine Speicherseite (übliche Größe: 4 Byte oder 32 Byte) des EEPROMs. Andere Speicherseiten sind vom Ausfall einer Seite nicht betroffen oder werden nicht beeinflusst. Dies kann im Rahmen der Speicherorganisation für Strategien zur Fehlerüberbrückung genutzt werden.

Die Datenablage im EEPROM funktioniert auf der Grundlage der Speicherung elektrischer Ladungen in winzigen Kondensatoren. Wie alle Kondensatoren, haben auch diese einen Leckstrom. Dies führt dazu, dass die im Kondensator enthaltene Ladung im Laufe der Zeit abfließt und die gespeicherte Information nicht mehr verfügbar ist. Durch hohe Temperaturen wird dies noch zusätzlich begünstigt. Deshalb ist der Datenerhalt des EEPROMs nicht unbegrenzt, sondern wird von den Halbleiterherstellern lediglich für 10 Jahre garantiert.

Das Problem bei der zeitlichen Angabe des Datenerhaltes besteht darin, dass die Messung nicht direkt erfolgen kann, da man sonst 10 Jahre warten müsste. Die Entladezeit lässt sich nur über die Ermittlung des Kondensator-Leckstroms berechnen. Da sich auch dies messtechnisch fast nicht durchführen lässt, ermittelt man den maximalen Datenerhalt durch Variation der Umweltparameter wie Temperatur oder Ladespannung und stellt daraus eine Schlussfolgerung auf. Außerdem hängt der Datenerhalt von der Anzahl der Schreib-/Löschzyklen ab. Die spezifizierten 10 Jahre gelten für die maximale Belastung,

² Siehe auch Abschnitt 3.4.2 „Speicherarten“.

sodass in der Regel eine weitaus höhere Lebensdauer erwartet werden kann. Für die Konzeption einer Anwendung bedeutet dies, dass man von 10 Jahren Datenerhalt ausgehen kann und diesen Maximalwert möglichst nicht erreichen sollte.

Eine grundsätzliche Aussage zur Lebensdauer einer Chipkarte ist nur unter Abwägung aller Rahmenbedingungen möglich. Ein praxisnaher Richtwert für die Lebensdauer ist drei bis fünf Jahre für eine übliche Anwendung ohne besondere Anforderungen. In manchen Bereichen, wie z. B. GSM, gehen viele Firmen auch dazu über, die Chipkarten nur bei Bedarf, d. h. bei Ausfall, auszuwechseln, was aus wirtschaftlicher Sicht von Vorteil sein kann.

Datenübertragung

Die Geschwindigkeit der Abwicklung von Kommandos an die Chipkarte hängt neben der internen Ausführungsgeschwindigkeit vor allem von der Datenübertragungsrate der Schnittstelle Terminal – Chipkarte ab.³

Tabelle 15.1 Typische Datenübertragungsraten von Chipkarten.

Funktion	Durchsatz bei 3,5 MHz Takt	Durchsatz bei 4,9 MHz Takt
Übertragung von Daten mit Teiler 372	9 600 Bit/s	13 212 Bit/s
Übertragung von Daten mit Teiler 512	6 975 Bit/s	9 600 Bit/s
Übertragung von Daten mit Teiler 372, T = 0/T = 1	≈ 7 680 Bit/s	≈ 10 570 Bit/s
Übertragung von Daten mit Teiler 372, T = 1 und Secure Messaging (Authentic-Modus)	≈ 3 800 Bit/s	≈ 5 200 Bit/s
Übertragung von Daten mit Teiler 372, T = 1 und Secure Messaging (Combined-Modus)	≈ 1 900 Bit/s	≈ 2 600 Bit/s

Aufgrund der asynchronen seriellen Schnittstelle muss bei der Übertragung von einem Byte dieses auf 12 Bit verlängert sein, da neben den 8 Bit Nutzdaten noch Startbit, Paritätsbit und Stopbits übertragen werden. Damit kann man bei der Datenübertragung mit einem Richtwert von 1,25 ms/Byte bei einer Übertragungsrate von 9600 Bit/s arbeiten. Addiert man noch einen zusätzlichen Aufwand für das zwingend notwendige Übertragungsprotokoll von 20 %, erhält man einen Wert von 1,5 ms/Byte bei der Datenübertragung. Für die Praxis und zur Abschätzung der Dauer von Datenübertragungen ist die dabei erreichte Genauigkeit ausreichend.

Ausführungsgeschwindigkeiten von Algorithmen

Da Chipkarten oft als geschützte Computer zur Abarbeitung von Algorithmen dienen, sind dazu einige typische Kennzahlen angegeben. Diese befinden sich separiert nach Algorithmen in Abschnitt 4.7 „Kryptologie“.

15.1.2 Anwendungen

Neben den spezifischen Belangen des Mikrocontrollers sind bei der Konzeption und Entwicklung einer Chipkarten-Anwendung noch weitere Aspekte zu berücksichtigen. Dies betrifft im Falle der Konzeption vor allem die Gebiete der Schlüsselaufteilung, der Verwaltung von Anwendungsdaten und der grundsätzlichen Prinzipien des Datenaustausches.

³ Siehe auch Abschnitt 6.4 „Übertragungsprotokolle“.

Schlüsselmanagement

In allen Anwendungen, die kryptografische Algorithmen benutzen, steckt die Sicherheit in der Geheimhaltung der entsprechenden Schlüssel. Wird ein geheimer Schlüssel aus irgendeinem Grund bekannt, sind alle darauf aufbauenden Sicherheitsmechanismen anfällig. Da sich dieser Fall prinzipiell nicht mit absoluter Sicherheit ausschließen lässt, muss man entsprechende Vorsorge treffen.⁴

Die trivialste und kostspieligste aller Möglichkeiten ist der Austausch aller Chipkarten, was bei größeren Anwendungen jedoch wirtschaftlich indiskutabel ist. Als Richtwert kann man beim notwendigen Tausch von Chipkarten einen direkten Schaden von 30 Euro pro Chipkarte annehmen. In der Praxis werden deshalb andere Verfahren benutzt, um die Auswirkungen von bekannt gewordenen Schlüsseln zu minimieren.

Grundsätzlich sollten sich in Chipkarten nur kartenindividuelle Schlüssel befinden. Jede Chipkarte hat damit individuelle Schlüssel, die im Falle des Bekanntwerdens nur zum Klonen dieser einen Karte benutzt werden könnten. Der dazugehörige Hauptschlüssel muss aber unter allen Umständen geheim bleiben.

Um auch den Fall der Kompromittierung des Hauptschlüssels abzusichern, kann man mehrere Generationen von kartenindividuellen Schlüsseln in die Karte einbringen, sodass man gegebenenfalls auf eine neue Schlüsselgeneration umschalten kann.

Das dritte Prinzip im Zusammenhang mit Schlüsseln ist das Trennen nach dem Verwendungszweck. In der Praxis bedeutet dies, dass z. B. ein Schlüssel für die Authentisierung nicht zur Verschlüsselung benutzt werden darf. Damit kann man die Auswirkungen, die ein bekannt gewordener Schlüssel verursacht, so gering wie möglich halten.

Die obigen drei Prinzipien sollten bei jeder größeren Anwendung unbedingt Beachtung finden, da sie dem Anwendungsbetreiber ein wesentlich sichereres System garantieren und ihm unter Umständen viel Geld für den Austausch aller Chipkarten ersparen.

Eines darf jedoch in diesem Zusammenhang nicht verschwiegen werden: Jede Einzelne dieser Forderungen führt dazu, dass die Zahl der Schlüssel zunimmt, was sehr schnell zu Speicherproblemen führt und dann auch eine mehr oder minder aufwendige Schlüsselverwaltung erfordert. In der Praxis ist deshalb sehr genau abzuwägen, ob man wirklich alle drei Forderungen erfüllt bzw. ob man nicht bei bestimmten Mechanismen Kompromisse eingeht.

Daten

Bei der Speicherung von Daten in Chipkarten sollten die gleichen Prinzipien eingehalten werden, wie sie im Bereich der Informationstechnik Standard sind. Dies bedeutet für eine Anwendung, dass klassifizierende Nummernsysteme so weit wie möglich zu vermeiden sind, da sie oft schon bei kleinen Änderungen oder Erweiterungen „platzen“. Identifizierende Nummernsysteme sind im Gegensatz dazu jedoch oft zu abstrakt, sodass in der Praxis meist Mischnummern vorherrschen.

Ein gutes Beispiel dafür ist das deutsche Telefonnummernsystem. Der erste Teil der Nummer, die Vorwahl, ist klassifizierend. Man kann bei Kenntnis dieser Nummer genau sagen, in welchem Gebiet der entsprechende Ort liegt. Der zweite Teil, die eigentliche Rufnummer, ist identifizierend, da sie, zumindest bei kleineren Städten, keine Informatio-

⁴ Siehe auch Abschnitt 4.8 „Schlüsselmanagement“.

nen über den genauen Ort des Teilnehmers beinhaltet. Die beiden Teile zusammen ergeben eine typische Mischnummer, die bei Erweiterungen weitgehend problemlos ist.

Um Datenelemente flexibel und in mehreren unterschiedlichen Versionen handhaben zu können, eignen sich sehr gut ASN.1-codierte Datenobjekte.⁵ Ein Richtwert aus der Praxis für den dazu erforderlichen Overhead an Codierungsaufwand bei BER-TLV ist etwa 25 % von den Nutzdaten. Dies gilt nur für kleine Datenmengen, wie sie aber gerade im Chipkartenbereich üblich sind.

Eine grundsätzliche Anmerkung in Bezug auf die Speicherfähigkeit von Chipkarten muss ebenfalls noch gemacht werden. Anwendungen mit einer Größe der Nutzdaten über 10 kByte sind selten, da der zur Verfügung stehende Speicher in der Regel immer sehr knapp ist.

Bei der Planung einer neuen Anwendung ist zumindest überschlagsmäßig der in der Karte zu reservierende Speicher zu berechnen. Dazu benötigt man aber nicht nur die Größe der Nutzdaten, sondern auch den dafür notwendigen Umfang an Verwaltungsdaten. Bei den neueren Betriebssystemen, die mehrere Anwendungen in einer Chipkarte gleichzeitig zulassen und eine objektorientierte Dateiverwaltung besitzen, ist der Anteil an Verwaltungsdaten relativ hoch. Die Größe der Header pro Dateityp bewegt sich dabei meist im Bereich zwischen 16 Byte und 32 Byte. Die Größe dieser Verwaltungsdaten pro Datei ist aber fest, d.h. sie hängt nicht von der Anzahl der in der Datei enthaltenen Nutzdaten ab. Es hat keinen Einfluss auf die Größe des Headers, ob ein Byte oder 200 Byte Nutzdaten vorhanden sind. Dies führt dazu, dass man versucht, nicht für jedes Datenelement eine einzelne Datei anzulegen, da sonst zu viel Speicher für die Verwaltung verschwendet würde.

Datenaustausch

Der Datenaustausch zwischen Chipkarte und Terminal erfordert vor allem die Beachtung zweier Dinge: Die serielle Schnittstelle der Chipkarte ist gemessen an der Leistungsfähigkeit heutiger Rechnersysteme sehr langsam. Die üblichen 9600 Bit/s führen dazu, dass die Datenübertragung zwar sehr robust gegenüber Störungen ist, doch benötigt der Austausch von Daten auch sehr viel Zeit. Deshalb ist es in jedem Fall empfehlenswert, darauf zu achten, dass sich der Datenaustausch zwischen Terminal und Chipkarte auf das Wesentliche beschränkt. Daten, die das Terminal im Laufe einer Sitzung bereits einmal erhalten hat, sollte dieses nicht noch einmal anfordern. Bei Anwendungen, in denen der Zeitbedarf nur sehr untergeordnete Priorität hat, muss dies natürlich nicht beachtet werden. Sollten aber Personen am Ablauf beteiligt sein, erfordert dies auf jeden Fall eine Minimierung des Datenaustausches auf der Schnittstelle.

Im Zusammenhang mit dem Datenaustausch stehen auch die folgenden Punkte: Mit Secure Messaging⁶ lassen sich im Prinzip alle Daten auf der Schnittstelle zwischen Terminal und Chipkarte sehr gut schützen. Ein Angriff über die Schnittstelle ist dadurch so gut wie unmöglich. Doch neben diesem Vorteil sinkt dabei die effektive Übertragungsgeschwindigkeit. Deshalb sollten Daten nur wenn es für die Sicherheit des Systems unumgänglich ist, im Rahmen von Secure Messaging übertragen werden.

Mit Ausnahme der geheimen Schlüssel reicht es für fast alle Nutzdaten, wenn sie mit nur einem MAC geschützt sind (Authentic-Modus). Dies reduziert die Übertragungsge-

⁵ Siehe auch Abschnitt 4.1 „Strukturierung von Daten“.

⁶ Siehe auch Abschnitt 6.6 „Sicherung der Datenübertragung“.

schwindigkeit nicht so stark wie die noch zusätzliche Verschlüsselung im Combined-Modus. Auch ist die Datenübertragung auf der Schnittstelle transparent und kann von außen ohne weiteres überprüft werden. Im Zusammenhang mit dem Datenschutz kann dies unter Umständen notwendig sein, weil sich auf diese Weise jederzeit und ohne Kenntnis von geheimen Schlüsseln überprüfen lässt, welche Daten die Schnittstelle passieren.

15.1.3 System

Sicherheit

In vielen Anwendungen wird der Tripel-DES als kryptografischer Algorithmus verwendet. Die Gründe dafür sind vielschichtig, doch in den meisten Fällen ist wohl ausschlaggebend, dass der DES bzw. seine Kaskadierung der Tripel-DES sehr bekannt ist und sich somit niemand auf unbekanntes Gebiet wagen muss. Jedoch existieren neben dem DES noch viele andere gute Algorithmen zur Ver- und Entschlüsselung von Daten. Verursacht durch den hohen Bekanntheitsgrad, ist der DES auch den meisten Angriffen ausgesetzt, von denen möglicherweise irgendwann einer erfolgreich ist. Deshalb sollte man bei größeren Chipkartenprojekten die Möglichkeit ins Auge fassen, alternativ zum DES einen anderen kryptografischen Algorithmus einzusetzen. Damit lassen sich die Auswirkungen eines erfolgreichen Angriffs auf den DES minimieren. Ein sehr gutes Beispiel für diese Philosophie stellen die aktuellen kryptografischen Algorithmen bei GSM dar, die spezielle Entwicklungen für diese Anwendung und zum Teil völlig unabhängig vom DES sind.

Benutzerschnittstelle

Entscheidend für ein erfolgreiches Chipkartenprojekt ist die Akzeptanz durch die Benutzer. Dies hat zwar sehr viel mit der Mensch-Maschine-Schnittstelle am Terminal zu tun, doch auch die Schnittstelle zwischen Terminal und Chipkarte hat Einfluss darauf. Erfahrungsgemäß kann man davon ausgehen, dass es Probleme mit der Benutzerakzeptanz gibt, wenn eine Aktion mehr als eine Sekunde andauert. Der Benutzer vermutet dann oft einen technischen Defekt und versucht, die Chipkarte aus dem Terminal zu ziehen, was zu einem unkontrollierten Abbruch der Sitzung führt. Um dies zu vermeiden, sollte man alle Abläufe zwischen Terminal und Chipkarte dahingehend optimieren, dass deren Zeitdauer unter einer Sekunde liegt.

Eines darf man vor allem in diesem Bereich der Schnittstelle Mensch-Maschine nicht übersehen; Ist die Akzeptanz unzureichend, ergeben sich auch wesentlich mehr technische Probleme aufgrund nicht vorhersehbarer Eingriffe (z. B. vorzeitiges Herausziehen der Chipkarte aus dem Terminal).

Konzeption

Bei der Konzeption eines informationstechnischen Systems mit Chipkarten sollte das Prinzip von Kerckhoff immer Anwendung finden, d. h. die Sicherheit des Systems beruht allein auf den geheimen Schlüsseln.

Die Prüfung eines Systems ist nämlich sehr schwierig, falls die Sicherheit auf der Geheimhaltung von Informationen beruht, weil sie dann nur von einem sehr eingeschränkten Personenkreis vorgenommen werden kann. Da dieser Personenkreis auch noch meist das System konzipierte, kann eine Prüfung auf dieser Grundlage qualitativ nicht hochwertig

sein. Eine totale Preisgabe aller internen Informationen ist jedoch ebenfalls problematisch, weil ein potenzieller Angreifer alle Informationen frei Haus erhält. Es ist also in der Praxis meist ein Mittelweg sinnvoll, bei dem die grundlegenden Informationen über das System öffentlich sind, spezielle Dinge aber geheim.

Müssen, ganz im Gegensatz zu Kerckhoff, Sachverhalte innerhalb eines Chipkartenprojektes vertraulich bleiben, dann bietet sich an, die Sicherheit nicht nur auf eine einzelne Person zu monopolisieren, sondern das Wissen auf mehrere Personen zu verteilen, von denen jede Einzelne nur einen Teil des Systems kennt, aber keine das ganze.

Im Zusammenhang mit der Konzeption von auf Chipkarten basierenden Systemen muss immer das Gesamtsystem im Auge behalten werden und nicht nur die Chipkarte und deren unmittelbare Systemumgebung. Dies ist der häufigste Fehler, der im Laufe vieler Projekte zu beobachten ist. Es handelt sich aufgrund des aktiven Charakters der Komponente Chipkarte immer um verteilte Systeme, die selbstständig agieren müssen und in denen kein reines Master-Slave-Verhalten vorherrscht. Nach der Systemintegration muss das gesamte System reibungslos zusammenarbeiten und nicht nur die Chipkarte mit dem Terminal. Deshalb ist es schon sehr frühzeitig bei der Projektierung notwendig, eine gesamtheitliche Betrachtungsweise zu gewinnen.

15.1.4 Normenkonformität

Chipkarten sind wie nur sehr wenige andere Komponenten der gesamten Informationstechnik stark an Normen orientiert. Dies ergibt sich zum einen aus der Notwendigkeit der Kompatibilität vieler verschiedener Chipkarten- und Terminal-Hersteller untereinander und zum anderen aus der erforderlichen Interoperabilität der verschiedenen Chipkartensystemen zueinander. Im Folgenden sind deshalb eine Reihe von allgemeinen internationalen Normen angegeben, die üblicherweise von heutigen Chipkarten-Betriebssystemen unterstützt werden sollten bzw. zu denen diese kompatibel sein sollten.⁷ Für tatsächliche Anwendungen sind dann selbstverständlich noch die entsprechenden nationalen Normen und Spezifikationen zu beachten.

Die international wichtigste Normenreihe ist die ISO/IEC 7816. Sie legt die wesentlichen physikalischen und informationstechnischen Rahmenbedingungen fest. Hinsichtlich Dateisystem, Kommandos und Dateizugriffsbedingungen sind dabei die Teile 4, 8 und 9 von großer Bedeutung. Dabei sollte jedoch bedacht werden, dass diese Normenreihe eine Art Baukasten festlegt, innerhalb dessen man bestimmte Dinge auswählen kann, d. h. nicht alle Punkte der Norm müssen jeweils von einer Chipkarte vollständig in allen Punkten erfüllt werden.

Im Bereich Telekommunikation ist die Spezifikation TS 102.221 für die UICC ein zentrales Dokument. Des Weiteren sind die Spezifikationen GSM 11.11 für SIM und TS 31.102 für USIM repräsentative Beispiele für weltweite Telekommunikations-Anwendungen.

Im Bereich Zahlungsverkehr ist die EMV-Spezifikation ein Faktum, an dem sich alle anderen Spezifikationen orientieren. Hinsichtlich elektronischer Geldbörsen ist einerseits EN 1546 eine Art Grundlagennorm und andererseits CEPS das exemplarische Beispiel für eine internationale Geldbörse.

⁷ Ein kommentiertes Verzeichnis der Chipkartennormen findet sich in Abschnitt 16.5 „Kommentiertes Normen- und Spezifikationsverzeichnis“.

Chipkarten-Betriebssysteme mit ausführbarem Programmcode reduzieren sich zurzeit praktisch vollständig auf Java Card und die dazugehörigen Spezifikationen. Beim Nachladen von Anwendungen wird beinahe ausschließlich die Open Platform Spezifikation herangezogen.

Chipkarten im Umfeld der digitalen Signatur basieren häufig auf der PKCS #15-Spezifikation. Zertifikate werden in diesem Zusammenhang oftmals in X.509-konformen Formaten gespeichert.

15.2 Formelsammlung zur Abschätzung von Ausführungszeiten

Bei der Konzeption von Chipkarten-Anwendungen oder dem Design von neuen Chipkarten-Kommandos müssen oft Abschätzungen über den Zeitbedarf gemacht werden. Selbst mithilfe von Erfahrungswerten ist es relativ schwierig und ziemlich ungenau, durch Schätzungen ausreichend genaue Zahlen zu erhalten. Im folgenden Abschnitt sind die grundlegenden Formeln zur Berechnung von Ausführungszeiten bei Chipkarten-Betriebssystemen und kontaktbehafteter Übertragung zusammengefasst. Richtig angewandt, führen sie zu Ergebnissen in akzeptabler Genauigkeit. Sie sollten aber keinesfalls als unumstößlich und von unbeschränkter Genauigkeit angesehen werden, da sie für rechnerische Abschätzungen gedacht sind, bei denen sich durchaus Fehler in der Größenordnung von zehn Prozent einstellen können. Deshalb sei hier auch gleich der Ratschlag gegeben, in kritischen Fällen einen entsprechenden Sicherheitsaufschlag hinzuzurechnen.

Die Variablennamen in den aufgeführten Formeln sollten selbsterklärend sein, deshalb sind relativ lange Namen notwendig. So weit nicht explizit angegeben, sind alle festen Zeiten auf die Standardtaktfrequenz von 3,5712 MHz ausgelegt. Über den Proportionalitätsfaktor PF aus den Formeln (15.9) und (15.10) können die vorgegebenen Zeiten aber für alle anderen Taktfrequenzen umgerechnet werden. Es wird in allen Fällen angenommen, dass die jeweiligen Kommandos fehlerfrei ausgeführt werden und sowohl bei der Datenübertragung als auch bei eventuell stattfindenden EEPROM-Operationen keine Fehler auftreten. Bei der Datenübertragung wurde des Weiteren angenommen, dass sofort nach Empfang des letzten Bits eines Kommandos mit der Abarbeitung begonnen wird, d. h. dass eine möglichst kleine Zeichenwartezeit CWT gewählt wurde.

Kommandoabarbeitung

Die nachfolgend aufgeführten Formeln sind die Grundlage für alle Zeitberechnungen. Sie trennen die Gesamtzeit in die beiden Zeiten für Datentransfer und eigentliche Kommandoabarbeitung auf.

$$t_{\text{gesamt}} = t_{\text{Datentransfer}} + t_{\text{ICC intern}} \quad (15.1)$$

$$t_{\text{Datentransfer}} = t_{\text{Datentransfer Kommando}} + t_{\text{Datentransfer Antwort}} \quad (15.2)$$

$$t_{\text{ICC intern}} = t_{\text{Kommandointerpreter}} + t_{\text{Kommandoabarbeitung}} \quad (15.3)$$

Die Zeit, die der Kommandointerpreter für seine Aktionen benötigt, ist nur von der angelegten Taktfrequenz abhängig und hier für 3,5712 MHz angegeben.

$$t_{\text{Kommandointerpreter}} \approx 1,5 \text{ ms} \tag{15.4}$$

$$t_{\text{Kommandoabarbeitung}} = t_{\text{EEPROM}} + t_{\text{Kryptoalgorithmus}} + t_{\text{Kommandocode}} \tag{15.5}$$

Die exakte Abarbeitungszeit eines Kommandos lässt sich nur durch detaillierte Analyse des Programmablaufs auf Maschinencodeebene ermitteln. Verzweigungen und Schleifen im Programm sowie unterschiedliche Kommandooptionen würden zu höchst komplexen Formeln führen, die für die Praxis unbrauchbar sind. Deshalb wird hier eine einfache Dreiteilung des Zeitbedarfs in Abhängigkeit von der Komplexität des Kommandos verwendet. Am wenigsten Zeit benötigen einfache Kommandos, wie beispielsweise SELECT FILE oder READ BINARY. Kommandos mittlerer Komplexität (z. B.: INTERNAL AUTHENTICATE) benötigen für ihren internen Programmablauf schon etwas mehr Zeit, und Kommandos mit hoher Komplexität (z. B.: DEBIT IEP) beanspruchen am meisten Zeit für die interne Abarbeitung. Dabei darf nicht übersehen werden, dass diese pauschalierten Werte nicht etwa die Aufrufe von Kryptoalgorithmen oder EEPROM-Schreibvorgängen abdecken, sondern nur die notwendigen internen Berechnungen und Abfragen.

$$t_{\text{Kommandocode}} \approx 5 \text{ ms (für Kommandos geringer Komplexität)} \tag{15.6}$$

$$t_{\text{Kommandocode}} \approx 12 \text{ ms (für Kommandos mittlerer Komplexität)} \tag{15.7}$$

$$t_{\text{Kommandocode}} \approx 20 \text{ ms (für Kommandos großer Komplexität)} \tag{15.8}$$

Tabelle 15.2 Definition und Beschreibung der Variablen für die Formeln (15.1) bis (15.8).

Variable	Einheit	Beschreibung
$t_{\text{Datentransfer}}$	s	Zeitdauer für die Datenübertragung eines Kommandos und der dazugehörigen Antwort.
$t_{\text{Datentransfer Antwort}}$	s	Zeitdauer für die Datenübertragung einer Antwort auf ein Kommando.
$t_{\text{Datentransfer Kommando}}$	s	Zeitdauer für die Datenübertragung eines Kommandos.
t_{EEPROM}	s	Zeitdauer für das Schreiben von Daten in das EEPROM.
t_{gesamt}	s	Zeitdauer für den Empfang eines Kommandos, das Abarbeiten und Senden der Antwort.
$t_{\text{ICC intern}}$	s	Zeitdauer für die interne Verarbeitung eines Kommandos in der Chipkarte.
$t_{\text{Kommandoabarbeitung}}$	s	Zeitdauer für das Abarbeiten eines Kommandos.
$t_{\text{Kommandocode}}$	s	Zeitdauer für die Abarbeitung eines speziellen Programms für ein bestimmtes Kommando (z. B. eines Kryptoalgorithmus).
$t_{\text{Kommandointerpreter}}$	s	Zeitdauer für die Analyse des Kommandos und den Aufruf des dazugehörigen Programmcodes.
$t_{\text{Kryptoalgorithmus}}$	s	Zeitdauer für die Ausführung eines kryptografischen Algorithmus.

Proportionalitätsfaktor für vordefinierte Funktionen

Sind bereits Zeitangaben für Funktionen und für eine bestimmte Taktfrequenz vorhanden, dann müssen diese bei Bedarf über einen Proportionalitätsfaktor auf die aktuell verwendete Taktfrequenz umgerechnet werden.

$$PF = \frac{f_{\text{Referenz}}}{f_{\text{aktuell}}} \quad (15.9)$$

$$t_{\text{aktuell}} = t_{\text{Referenz}} \cdot PF \quad (15.10)$$

Tabelle 15.3 Definition und Beschreibung der Variablen für die Formel (15.9) und (15.10).

Variable	Einheit	Beschreibung
f_{aktuell}	MHz	Aktuelle Taktfrequenz.
f_{Referenz}	MHz	Frequenz, für die eine bestimmte taktfrequenzabhängige Zeitdauer angegeben ist.
t_{aktuell}	s	Aktuelle Zeitdauer für eine Aktion.
t_{Referenz}	s	Zeitdauer, für die eine bestimmte taktfrequenzabhängige Zeit angegeben ist.
PF	---	Proportionalitätsfaktor bei taktfrequenzabhängigen Programmroutinen.

EEPROM-Operationen

Um Daten ins EEPROM zu schreiben, muss dieses je nach Inhalt der zu überschreibenden Daten vorher gelöscht werden. Die beim Löschen benutzte Pagegröße kann bei manchen Chipkarten-Mikrocontrollern unterschiedlich zur Pagegröße beim Schreibvorgang sein. Die folgenden Formeln berücksichtigen dies.

Um festzustellen, ob eine EEPROM-Page vorher gelöscht werden muss, ist es notwendig, den aktuellen Dateninhalt und die neuen Daten zu kennen. Konservativ gerechnet sollte man aber annehmen, dass die zu beschreibenden Pages immer vorher gelöscht werden müssen.

Zur Schreib- und Löschdauer von EEPROM-Operationen sei an dieser Stelle noch ein kleiner Hinweis angebracht. Die zurzeit gebräuchlichen Mikrocontroller für Chipkarten besitzen keine interne Zeitbasis, sodass einzig und allein die angelegte Taktfrequenz als Referenz vom Betriebssystem verwendet wird. Hat ein Chipkarten-Mikrocontroller nun eine maximale zulässige Taktfrequenz von beispielsweise 5 MHz, werden alle EEPROM-Schreibroutinen für diese Maximalfrequenz ausgelegt. Dies führt aber dazu, dass die Schreibdauer bei einer geringeren Frequenz als der maximalen proportional länger ist. Für genaue Berechnungen muss man dies über einen Proportionalitätsfaktor berücksichtigen. Da dies jedoch vom Typ des Mikrocontrollers und auch von der im Chipkarten-Betriebssystem parametrisierten maximalen Taktfrequenz abhängt, wurde dieser mögliche Effekt hier nicht berücksichtigt. In Zukunft wird er ohnehin an Bedeutung verlieren, da die neuesten Mikrocontroller über eine interne Zeitbasis verfügen und so die EEPROM-Operationen immer mit der gleichen Zeit und unabhängig von der angelegten Taktfrequenz durchgeführt werden.

$$t_{\text{EEPROM}} = t_{\text{EEPROM löschen}} + t_{\text{EEPROM schreiben}} \tag{15.11}$$

$$t_{\text{EEPROM löschen}} = \frac{t_{\text{Page löschen}}}{\text{PG}_{\text{löschen}}} \cdot n \tag{15.12}$$

$$t_{\text{EEPROM schreiben}} = \frac{t_{\text{Page schreiben}}}{\text{PG}_{\text{schreiben}}} \cdot n \tag{15.13}$$

Tabelle 15.4 Definition und Beschreibung der Variablen und Formeln (15.11) bis (15.13) für EEPROM-Operationen.

Variable	Einheit	Beispiel	Beschreibung
n	Byte	20 Byte	Anzahl der ins EEPROM zu schreibenden Bytes. Der Wert muss auf die jeweilige Pagegröße aufgerundet werden.
$\text{PG}_{\text{löschen}}$	Byte	32 Byte	Pagegröße für das Löschen.
$\text{PG}_{\text{schreiben}}$	Byte	4 Byte	Pagegröße für das Schreiben.
t_{EEPROM}	s	21 ms	Zeitdauer, um n Byte Daten in das EEPROM zu schreiben und dieses bei Bedarf vorher zu löschen.
$t_{\text{EEPROM löschen}}$	s	3,5 ms	Zeitdauer zum Löschen von n Byte im EEPROM.
$t_{\text{EEPROM schreiben}}$	s	17,5 ms	Zeitdauer zum Schreiben von n Byte im EEPROM.
$t_{\text{Page löschen}}$	s/Byte	3,5 ms/4 Byte	Zeitdauer zum Löschen einer EEPROM-Page.
$t_{\text{Page schreiben}}$	s/Byte	3,5 ms/4 Byte	Zeitdauer zum Schreiben einer EEPROM-Page.

Datenübertragung

Die Zeitdauer, die für die Übertragung von Kommando und nachfolgender Antwort notwendig ist, hängt in erster Linie von der Anzahl der zu übertragenden Daten ab. Der Aufbau von den zugrunde liegenden TPDU (*transport protocol data unit*) und APDU (*application protocol data unit*) ist in den Abschnitten 6.4.2 „Übertragungsprotokoll T = 0“ und 6.4.3 „Übertragungsprotokoll T = 1“ im Detail beschrieben.

$$t_{\text{Kommando}} = t_{\text{Bytetransfer}} \cdot n_{\text{Kommandodaten}} \tag{15.14}$$

$$t_{\text{Antwort}} = t_{\text{Bytetransfer}} \cdot n_{\text{Antwortdaten}} \tag{15.15}$$

$$n_{\text{Kommandodaten}} = n_{\text{Schicht 2}} + n_{\text{Header}} + n_{\text{Body}} \tag{15.16}$$

$$n_{\text{Antwortdaten}} = n_{\text{Schicht 2}} + n_{\text{Trailer}} + n_{\text{Body}} \tag{15.17}$$

Tabelle 15.5 Definition und Beschreibung der Variablen für die Formeln (15.14) bis (15.17).

Variable	Einheit	typ. Wert	Beschreibung
$n_{\text{Antwortdaten}}$	Byte	---	Anzahl der übertragenen Daten.
n_{Body}	Byte	---	Anzahl der Bytes des Kommando-Bodies bzw. des dazugehörigen Antwort-Bodies. Falls im Kommando ein Body vorhanden ist, enthält er auch eine 1 oder 2 Bytes große Längenangabe für Kommando-Body bzw. Antwort-Body.
n_{Header}	Byte	4	Anzahl der Bytes für den Header des Kommandos. Für das T = 1 Protokoll sind dies die Bytes CLA, INS, P1 und P2.
$n_{\text{Kommandodaten}}$	Byte	---	Anzahl der übertragenen Daten.
$n_{\text{Schicht 2}}$	Byte	4	Anzahl der Bytes für die Übertragungsschicht 2. Für das T = 1 Protokoll sind dies die Bytes NAD, PCB, LEN, EDC.
n_{Trailer}	Byte	2	Anzahl der für den Trailer einer Antwort notwendigen Daten. Dies sind die beiden Bytes SW1 und SW2.
t_{Antwort}	s	---	Dauer der Übertragung für die Antwort auf ein Kommando.
$t_{\text{Bytetransfer}}$	s	---	Dauer der Übertragung für ein Byte.
t_{Kommando}	s	---	Dauer der Übertragung für ein Kommando.

Tabelle 15.6 Definition und Beschreibung der Variablen für die Formel (15.18).

Variable	Einheit	typ. Wert	Beschreibung
D	Bit · Byte/MHz · ms	1 Bit · Byte /MHz · ms	Bit Rate Adjustment Factor
f	MHz	3,5712 MHz	Taktfrequenz
F	1	372	clock rate conversion factor
n	Bit	12 Bit	Anzahl von Bits pro Byte (1 Startbit, 8 Datenbits, 1 Paritätsbit, 2 Stopbits)
$t_{\text{Bytetransfer}}$	ms	1,25 ms	Dauer der Übertragung für ein Byte

$$t_{\text{Bytetransfer}} = \frac{1}{D} \cdot \frac{F}{f} \cdot n \quad (15.18)$$

Rechenbeispiel: Chipkarten-Kommando READ BINARY

Im Folgenden soll eine rechnerische Abschätzung für den Zeitbedarf eines einfach aufgebauten Chipkarten-Kommandos gemacht werden. Für die Berechnung wird das Kommando READ BINARY gewählt. Als Rahmenbedingungen seien das Übertragungsprotokoll T = 1 mit 372 als Teiler und eine Taktfrequenz von 5 MHz gewählt. Die Dauer für die Übertragung eines Bytes berechnet sich aus der Formel (15.18).

$$t_{\text{Bytetransfer}} = \frac{1}{D} \cdot \frac{F}{f} \cdot n = \frac{1}{1} \cdot \frac{372}{5000000} \cdot 12 \text{ Bit} \approx 0,89 \frac{\text{ms}}{\text{Byte}}$$

Für die Übertragung mit T = 1 werden für Schicht 2 zusätzlich vier Bytes zu den Schicht-7-Daten benötigt. Der Kommando-Header von READ BINARY hat eine Länge von 4 Bytes

(CLA, INS, P1, P2), der dazugehörige Body eine Länge von einem Byte (Le), und die Antwort besteht aus den gelesenen Daten der Länge Le und der Trailer aus den Statuswörtern SW1 und SW2. Damit lässt sich aus den Formeln (15.16) und (15.17) die Anzahl der zu übertragenden Daten für Kommando und Antwort in Abhängigkeit von Le errechnen.

$$n_{\text{Kommandodaten}} = n_{\text{Schicht 2}} + n_{\text{Header}} + n_{\text{Body}}$$

$$n_{\text{Kommandodaten}} = (4 + 4 + 1) \text{ Byte} = 9 \text{ Byte}$$

$$n_{\text{Antwortdaten}} = n_{\text{Schicht 2}} + n_{\text{Body}} + n_{\text{Trailer}}$$

$$n_{\text{Antwortdaten}} = (4 + \text{Le} + 2) \text{ Byte} = (\text{Le} + 6) \text{ Byte}$$

Daraus kann nun mithilfe der Formeln (15.14) und (15.15) die jeweilige Zeitdauer für die Übertragung von Kommando und Antwort ermittelt werden.

$$t_{\text{Kommando}} = t_{\text{Bytetransfer}} \cdot n_{\text{Kommandodaten}} = 0,89 \frac{\text{ms}}{\text{Byte}} \cdot 9 \text{ Byte} = 8,01 \text{ ms}$$

$$t_{\text{Antwort}} = t_{\text{Bytetransfer}} \cdot n_{\text{Antwortdaten}}$$

$$t_{\text{Antwort}} = 0,89 \frac{\text{ms}}{\text{Byte}} \cdot (\text{Le} + 6 \text{ Byte}) = 0,89 \cdot (\text{Le} + 6 \text{ Byte}) \frac{\text{ms}}{\text{Byte}}$$

Das Kommando ist von geringer Komplexität, weshalb eine Ausführungszeit von 5 ms bei 3,5712 MHz angenommen wird. Daraus ergibt sich nach Umrechnung auf 5 MHz Taktfrequenz (Formeln (15.9) und (15.10)) Folgendes:

$$\text{PF} = \frac{f_{\text{Referenz}}}{f_{\text{aktuell}}} = \frac{3,5712 \text{ MHz}}{5 \text{ MHz}} = 0,714$$

$$t_{\text{aktuell}} = t_{\text{Referenz}} \cdot \text{PF} = 5 \text{ ms} \cdot 0,714 = 3,57 \text{ ms}$$

Für READ BINARY müssen weder Daten ins EEPROM geschrieben werden, noch ist es notwendig, einen Kryptoalgorithmus aufzurufen. Deshalb errechnet sich die Zeitdauer für die interne Kommandoabarbeitung wie folgt:

$$t_{\text{Kommandoabarbeitung}} = t_{\text{EEPROM}} + t_{\text{Kryptoalgorithmus}} + t_{\text{Kommandocode}}$$

$$t_{\text{Kommandoabarbeitung}} = 0 \text{ ms} + 0 \text{ ms} + 3,57 \text{ ms} = 3,57 \text{ ms}$$

Unter der zusätzlichen Annahme, dass der Kommandointerpreter für seine Funktion etwa 1,5 ms bei 3,5712 MHz benötigt, errechnet sich die interne Ausführungszeit des Kommandos.

$$t_{\text{aktuell}} = t_{\text{Referenz}} \cdot \text{PF} = 1,5 \text{ ms} \cdot 0,714 \approx 1 \text{ ms}$$

$$t_{\text{ICC intern}} = t_{\text{Kommandointerpreter}} + t_{\text{Kommandoabarbeitung}} = 1 \text{ ms} + 3,5 \text{ ms} = 4,5 \text{ ms}$$

Nun können alle ermittelten Werte in die Formel (15.2) eingesetzt werden, und es ergibt sich eine Formel für die Dauer des READ BINARY-Kommandos in Abhängigkeit der gelesenen Daten.

$$t_{\text{Datentransfer}} = t_{\text{Datentransfer Kommando}} + t_{\text{Datentransfer Antwort}}$$

$$t_{\text{Datentransfer}} = 8,01 \text{ ms} + 0,89 \cdot (\text{Le} + 6 \text{ Byte}) \frac{\text{ms}}{\text{Byte}}$$

$$t_{\text{gesamt}} = t_{\text{Datentransfer}} + t_{\text{ICC intern}} = 8,01 \text{ ms} + 0,89 \cdot (\text{Le} + 6 \text{ Byte}) \frac{\text{ms}}{\text{Byte}} + 4,5 \text{ ms}$$

$$t_{\text{gesamt}} = \left(12,51 + 0,89 \cdot (\text{Le} + 6) \frac{1}{\text{Byte}} \right) \text{ms} = \left(17,85 + 0,89 \cdot \frac{\text{Le}}{\text{Byte}} \right) \text{ms}$$

Dieses rechnerische Ergebnis stimmt größenordnungsmäßig relativ gut mit der empirisch ermittelten Formel (15.21) für READ BINARY für 3,5712 MHz im folgenden Abschnitt überein. Die Differenzen kommen durch zeitliche Abweichungen bei der Datenübertragung und die oben getroffenen Annahmen über den Zeitbedarf für betriebssysteminterne Abläufe zustande.

Rechenbeispiel: Initialisierung einer Chipkarte

In diesem Zahlenbeispiel soll überschlagsmäßig die Dauer der Initialisierung einer Chipkarte berechnet werden. Für die Initialisierung müssen beispielsweise 5 kByte Daten in das EEPROM geschrieben werden. Die Initialisierungsdaten werden mit dem Protokoll T = 1 und einem Teiler von 372 übertragen. Die Taktfrequenz beträgt 3,5712 MHz.

Aus Formel (15.18) errechnet sich die Zeitdauer für die Übertragung eines einzelnen Bytes:

$$t_{\text{Bytetransfer}} = \frac{1}{D} \cdot \frac{F}{f} \cdot n = \frac{1}{1 \frac{\text{Bit} \cdot \text{Byte}}{\text{MHz} \cdot \text{ms}}} \cdot \frac{372}{3,5712 \text{ MHz}} \cdot 12 \text{ Bit} \approx 1,25 \frac{\text{ms}}{\text{Byte}}$$

Unter der Annahme, dass das Kommando zur Initialisierung einen Header von 4 Byte Länge besitzt (CLA, INS, P1, P2), die Längenangabe im Body ein Byte lang ist (Lc), 100 Byte Nutzdaten übertragen werden und die Antwort nur aus SW1 und SW2 besteht, errechnet sich die Anzahl der zu übertragenden Daten für Kommando und Antwort.

$$n_{\text{Kommandodaten}} = n_{\text{Schicht 2}} + n_{\text{Header}} + n_{\text{Body}}$$

$$n_{\text{Kommandodaten}} = (4 + 4 + 1 + 100) \text{ Byte} = 109 \text{ Byte}$$

$$n_{\text{Antwortdaten}} = n_{\text{Schicht 2}} + n_{\text{Body}} + n_{\text{Trailer}} = (4 + 0 + 2) \text{ Byte} = 6 \text{ Byte}$$

Daraus kann nun wiederum die Zeitdauer für die Übertragung von Kommando und Antwort ermittelt werden.

$$t_{\text{Kommando}} = t_{\text{Bytetransfer}} \cdot n_{\text{Kommandodaten}} = 1,25 \frac{\text{ms}}{\text{Byte}} \cdot 109 \text{ Byte} = 136,25 \text{ ms}$$

$$t_{\text{Antwort}} = t_{\text{Bytetransfer}} \cdot n_{\text{Antwortdaten}} = 1,25 \frac{\text{ms}}{\text{Byte}} \cdot 6 \text{ Byte} = 7,5 \text{ ms}$$

Es müssen 100 Byte Daten ins EEPROM geschrieben werden. Mit der zusätzlichen Annahme, dass eine EEPROM-Page 4 Byte groß ist und eine Schreibdauer von 3,5 ms per

Page besitzt, lässt sich der zeitliche Aufwand für das Schreiben ins EEPROM je Kommando ermitteln.

$$t_{\text{EEPROM schreiben}} = \frac{t_{\text{Page schreiben}}}{\text{PG}_{\text{schreiben}}} \cdot n = \frac{3,5 \frac{\text{ms}}{4 \text{ Byte}}}{4 \text{ Byte}} \cdot 25 = 3,5 \text{ ms} \cdot 25 = 87,5 \text{ ms}$$

Des Weiteren wird angenommen, dass das EEPROM vor dem Schreibvorgang nicht gelöscht werden muss, da dies bereits beim Test des Mikrocontrollers geschehen ist.

$$t_{\text{EEPROM}} = t_{\text{EEPROM löschen}} + t_{\text{EEPROM schreiben}} = 0 \text{ ms} + 87,5 \text{ ms} = 87,5 \text{ ms}$$

Für die Initialisierung ist kein Aufruf eines Kryptoalgorithmus notwendig, und das dafür eingesetzte Kommando ist intern einfach aufgebaut. Deshalb kann man für den Kommandocode eine Ausführungszeit von zirka 5 ms annehmen.

$$t_{\text{Kommandoabarbeitung}} = t_{\text{EEPROM}} + t_{\text{Kryptoalgorithmus}} + t_{\text{Kommandocode}}$$

$$t_{\text{Kommandoabarbeitung}} = 87,5 \text{ ms} + 0 \text{ ms} + 5 \text{ ms} = 92,5 \text{ ms}$$

Nun müssen nur noch die errechneten Werte in die Formeln (15.3) und (15.2) eingesetzt werden, und dann ist die Berechnung des Zeitbedarfs für ein Initialisierungskommando mit 100 Byte Initialisierungsdaten abgeschlossen. Der Kommandointerpreter benötigt dabei zusätzliche 1,5 ms.

$$t_{\text{ICC intern}} = t_{\text{Kommandoabarbeitung}} + t_{\text{Kommandointerpreter}} = 92,5 \text{ ms} + 1,5 \text{ ms} = 94 \text{ ms}$$

$$t_{\text{Datentransfer}} = t_{\text{Datentransfer Kommando}} + t_{\text{Datentransfer Antwort}}$$

$$t_{\text{Datentransfer}} = 136,5 \text{ ms} + 7,5 \text{ ms} = 144 \text{ ms}$$

$$t_{\text{gesamt}} = t_{\text{Datentransfer}} + t_{\text{ICC intern}} = 144 \text{ ms} + 94 \text{ ms} = 238 \text{ ms}$$

Damit hat man berechnet, dass es 238 ms dauert, um 100 Byte Daten zur Chipkarte zu übertragen, in das EEPROM zu schreiben und dann dem Terminal eine Bestätigung der erfolgreichen Kommandoabarbeitung zukommen zu lassen.

Laut Vorgabe müssen jedoch 5 kByte (= 5120 Byte) Daten in den Speicher geschrieben werden. Dazu benötigt man vereinfachend etwa die 52-fache Zeitdauer.

$$t_{\text{Initialisierung}} = t_{\text{gesamt}} \cdot 52 = 238 \text{ ms} \cdot 52 = 12,4 \text{ s} \approx 13 \text{ s}$$

Der Initialisierungsvorgang dauert somit nach obigem Rechenweg 12,4 Sekunden. Unter Zugabe eines kleinen Sicherheitszuschlags kann man daher davon ausgehen, dass die Initialisierung die 13-Sekunden-Marke nicht überschreiten wird. Sollten jedoch beim Initialisierungsvorgang Übertragungs- oder EEPROM-Schreibfehler auftreten, ist dies in der Rechnung nicht berücksichtigt. So etwas muss man in der Praxis jedoch als eine Form von „höherer Gewalt“ ansehen, das rechnerisch nur durch statistische Abschätzungen greifbar ist.

15.3 Zeitfunktionen typischer Chipkarten-Kommandos

Die folgenden Formeln beruhen auf Zeitmessungen an echten Chipkarten-Betriebssystemen und anschließender Berechnung von linearen Gleichungen. Die Grundlage für alle Formeln ist ein Chipkarten-Betriebssystem mit T = 1 Übertragungsprotokoll, einem Clock Rate Conversion Factor von 372 und einem Bit Rate Adjustment Factor (D) von eins. Der Mikrocontroller wird mit 3,5712 MHz getaktet, und die Schreib-/Löschzeit des EEPROM beträgt 3,5 ms für eine 4 Byte große Page. Es wird auch grundsätzlich davon ausgegangen, dass während der Datenübertragung und eines eventuell notwendigen EEPROM-Schreib-/Löschvorgangs keine Fehler auftreten. Die Formeln sind gültig für alle n zwischen 1 und 254.

Die Formeln (15.19) und (5.20) sind zur Zeitberechnung des Kommandos SELECT FILE mit der Option Selektion mit 2 Byte langer FID oder einem n Byte langen DF Namen gedacht.

$$t_{\text{gesamt SELECT FILE mit FID}} \approx 23 \text{ ms}$$

(15.19)

$$t_{\text{gesamt SELECT FILE mit DFName}} \approx (20,75 + 1,26 \cdot n) \text{ ms}$$

(15.20)

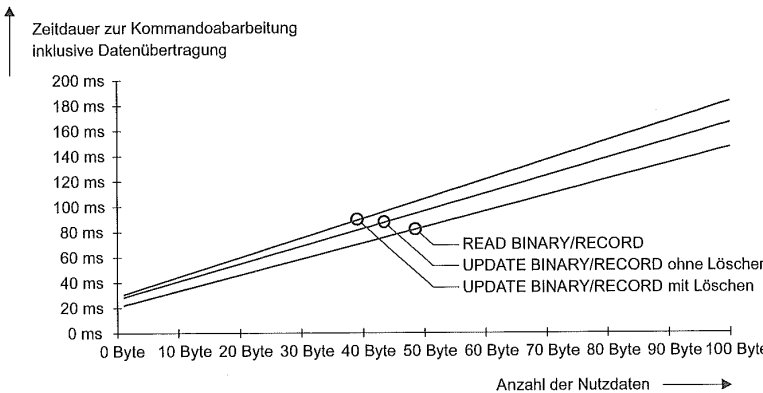


Bild 15.2 Zeitbedarf für Chipkarten-Kommandos in Abhängigkeit der Nutzdaten inklusive der dafür notwendigen Datenübertragung für Kommando und Antwort. Folgende Basis wurde für die Erstellung der Tabelle benutzt: Übertragungsprotokoll T = 1, Teiler 372, Taktfrequenz der Chipkarte 3,5712 MHz, 3,5 ms/4 Byte EEPROM-Schreib-/Löschzeit.

Die Formeln (15.21) bis (15.24) lassen sich zur Abschätzung des Zeitbedarfs für Lesekommandos für Dateien mit transparenter und Record-orientierter Struktur benutzen. Die Variable n ist dabei die Anzahl der zu lesenden Daten in der Einheit Byte. Sollten ein READ BINARY oder READ RECORD mit impliziter Dateiselektion verwendet werden, so sind auch diese Formeln brauchbar, da die durch den impliziten Selektionsvorgang entstehende Zeitabweichung vernachlässigbar ist.

$$t_{\text{gesamt READ BINARY}} \approx (20,77 + 1,26 \cdot n) \text{ ms} \quad (15.21)$$

$$t_{\text{ICC intern READ BINARY}} \approx (2,02 + 0,01 \cdot n) \text{ ms} \quad (15.22)$$

$$t_{\text{gesamt READ RECORD}} \approx (20,70 + 1,26 \cdot n) \text{ ms} \quad (15.23)$$

$$t_{\text{ICC intern READ RECORD}} \approx (1,95 + 0,01 \cdot n) \text{ ms} \quad (15.24)$$

Die Formeln (15.25) bis (15.32) können für die rechnerische Abschätzung des Kommandos UPDATE BINARY und UPDATE RECORD mit und ohne implizite Dateiselektion benutzt werden. Die Zeitdauer hängt vor allem davon ab, ob es notwendig ist, vor dem EEPROM-Schreibvorgang die entsprechenden EEPROM-Pages zu löschen, weshalb die Formeln diesbezüglich aufgeteilt sind. Bei Formeln, die ein vorheriges Löschen berücksichtigen, wurde immer davon ausgegangen, dass alle Pages gelöscht werden müssen. Die Anzahl der zu schreibenden Daten wird mit n in der Einheit Byte angegeben.

$$t_{\text{gesamt UPDATE BINARY ohne Löschen}} \approx (25,55 + 1,39 \cdot n) \text{ ms} \quad (15.25)$$

$$t_{\text{ICC intern UPDATE BINARY ohne Löschen}} \approx (6,8 + 0,14 \cdot n) \text{ ms} \quad (15.26)$$

$$t_{\text{gesamt UPDATE BINARY mit Löschen}} \approx (27,26 + 1,54 \cdot n) \text{ ms} \quad (15.27)$$

$$t_{\text{ICC intern UPDATE BINARY mit Löschen}} \approx (8,51 + 0,29 \cdot n) \text{ ms} \quad (15.28)$$

$$t_{\text{gesamt UPDATE RECORD ohne Löschen}} \approx (25,35 + 1,38 \cdot n) \text{ ms} \quad (15.29)$$

$$t_{\text{ICC intern UPDATE RECORD ohne Löschen}} \approx (6,7 + 0,14 \cdot n) \text{ ms} \quad (15.30)$$

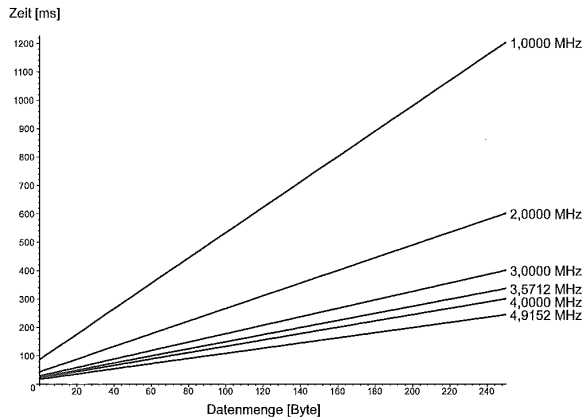


Bild 15.3 Diagramm zur Ermittlung der Übertragungszeit von Daten in Abhängigkeit vom Takt. Die dargestellten Werte beruhen auf folgenden Annahmen: Übertragungsprotokoll T = 1, Teiler 372, keine zusätzliche Schutzzeit (N = 12), kein Chaining, Prüfsummenverfahren XOR, keine Übertragungsfehler und Kommandos im Case 2 oder 3 (z. B.: READ BINARY oder UPDATE BINARY).

$$t_{\text{gesamt UPDATE RECORD mit Löschen}} \approx (27,13 + 1,5 \cdot n) \text{ ms} \quad (15.31)$$

$$t_{\text{ICC intern UPDATE RECORD mit Löschen}} \approx (8,4 + 0,28 \cdot n) \text{ ms} \quad (15.32)$$

Unter Verwendung obiger Formeln wurden die folgenden Diagramme erstellt. Mit ihnen können erste Abschätzungen zu Datenübertragungszeiten und EEPROM-Schreibzeiten schnell und einfach durchgeführt werden.

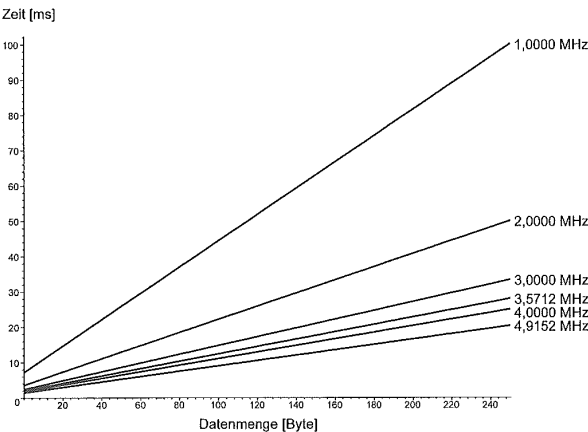


Bild 15.4 Diagramm zur Ermittlung der Übertragungszeit von Daten in Abhängigkeit vom Takt. Die dargestellten Werte beruhen auf folgenden Annahmen: Übertragungsprotokoll T = 1, Teiler 64, keine zusätzliche Schutzzeit (N = 12), kein Chaining, Prüfsummenverfahren XOR, keine Übertragungsfehler und Kommandos im Case 2 oder 3 (z. B.: READ BINARY oder UPDATE BINARY).

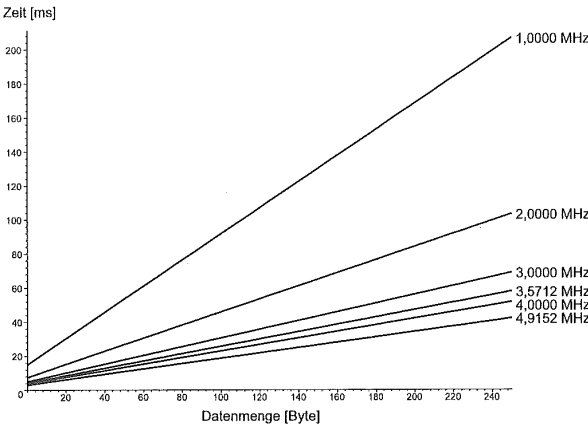


Bild 15.5 Diagramm zur Ermittlung der Übertragungszeit von Daten in Abhängigkeit vom Takt. Die dargestellten Werte beruhen auf folgenden Annahmen: Übertragungsprotokoll T = 1, Teiler 31, keine zusätzliche Schutzzeit (N = 12), kein Chaining, Prüfsummenverfahren XOR, keine Übertragungsfehler und Kommandos im Case 2 oder 3 (z. B.: READ BINARY oder UPDATE BINARY).

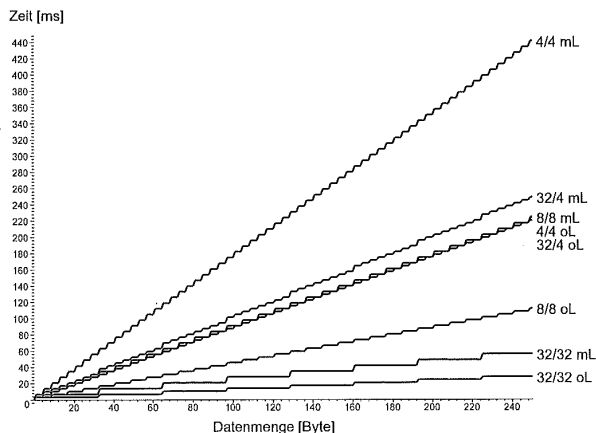


Bild 15.6 Diagramm zur Ermittlung der EEPROM-Schreibzeit von Daten in Abhängigkeit von der Pagegröße, jedoch unabhängig vom Takt. Dabei wurde angenommen, dass keine Lösch-/Schreibfehler auftreten und des Weiteren folgende Bezeichnungen benutzt werden: Der erste numerische Wert gibt die Pagegröße zum Löschen und der zweite numerische Wert die Pagegröße zum Schreiben an. Die Abkürzung „mL“ kennzeichnet das vorangegangene Löschen aller zu beschreibenden Pages und „oL“ das Schreiben ohne vorheriges Löschen aller Pages.

15.4 Typische Ausführungszeiten von Kommandos

Die Grundlage der Tabellen sind die durchschnittlichen Ausführungszeiten von Kommandos auf einer Chipkarte im Gutfall. Zur Zeitmessung wurden verschiedene Chipkarten mit unterschiedlichen Betriebssystemen herangezogen. Die Werte sind Mittelwerte und können im Einzelfall je nach Betriebssystem stark voneinander abweichen. Gemeinsam war bei allen gemessenen Chipkarten das Übertragungsprotokoll T = 1, eine Taktfrequenz von 3,5712 MHz, ein Teiler von 372, eine Schreib-/Löschzeit von 3,5 ms/4 Byte und ein Software-basierter DES-Algorithmus mit einer Geschwindigkeit von 17 ms/8 Byte.

Den Ausführungszeiten der mit „“ gekennzeichneten Kommandos ist eine starke Abhängigkeit von der jeweiligen Art der Implementierung und des unterstützten Funktionsumfangs eigen. Deshalb können Zeitangaben für diese Kommandos stark voneinander abweichen.

Tabelle 15.7 Durchschnittswerte von an unterschiedlichen Chipkarten-Betriebssystemen gemessenen Ausführungszeiten des Kommandos READ BINARY, das ein ähnliches Zeitverhalten wie READ RECORD hat. Die Werte in Klammern sind die Anzahl der gelesenen Daten. Die Messbedingungen werden im Text beschrieben.

Kommando	Ausführungszeit ohne Datenübertragung	Ausführungszeit mit Datenübertragung
READ BINARY (1 Byte)	2,02 ms	22,02 ms
READ BINARY (2 Byte)	2,03 ms	23,28 ms
READ BINARY (3 Byte)	2,04 ms	24,54 ms

Tabelle 15.7 (Fortsetzung)

Kommando	Ausführungszeit ohne Datenübertragung	Ausführungszeit mit Datenübertragung
READ BINARY (4 Byte)	2,04 ms	25,79 ms
READ BINARY (5 Byte)	2,05 ms	27,05 ms
READ BINARY (10 Byte)	2,12 ms	33,37 ms
READ BINARY (20 Byte)	2,23 ms	45,98 ms
READ BINARY (50 Byte)	2,54 ms	83,79 ms
READ BINARY (100 Byte)	2,98 ms	146,73 ms

Tabelle 15.8 Durchschnittswerte von an unterschiedlichen Chipkarten-Betriebssystemen gemessenen Ausführungszeiten des Kommandos UPDATE BINARY ohne und mit vorherigem Löschen der betreffenden EEPROM-Pages. UPDATE BINARY hat ein ähnliches Zeitverhalten wie UPDATE RECORD. Die Pagegröße betrug 4 Byte und die Schreib-/Löschzeit 3,5 ms. Die Werte in Klammern sind die Anzahl der geschriebenen Daten. Die sonstigen Messbedingungen werden im Text beschrieben.

Kommando	Ausführungszeit ohne Datenübertragung	Ausführungszeit mit Datenübertragung
UPDATE BINARY ohne Löschen (1 Byte)	6,95 ms	26,95 ms
UPDATE BINARY ohne Löschen (2 Byte)	7,01 ms	28,26 ms
UPDATE BINARY ohne Löschen (3 Byte)	7,03 ms	29,53 ms
UPDATE BINARY ohne Löschen (4 Byte)	7,11 ms	30,86 ms
UPDATE BINARY ohne Löschen (5 Byte)	7,12 ms	32,12 ms
UPDATE BINARY ohne Löschen (10 Byte)	7,33 ms	38,58 ms
UPDATE BINARY ohne Löschen (20 Byte)	12,33 ms	56,08 ms
UPDATE BINARY ohne Löschen (50 Byte)	18,16 ms	99,41 ms
UPDATE BINARY ohne Löschen (100 Byte)	18,81 ms	162,56 ms
UPDATE BINARY mit Löschen (1 Byte)	9,42 ms	29,42 ms
UPDATE BINARY mit Löschen (2 Byte)	9,51 ms	30,76 ms
UPDATE BINARY mit Löschen (3 Byte)	9,52 ms	32,02 ms
UPDATE BINARY mit Löschen (4 Byte)	9,48 ms	33,23 ms
UPDATE BINARY mit Löschen (5 Byte)	9,62 ms	34,62 ms
UPDATE BINARY mit Löschen (10 Byte)	9,85 ms	41,10 ms
UPDATE BINARY mit Löschen (20 Byte)	17,41 ms	61,16 ms
UPDATE BINARY mit Löschen (50 Byte)	25,87 ms	107,12 ms
UPDATE BINARY mit Löschen (100 Byte)	35,34 ms	179,09 ms

Tabelle 15.9 An unterschiedlichen Chipkarten-Betriebssystemen gemessene Ausführungszeiten von typischen Kommandos. Die Messbedingungen werden im Text beschrieben.

Kommando	Ausführungszeit ohne Datenübertragung	Ausführungszeit mit Datenübertragung
ASK RANDOM (8 Byte Zufallszahl)	26 ms	55 ms
CREDIT ☞	175 ms	222 ms
DEBIT ☞	235 ms	270 ms
EXTERNAL AUTHENTICATE	22 ms	51 ms
GET CARD DATA (8 Byte)	4 ms	33 ms
INITIALIZE IEP for Load ☞	89 ms	173 ms
INITIALIZE IEP for Purchase ☞	135 ms	201 ms
INTERNAL AUTHENTICATE	26 ms	65 ms
INVALIDATE	15 ms	34 ms
MUTUAL AUTHENTICATE	95 ms	163 ms
REHABILITATE	15 ms	33 ms
SEEK	3 ms	22 ms
SELECT FILE (mit 8 Byte AID)	3 ms	32 ms
SELECT FILE (mit 2 Byte FID)	3 ms	24 ms
VERIFY (8 Byte PIN) ☞	27 ms	56 ms

15.5 Hilfsmittel zur Anwendungsgenerierung

Für das Design von Chipkarten-Anwendungen existieren diverse Programme auf PC-Basis. Damit ist es ohne interne Kenntnisse des jeweiligen Betriebssystems der Chipkarte möglich, schnell und einfach komplette Anwendungen zu erstellen.

Mit den Werkzeugen wird üblicherweise zuerst ein Dateibaum aufgebaut, in dem sich dann die verschiedenen Anwendungen (d. h. DFs) und die dazugehörigen EFs befinden. Bei den EFs ist es natürlich notwendig, sowohl die Dateistruktur als auch die dazugehörigen Zugriffsbedingungen festzulegen. Besitzt das Chipkarten-Betriebssystem noch einen Zustandsautomaten für Kommandos, dann kann dieser ebenfalls auf der grafischen Benutzeroberfläche des PCs definiert werden. Teilweise lassen diese Konstruktionsprogramme für Anwendungen auch eine Konsistenzprüfung der definierten Zustandsautomaten zu. Da die so festgelegte Anwendung in ihren EFs diverse Daten und Schlüssel benötigt, kann man abschließend eine Verbindung mit einer Datenbank festlegen. Auf diese Weise sind dann die Inhalte der EFs in den einzelnen Karten mit den Datensätzen der Datenbank verbunden.

Ist die komplette Anwendung im Applikationsgenerator definiert, können noch verschiedene Rahmenparameter für das Betriebssystem der Chipkarte eingestellt werden. Dies ist beispielsweise das Übertragungsprotokoll und der dazugehörige Teiler. Nun kann die Anwendung versuchsweise in eine oder mehrere Chipkarten mit passender Speichergröße

geladen werden. Nachdem auf diese Weise einige Testkarten erstellt wurden, wird man sie in einem Terminal ausprobieren. Falls sich dabei herausstellt, dass weitere Änderungen notwendig sind, ist es möglich, die Anwendung in der Chipkarte zu löschen und anschließend eine revidierte Version in die Chipkarte zu laden.

Verlaufen alle Tests zufriedenstellend und benötigt man eine größere Stückzahl von Karten, kann in einer regulären Kartenfertigung oder mit einem Desktop-Personalisierungsgerät die gewünschte Anzahl Chipkarten hergestellt werden. Die mit dem PC-Programm erstellten Anwendungsdaten (Dateien, Kommandos, Zustände, ...) bilden dabei die Grundlage für die Komplettierung, Initialisierung und Personalisierung der Chipkarten, sodass die Durchlaufzeit in der Fertigung trotz hoher Flexibilität sehr niedrig bleibt.

Parallel zu oben vorgestellten Werkzeugen für die Gestaltung und Erstellung von Anwendungen gibt es auch Chipkarten-Simulatoren. Diese verhalten sich genau wie normale Chipkarten, bestehen aber lediglich aus einer Dummy-Chipkarte mit Anschlussleitungen zu einer PC-Schnittstelle. Auf diesem PC simuliert eine entsprechende Software dann die Chipkarte in Echtzeit. Natürlich lassen sich auch am PC alle Anwendungen, wie oben beschrieben, erzeugen und testen. Man benötigt aber in jedem Falle immer den PC zur Simulation, was in vielen Fällen aufgrund der Größe dieses Simulationsaufbaus problematisch ist.

Obiges Vorgehen muss bei Verwendung von Chipkarten mit nachladbarem ausführbarem Programmcode (z. B. Java Card) um den Schritt der Software-Entwicklung für die Chipkarten-Anwendung ergänzt werden. Da sich jedoch auch bei heutigen Java Card-basierten Chipkarten-Betriebssystemen die eigentlichen Anwendungen in der Praxis nicht mehr von denen auf der Grundlage herkömmlicher Chipkarten-Betriebssysteme unterscheiden, bleiben die weiteren Schritte identisch.

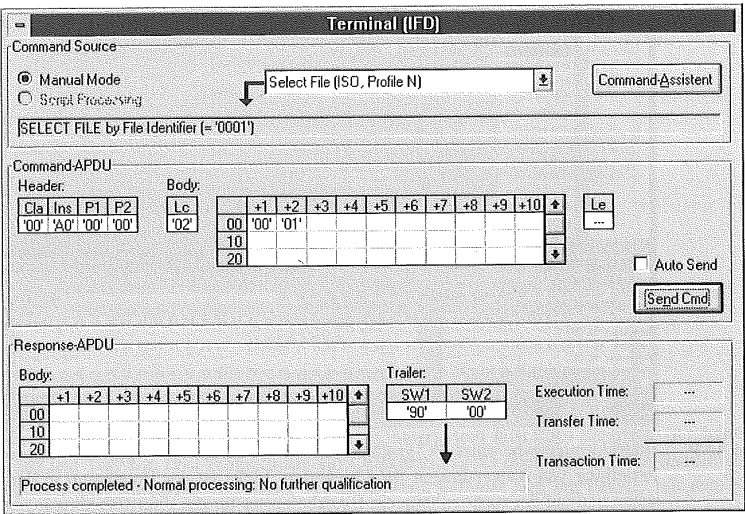


Bild 15.7 Das Terminalfenster des Simulationsprogramms „The Smart Card Simulator“ [Rankl].

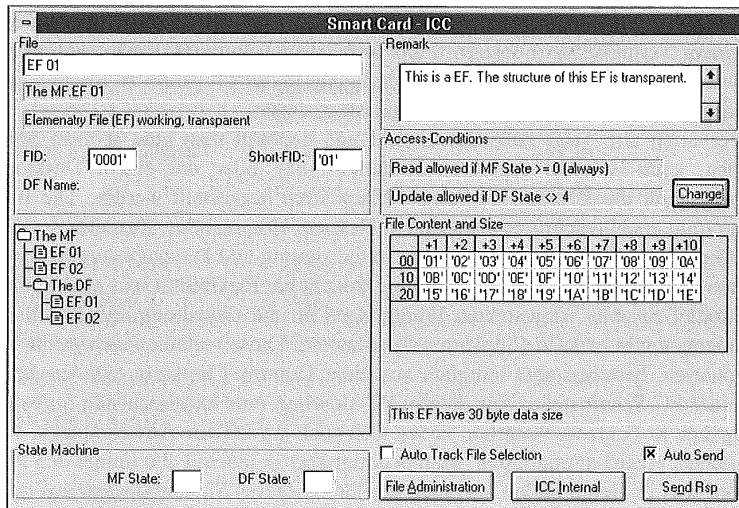


Bild 15.8 Das Chipkartenfenster des Simulationsprogramms „The Smart Card Simulator“ [Rankl].

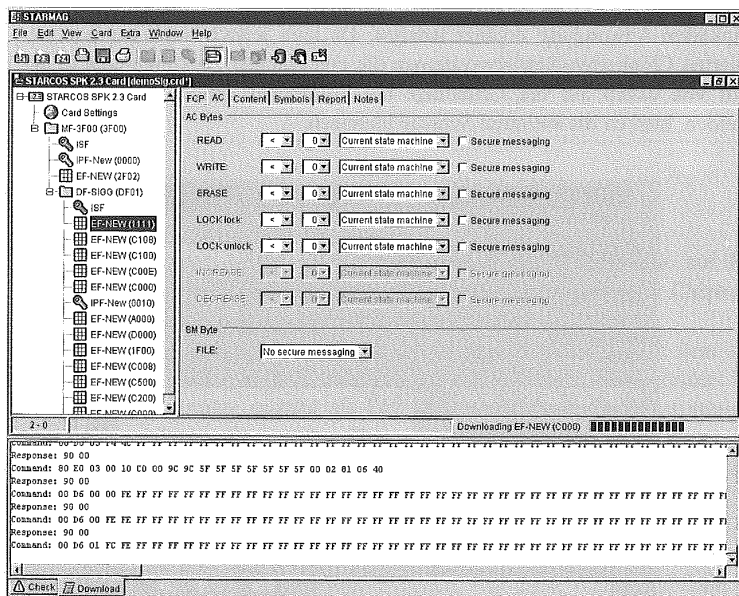


Bild 15.9 Bildschirmansicht eines PC-gestützten Softwarewerkzeugs (Applikationsgenerator) für die Erstellung von Anwendungen für Chipkarten. Links ist der Dateibaum in der Chipkarte zu sehen und auf der rechten Seite ein Eingabefenster für die Zugriffsbedingungen auf eine Datei. Unten befindet sich ein Fenster für die übertragenen Daten auf APDU-Ebene. (Giesecke & Devrient)

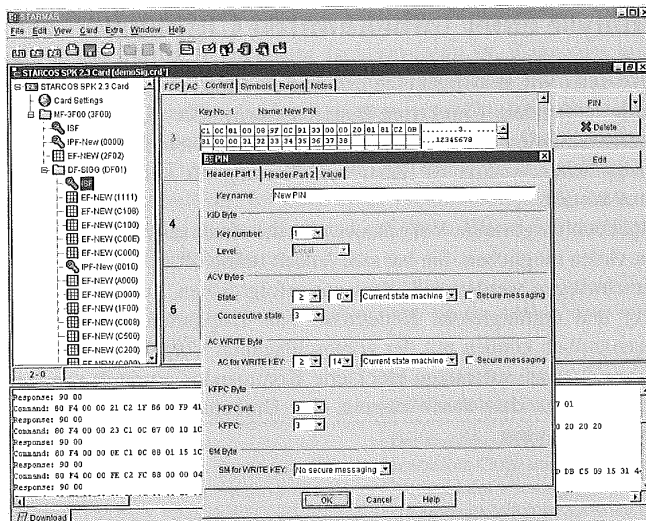


Bild 15.10 Bildschirmausgabe eines PC-gestützten Softwarewerkzeugs (Applikationsgenerator) für die Erstellung von Anwendungen für Chipkarten. Links ist der Dateibaum in der Chipkarte zu sehen. Auf der rechten Seite im Hintergrund ist der Inhalt der aktuell ausgewählten Datei dargestellt. Im überlappenden Fenster können die Zugriffsregeln im Rahmen eines EF_{ARR} editiert werden. (Giesecke & Devrient)

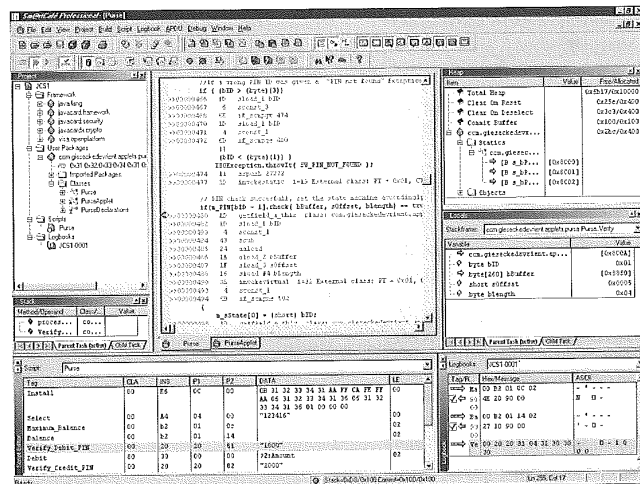


Bild 15.11 Beispiel für eine Entwicklungsumgebung mit integriertem Java Card Simulator für Entwicklung und Test von Java-Programmen auf Chipkarten. Links oben sind in einer Baumstruktur Klassen und Methoden dargestellt, rechts daneben befindet sich ein Fenster mit dem Java Source Code und daraus übersetztem Bytecode. Die Fenster darunter und ganz rechts zeigen Stack, Heap und diverse Variablen. Ganz unten befindet sich die Anzeige und Dekodierung für den Scriptprozessor. (Giesecke & Devrient)

15.6 Analyse einer unbekannten Chipkarte

Manchmal kommt es vor, dass man eine unbekannte Chipkarte analysieren muss, um beispielsweise das verwendete Chipkarten-Betriebssystem oder die darauf enthaltenen Anwendungen herauszufinden. Meist möchte man den dafür notwendigen Aufwand in zielgerechten Grenzen halten, sodass die Analyseausstattung oft nur aus einem Rechner und einem Terminal besteht.

Die im Folgenden vorgestellte Vorgehensweise der Analyse einer unbekannten Chipkarte ist eine von vielen möglichen. Sie hat sich jedoch schon oft in der Praxis bewährt, wobei hier nicht verschwiegen werden soll, dass zur erfolgreichen Auswertung der Ergebnisse viel Erfahrung und umfangreiche Kenntnisse von Chipkarten-Anwendungen vonnöten sind. Selbstverständlich können mit der vorgeschlagenen Vorgehensweise weder geheime Daten noch über Zustandsautomaten blockierte Kommandos ermittelt werden. Die Erfahrung hat jedoch gezeigt, dass damit zumeist eine Grobeinstufung der unbekannten Chipkarte getroffen werden kann.

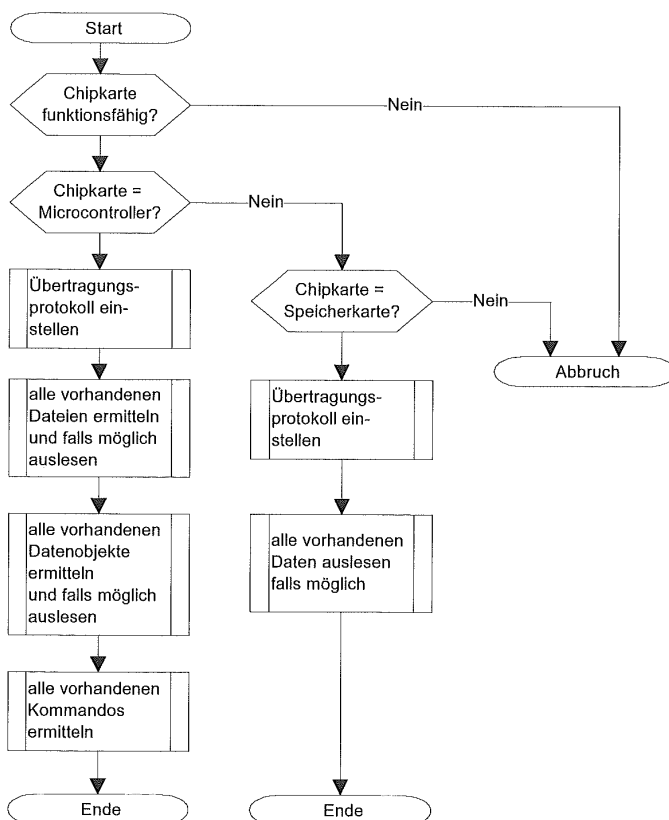


Bild 15.12 Ein möglicher prinzipieller Ablauf bei der Analyse einer unbekannten Chipkarte.

Das Flussdiagramm in Bild 15.12 zeigt den Ablauf im Überblick. In einem ersten Schritt wird ermittelt, ob die Chipkarte überhaupt funktionsfähig ist. Kann dies mit einem „Ja“ beantwortet werden, folgt im nächsten Schritt die Trennung in Speicher- und Prozessorchipkarten. Anschließend wird bei beiden Varianten versucht, das entsprechende Übertragungsprotokoll einzustellen und vorhandene Daten im Speicher, Dateien oder Datenobjekte zu lesen. Anhand dieser Daten wird dann manuell versucht, die auf der Chipkarte befindliche Anwendung zu ermitteln. Im Falle einer Prozessorchipkarte kann man oft auch die unterstützten Kommandos ausforschen und auf diese Weise Schlussfolgerungen über das jeweilige Chipkarten-Betriebssystem ziehen.

Tabelle 15.10 Pseudocode für eine mögliche Vorgehensweise bei der Analyse einer unbekannten kontaktbehafteten Chipkarte

AnalysisUnknownCard:	Programm für die Analyse einer unbekannten Chipkarte
führe Anschaltsequenz als Mikroprozessor-Chipkarte durch und wähle dabei Versorgungsspannung gemäß ISO/IEC 7816-3 und ISO/IEC 7816-3 Amd.1 aus IF (ATR empfangen) THEN AnalysisSmartCard führe Anschaltsequenz als Speicher-Chipkarte durch IF (ATR empfangen) THEN AnalysisMemoryCard STOPP	Die typische Anschaltsequenz für Mikrocontroller-Chipkarten mit der entsprechenden Auswahl der Versorgungsspannung ist ein erfolgversprechender Einstieg in die Analyse. Eine Beschreibung der Vorgehensweise findet sich im Abschnitt 3.3.2 Versorgungsspannung. Falls ein ATR empfangen werden konnte, handelt es sich um eine Mikrocontroller-Chipkarte. Da es keine Mikrocontroller-Chipkarte ist, wird nun die Anschaltsequenz für Speicher-karten ausprobiert. Falls ein ATR empfangen werden konnte, handelt es sich um eine Speicherkarte. Die Chipkarte konnte nicht erkannt werden, da sie keinen ATR gesendet hat; deshalb Abbruch der Analyse.
AnalysisSmartCard:	Es handelt sich bei der unbekannten Chipkarte um eine Mikroprozessor-Chipkarte.
AUSGABE: empfangener ATR stelle gewünschtes Übertragungsprotokoll ein // Scanne alle Dateien selektiere EF _{DIR} IF (EF _{DIR} = vorhanden) THEN (Lese den Inhalt des EF _{DIR} AUSGABE: Inhalt EF _{DIR})	Aufgrund des ATR können evtl. Rück-schlüssel auf die Chipkarte bzw. die auf ihr enthaltenen Anwendungen getroffen werden. Es wird das im ATR gewünschte Übertragungsprotokoll (T = 0 oder T = 1) mit den entsprechenden Übertragungsparametern eingestellt. Falls ein EF _{DIR} vorhanden ist, befinden sich darin Informationen zu den auf der Chipkarte enthaltenen Anwendungen.

Tabelle 15.10 (Fortsetzung)

AnalysisUnknownCard:	Programm für die Analyse einer unbekannten Chipkarte
<pre> selektiere typische AIDs AUSGABE: Ergebnisse der Selektionsversuche selektiere alle EFs mittels FID im MF und versuche, die Inhalte der selektierten EFs zu lesen AUSGABE: Ergebnisse der Selektionsversuche, Zugriffsbedingungen und Leseversuche sowie ggf. die Dateiinhalte selektiere alle EFs mittels FID in jedem selektierbaren DF und versuche die selektierten EFs auszulesen AUSGABE: Ergebnisse der Selektionsversuche, Zugriffsbedingungen und Leseversuche sowie ggf. die Dateiinhalte // Scanne alle Datenobjekte versuche, alle Datenobjekte mittels GET DATA auszulesen AUSGABE: Ergebnisse der Leseversuche sowie ggf. die Datenobjektinhalte // Scanne alle Kommandos teste alle CLA und INS Kombinationen AUSGABE: mögliche erlaubte CLA und INS Kombinationen STOPP </pre>	In diesem Analyseschritt werden Selektionsversuche mit typischen AIDs durchgeführt. Hierbei kann die AID-Liste aus dem Anhang verwendet werden. Sind ein oder mehrere Selektionsversuche mittels AID positiv verlaufen, ist die entsprechende Anwendung eindeutig identifiziert. Ausgabe der Ergebnisse der Selektionsversuche. Unter Umständen können aufgrund der EF-Strukturen, Zugriffsbedingungen und EF-Inhalte Schlussfolgerungen auf die Chipkarte und deren Anwendungen getroffen werden. Ausgabe der Ergebnisse der Selektions- und Leseversuche. Unter Umständen können aufgrund der EF-Strukturen, Zugriffsbedingungen und EF-Inhalte Schlussfolgerungen auf die Chipkarte und deren Anwendungen getroffen werden. Handelt es sich um ein DF mit registriertem AID, so kann in den entsprechenden Spezifikationen (falls verfügbar) der Dateinhalt interpretiert werden. Ausgabe der Ergebnisse der Selektions- und Leseversuche. Unter Umständen können aufgrund der Datenobjektinhalte Schlussfolgerungen auf die Chipkarte und deren Anwendungen gezogen werden. Handelt es sich um ein DF mit registriertem AID, kann in den entsprechenden Spezifikationen (falls verfügbar) der Dateinhalt interpretiert werden. Mit dieser Methode (im Detail beschrieben im Abschnitt 8.2.4.2 Angriffe auf der logischen Ebene) können die Kommandos und das Secure Messaging des Chipkarten-Betriebssystems ermittelt werden. Ausgabe der Ergebnisse der Kommandoanalyse. Ende der Analyse.

Tabelle 15.7 (Fortsetzung)

AnalysisMemoryCard:	Es handelt sich bei der unbekannten Chipkarte um eine Speicherkarte.
stelle Übertragungsprotokoll ein	Probiere der Reihe nach alle Übertragungsprotokolle für Speicherkarten durch und verwende dasjenige Übertragungsprotokoll, das eingestellt ist, sobald korrekte Daten empfangen wurden.
AUSGABE: Ermitteltes Übertragungsprotokoll	Ausgabe des ermittelten Übertragungsprotokolls.
lese alle Daten aus der Speicherkarte	
AUSGABE: Dateninhalt der Speicherkarte	Unter Umständen können aufgrund des Dateninhalts Schlussfolgerungen auf die Speicherkarte und die Anwendung getroffen werden.
STOPP	Ende der Analyse.

15.7 Vorgehensmodelle und Prozessreifegrad

Zur Erstellung von Software gibt es die unterschiedlichsten Vorgehensweisen. Diese können durch Vorgehensmodelle allgemein beschrieben werden, sodass sie für die unterschiedlichen Softwareentwicklungsprojekte eingesetzt werden können. Vorgehensmodelle werden oft auch Prozess-Modelle genannt. Ein Vorgehensmodell beschreibt in allgemeiner Form die durchzuführenden Aktivitäten, die Reihenfolge der Aktivitäten, Verantwortlichkeiten und Kompetenzen. Im Übrigen kann man die Vorgehensmodelle zur Entwicklung von Software in den meisten Fällen für beinahe alle Entwicklungstätigkeiten und kreative Aktivitäten als Anleitung für das Realisieren verwenden.

Bei der Trivialvorgehensweise zur Softwareentwicklung setzt sich der Programmierer unmittelbar nach kurzer mündlicher Auftragserteilung an seinen Rechner und erstellt ein Programm, das er anschließend so lange anpasst, bis die am meisten störenden Fehler beseitigt sind und der Auftraggeber halbwegs zufriedengestellt ist. Diese Art des einfältigen Vorgehens findet man erstaunlicherweise nicht nur in Schilderungen aus den Anfängen der Softwareentwicklung, sondern in vielen Varianten auch noch heute bei kleinen und großen Entwicklungen. Unzweifelhaft kann man mit dieser „Garagenfirmenmentalität“ innovative und auch konkurrenzfähige Programme erstellen. Doch das Ergebnis hinsichtlich Fertigstellungstermin, Entwicklungskosten und Softwarequalität ist bei dieser Vorgehensweise nur in sehr grobem Rahmen vorhersagbar. Bei Softwareprojekten mit komplizierter Aufgabenstellung und mehreren Entwicklern kann dabei die Komplexität so groß werden, dass man entweder das zur Verfügung stehende Budget und den geplanten Zeitrahmen stark überschreitet oder womöglich das ganze Entwicklungsprojekt vor der Fertigstellung abgebrochen werden muss. Man schätzt, dass etwa 25 % aller Softwareprojekte unvollendet eingestellt werden. Um die drei klassischen Faktoren „Termin“, „Kosten“ und „Qualität“ möglichst genau vorhersagbar zu machen, werden deshalb bei professioneller Softwareentwicklung Vorgehensmodelle als Entwicklungsrahmen eingesetzt.

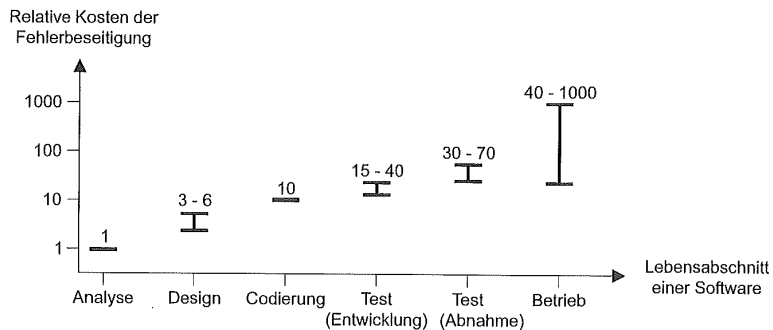


Bild 15.13 Gegenüberstellung der Kosten für die Beseitigung eines Fehlers in Abhängigkeit des Lebensabschnitts einer Software bezogen auf die Analyse. Die Daten basieren auf einer Veröffentlichung von [Boehm 81].

Industrielle Fertigungsprozesse – und dazu gehört selbstverständlich auch die Entwicklung von Software – weisen vier grundsätzliche Merkmale auf:

1. Der Entwicklungsprozess ist an Phasen orientiert.
2. Jede Entwicklungsphase liefert Ergebnisse, die als Grundlage für die Folgephase verwendet werden.
3. Die Ergebnisse einer Phase werden vor dem Start der Folgephase überprüft.
4. Das Ergebnis der Phasen sind abstrakte Darstellungen des Produkts, die jedoch von Phase zu Phase konkreter werden, bis in der letzten Phase das Produkt selbst entsteht.

Diese Merkmale stammen eigentlich aus dem klassischen Maschinenbau, sind jedoch prinzipiell auch auf qualifizierte Softwareentwicklung anwendbar. Sie sind die Basis für die in den folgenden Abschnitten vorgestellten Vorgehensmodelle.

Die Entwicklung von Software, und im Übrigen auch die meisten anderen Entwicklungstätigkeiten, erfordert vier generelle Tätigkeiten: Als erste Tätigkeit ist die Kardinalfrage „Was ist zu entwickeln?“ zu beantworten. Das heißt, das Entwicklungsziel muss möglichst genau und eindeutig festgelegt werden. Eindeutig festgelegt bedeutet, dass die allseits so beliebten Konjunktive wie „sollte“, „könnte“, „müsste“ in der Aufgabenstellung nicht verwendet werden dürfen. Die Praxis hat gezeigt, dass die während dieser Tätigkeit entstehenden Dokumente möglichst wenig Fließtext enthalten sollten. Ideal sind tabellarische Formen oder Grafiken, da beide wenig Spielraum für Ungenauigkeiten lassen. Das Dokument, das bei dieser Aktion entsteht, wird als Anforderungsspezifikation (*user requirement specification*) oder manchmal auch als Lastenheft bezeichnet. Die Tätigkeit zur Erstellung dieses Dokuments wird Analyse oder Anforderungsermittlung genannt.

Die Anforderungsspezifikation bildet die Grundlage für alle weiteren Tätigkeiten der Entwicklung. Vollständigkeit und Eindeutigkeit sind deshalb von elementarer Bedeutung für dieses Dokument, dass nach einem abschließenden Review auch nicht mehr geändert werden darf. Schon die Modifikation eines einzelnen Wortes in der Anforderungsspezifikation kann unter Umständen große Auswirkungen auf alle nachfolgenden Entwicklungsschritte haben.

Als Beispiel seien hier die Anforderungen für das Mobilfunksystem UMTS angeführt, in dem ursprünglich ausnahmslos asymmetrische Kryptoalgorithmen gefordert wurden. Nach

langen Diskussionen wurde zu einem relativ späten Zeitpunkt das „a“ im Wort „asymmetrisch“ gestrichen. Dies führte dazu, dass die auf den Anforderungen aufbauenden Spezifikationen und auch eine Reihe existierende Realisierungen komplett geändert werden mussten. Dieses extrem Beispiel soll die Bedeutung der Anforderungsspezifikation für alle weiteren Entwicklungsschritte eindringlich vor Augen führen.

Die zweite Kardinalfrage im Entwicklungsprozess lautet: „Wie entwickelt man?“ Dieses „Wie“ deckt einerseits die organisatorischen Rahmenbedingungen als auch andererseits die Struktur und den Aufbau der zu entwickelnden Software ab. Dabei werden alle Funktionen und die Daten der zu erstellenden Software vollständig und eindeutig beschrieben. Der Hauptzweck ist dabei, die Komplexität der Gesamtentwicklung auf ein beherrschbares Niveau zu reduzieren, Änderbarkeit und Wiederverwendbarkeit von Komponenten sicherzustellen und gegebenenfalls eine arbeitsteilige Realisierung vorzubereiten. Die Klärung der Wie-Frage ist bei professioneller Softwareentwicklung der Hauptaufwand des gesamten Entwicklungsprozesses. Die Tätigkeit trägt den Namen „Design“ oder „Entwurf“ und das Ergebnis dieser Tätigkeit die Bezeichnung „Pflichtenheft“ oder „Spezifikation“. In diesem Buch wird durchgehend für die Tätigkeit der Begriff „Design“ und für das dabei entstehende Dokument die Bezeichnung „Spezifikation“ benutzt.

Analog der Anforderungsspezifikation muss selbstverständlich auch die Spezifikation eindeutig sein und darf keinerlei Interpretationsspielräume lassen. Ein idealer Zustand ist dann erreicht, wenn man die erstellte und geprüfte Spezifikation außenstehenden Personen übergeben und diese dann anhand des Dokuments ohne Rückfragen die Software korrekt erstellen können.

Nachdem nunmehr geklärt ist, was entwickelt werden soll, wie es gemacht wird und wie es aufgebaut ist, kann mit der Realisierung begonnen werden. Dies ist der Zeitpunkt, zu dem Softwareentwickler mit Garagenfirmenmentalität überhaupt erst in die Entwicklung einsteigen. Hat ein Entwickler eine vollständige und nicht interpretationsfähige Spezifikation, so ist der Aufwand für die Realisierung im Verhältnis zu den beiden vorgehenden Schritten deutlich niedriger. Es wird nunmehr Modul für Modul programmiert und durch dieselbe Person auch ein Debugging durchgeführt. Das Ergebnis dieser Tätigkeit sind Softwaremodule, die in den meisten Fällen keine Trivialfehler mehr aufweisen sollten. Parallel zur Programmierung entsteht die so genannte Entwicklerdokumentation, die alle Unterlagen enthält, die der Programmierer während seiner Entwicklungstätigkeit erstellt. Hier hat die Praxis eindringlich gezeigt, dass direkt im Sourcecode dokumentiert werden sollte, da sich damit die Suche nach Dokumenten wesentlich einfacher gestaltet und auf diese Weise vor allem am wenigsten Information über die Zeit hinweg verloren geht. Dies wird durch entsprechende Werkzeuge (wie z. B. Javadoc) unterstützt, die dann automatisch aus den Informationen im Sourcecode adäquate Entwicklerdokumentationen erzeugen können. Für den Aufbau des Sourcecodes und die Dokumentation desselbigen sind im Allgemeinen entsprechende Richtlinien zu befolgen.

Nach der Realisierung wird das erstellte Modul oder mehrere Module im Verbund bis hin zum gesamten Programm geprüft. Dies wird auch Test genannt. Ideal ist es, wenn die Tester möglichst unabhängig von den Programmierern ihrer Tätigkeit nachgehen und die Programmierer als Testergebnis nur „Okay“ bzw. „nicht Okay“ erhalten. Der Vorteil dieses Vorgehens ist, dass die Programmierer nicht genau wissen, was getestet wird, und deshalb ihre Programme möglichst vollständig debuggen müssen. Auf diese Weise wird ein

relativ gutes „Vier-Augen-Prinzip“ eingeführt, bei dem deutlich mehr Fehler gefunden werden, als wenn die Programmierer *en detail* die zu bestehenden Tests kennen und ihre Software dahingehend perfektionieren. Der Aufbau und die Durchführung von Tests orientiert sich wiederum an eigenen Methodiken, die in Kapitel 9 ausführlicher beschrieben sind.

Nach bestandenen Tests in der Softwareentwicklung finden bei großen Systemen mit unterschiedlichen Komponenten und mehreren Komponentenlieferanten so genannte Kompatibilitätsprüfungen statt. Üblicherweise wird nach bestandenen Kompatibilitätsprüfungen auch noch eine explizite Abnahme der Software durch den Auftraggeber durchgeführt. Diese kann jedoch auch auf den vorangehenden Tests und Prüfungen basieren und nur noch formalen Charakter haben. Anschließend wird die Software freigegeben und kann eingesetzt werden.

Nach Auslieferung und während der Benutzung der Software kann je nach Anwendung Wartungs- und Pflegeaufwand anfallen. Dies ist die Beseitigung von nicht tolerierbaren Fehlern sowie kleinere funktionale Änderungen und Anpassungen an der bestehenden Software. Die Wartung von Software führt fast immer dazu, dass selbst gut strukturierte und organisierte Programme mit jeder neuen Version eine immer undurchsichtigere Struktur bekommen. Oftmals wird auch die dazugehörige Dokumentation bzw. die zugrunde liegende Spezifikation nicht mitgeführt, was zu einem Auseinanderlaufen der abstrakten Darstellung in der Spezifikation und dem realen Produkt führt. Aus dieser Falle gibt es zwei Auswege: Bei einfachen, kleineren Programmen führt man die Wartungsarbeiten ohne größeren Aufwand durch, plant aber parallel dazu eine neue vollständig überarbeitete Version (*refactoring*). Diese Vorgehensweise kann man sich jedoch nur bei kleineren Programmen oder bei Prototypen erlauben. Bei der Wartung größerer Programme muss äußerst sorgfältig vorgegangen werden, was bedeutet, dass alle Spezifikationen und Dokumente entsprechend angepasst werden müssen und auch bei Änderungen im Source Code entsprechend behutsam vorgegangen werden muss. Dies kann es auch erforderlich machen, dass vorbeugend größere Teile des Source Codes umgeschrieben werden müssen, um weiterhin eine klar strukturierte Architektur der Software zu gewährleisten.

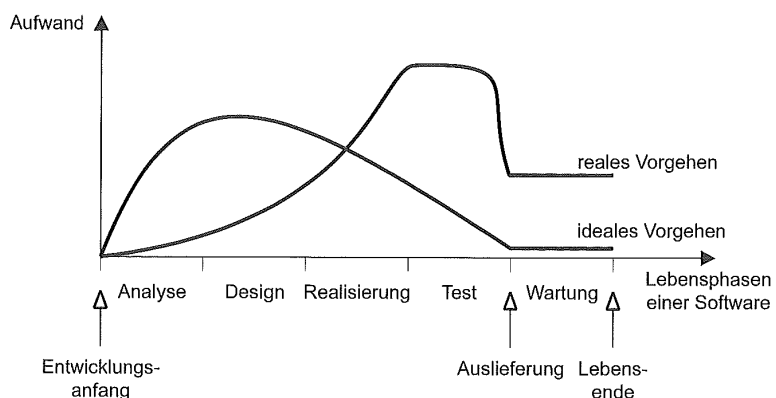


Bild 15.14 Gegenüberstellung der Aufwände für Entwicklung in Abhängigkeit der Software-Lebensphase beim idealen Vorgehen und bei vielen realen Entwicklungsprojekten.

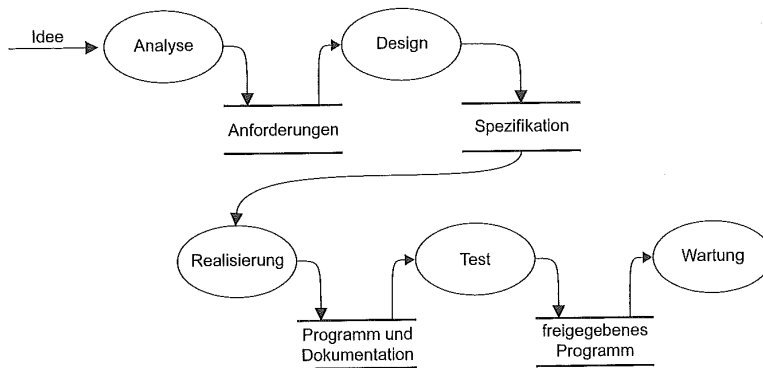


Bild 15.15 Datenflussdiagramm mit den wesentlichen Aktivitäten und Dokumenten bei einer typischen Software-Entwicklung für Chipkarten.

15.7.1 Vorgehensmodelle (*life cycle models*)

Die in den folgenden Abschnitten dargestellten Vorgehensmodelle zeigen in einheitlicher grafischer Form alle entwicklungsbezogenen Aktivitäten. Die grafische Darstellung ist in einer IT-konformen Notation, ähnlich den Datenflussdiagrammen, gehalten, um die Vorgehensmodelle auf einen Blick verständlich zu machen und gut untereinander vergleichen zu können. Dargestellt sind nur solche Modelle, die eine Reinform und keine Mischform darstellen. Reinformen haben den Vorteil, dass sowohl die positiven als auch die negativen Eigenschaften eines Modells deutlich ersichtlich sind. Außerdem wurde die Auswahl auf Vorgehensmodelle ausgedehnt, die bei der Entwicklung von Software für Chipkarten sinnvoll einsetzbar sind. Die Beschreibung der Vorgehensmodelle enthält jeweils genügend Informationen, um sie in ihren Grundzügen zu verstehen, anzuwenden und womöglich in bestimmten Fällen nicht zu benutzen.

Neben den aufgeführten Vorgehensmodellen existieren noch viele weitere Modelle zur Entwicklung von Software. Dies sind jedoch oft Mischformen der vorgestellten Modelle oder

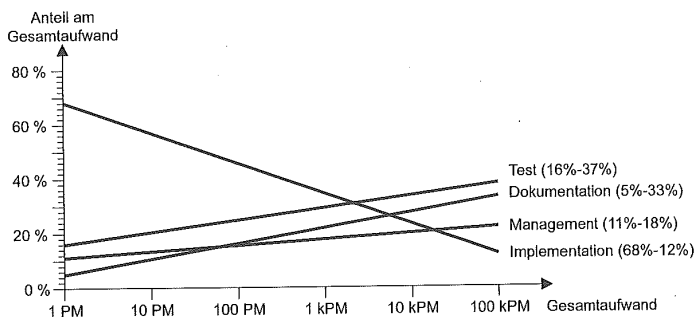


Bild 15.16 Gegenüberstellung der Verteilung der Aufwände am Gesamtaufwand einer Entwicklung. Die Summe der einzelnen separaten Aufwände beträgt immer 100 %, die Abkürzung PM steht für Personenmonat. Die Daten basieren auf einer Veröffentlichung von C. Jones [Jones 91].

Spezialisierungen. Manche Auftraggeber haben eigene Vorgehensmodelle definiert; dies ist insbesondere in sicherheits- und zuverlässigkeitssensiblen Bereichen wie in der Militär-, Kern- und Luftfahrttechnik der Fall. Tiefergehende Informationen zu diesem Thema finden sich beispielsweise in [Balzert 98].

15.7.1.1 Wasserfall-Modell

Das Prinzip des Wasserfall-Modells wurde 1956 von Benington mit integrierten Rücksprüngen 1970 von Royce publiziert und hat 1981 von Boehm seinen Namen von der treppenförmigen Anordnung der Aktivitäten, die einem Wasserfall ähnlich sehen, erhalten. Es war eines der ersten Vorgehensmodelle und ist gegenüber der Trivialvorgehensweise „from brain to keyboard“ ein deutlicher Fortschritt.

Die wesentlichen Merkmale des Wasserfall-Modells sind sein sequentieller Entwicklungsablauf und ein geradliniges Top-down-Vorgehen. Jede der aufgeführten Aktivitäten wird vollständig und in der vorgegebenen Reihenfolge durchgeführt, wobei es auch erlaubt ist, einen Rücksprung zurück zur vorhergehenden Aktivität durchzuführen. Dieses Vorgehensmodell ist stark dokumentengetrieben, was bedeutet, dass bei jeder Aktivität explizit Dokumente erstellt werden, die nach dem Abschluss der Aktivität, der in der Regel durch ein Review vonstatten geht, weiter verwendet werden. Das Wasserfall-Modell sieht eine Beteiligung des Auftraggebers nur am Anfang bei der Definition vor. Anschließend sind bis zur endgültigen Freigabe der Software nur noch die Entwickler eingebunden.

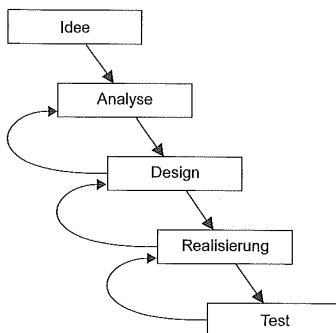


Bild 15.17 Überblickhafte Darstellung des auf die Belange der Software Entwicklung für Chipkarten angepassten Wasserfall-Modells.

Das Wasserfall-Modell ist ein einfaches Vorgehensmodell und benötigt nur wenig Koordinierungsaufwand. Die Anwendung dieses Modells führt zu einem eindeutigen und klar ersichtlichen Entwicklungsablauf. Der vorgegebene sequentielle Ablauf der Aktivitäten führt jedoch bei unvollständigen oder noch unbekannten Vorgaben sehr leicht zu Problemen, da die jeweils folgende Aktivität nur nach vollständigem Abschluss der vorhergehenden begonnen werden darf. So ist es im Wasserfall-Modell beispielsweise nicht vorgesehen, einfache Prototypen oder Tests während der Definitions- oder Entwurfsphase zur Erforschung von Realisierungsmöglichkeiten durchzuführen. Nachteilig ist in diesem Zusammenhang auch, dass der Auftraggeber nur am Anfang während der Definitionsphase beteiligt ist.

Das Wasserfall-Modell eignet sich gut für Entwicklungen, die sich nicht auf technischem Neuland bewegen und in ähnlicher Form mindestens einmal durchgeführt worden sind. Bei diesen Entwicklungen werden sich in der Regel keine Überraschungen einstellen, da sowohl die technischen als auch organisatorischen Rahmenbedingungen beim Start im Großen und Ganzen bekannt sind. Als Beispiel aus dem Chipkartenbereich sei hier die Portierung eines Chipkartenbetriebssystems von einem Chiptyp auf einen anderen aufzuführen. Dabei müssen „lediglich“ die jeweiligen Eigenschaften des Mikrocontrollers berücksichtigt und, sofern in Assembler programmiert wurde, die entsprechenden Maschinenbefehle umcodiert werden. Zusammengefasst kann man sagen, dass sich das Wasserfall-Modell vor allem für nicht schöpferische Entwicklungstätigkeiten eignet. Sobald technisches Neuland betreten wird und einfallsreiche schöpferische Entwicklungen notwendig sind, sollte kein Wasserfall-Modell verwendet werden, da es dafür nicht geeignet ist. Wird es dennoch verwendet, kann es bei der Realisierung zu massiven Problemen führen, da kritische Punkte vorab in Analyse und Design bei diesem Entwicklungsmodell nicht genügend erforscht werden können.

15.7.1.2 V-Modell

Das V-Modell ist vom Prinzip her ein Wasserfall-Modell mit integrierter Qualitätssicherung. Es hat seinen Namen von der üblicherweise v-förmigen Darstellung der erforderlichen Aktivitäten. Das V-Modell hat ebenfalls, analog zum Wasserfall-Modell, einen zeitlich sequentiellen Entwicklungsablauf. Den einzelnen Entwicklungsaktivitäten Analyse, Design und Realisierung stehen entsprechende Testaktivitäten gegenüber. Wenn nötig, ist ein Rücksprung auf die jeweils vorhergehende Aktivität möglich. Das V-Modell wurde ursprünglich für eingebettete Systeme (*embedded systems*) – darunter fallen auch Chipkarten-Mikrocontroller – entwickelt und ist ein verhältnismäßig aufwändiges Vorgehensmodell. Dem großen erforderlichen Aufwand, vor allem hinsichtlich der Dokumentation, steht jedoch eine sehr hohe Qualität der entwickelten Software gegenüber. Deshalb wird das V-Modell vor allem bei größeren Entwicklungen mit hohen Qualitätsanforderungen eingesetzt, wie beispielsweise bei Chipkarten-Betriebssystemen oder militärischen Anwendungen. Eine detaillierte Darstellung des V-Modells findet sich in [Dröschel 99].

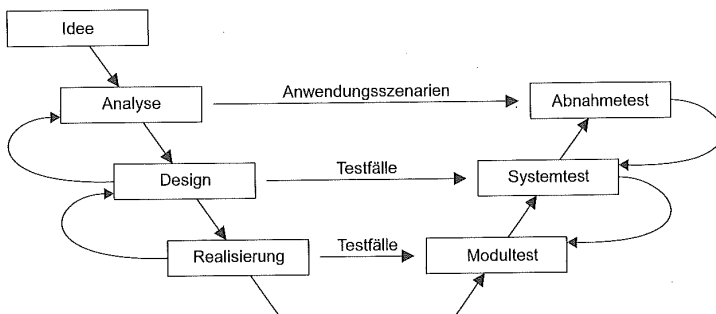


Bild 15.18 Überblickhafte Darstellung des auf die Belange der Software-Entwicklung für Chipkarten angepassten V-Modell.

In der Entwicklung von Software für Chipkarten kann das V-Modell sehr gut angewandt werden, wenn es darum geht, vorgegebene Spezifikationen in Programmcode mit hoher Qualität umzusetzen. Als Exempel sei hier die Erstellung einer GSM 11.11-Anwendung auf einem vorhandenen Chipkarten-Betriebssystem genannt. Damit sind die einzelnen Kommandos und das Dateisystem analog der detaillierten Schnittstellenspezifikation GSM 11.11 ohne größeren kreativen Einsatz zu programmieren. Wichtig ist vor allem, dass die Vorgaben der GSM 11.11 vollständig eingehalten werden.

Das V-Modell sollte nicht bei vollständigen Neuentwicklungen eingesetzt werden, da es, analog dem Wasserfall-Modell, keine iterativen Entwicklungsschritte und keine Beteiligung des Auftragsgebers nach der Analyse vorsieht. Auf einen Nenner gebracht, ist das V-Modell ideal für Entwicklungen, bei denen kein technisches Neuland betreten wird und Fehlerarmut und spezifikationstreue Priorität haben.

15.7.1.3 Prototypen-Modell

Sowohl Wasserfall- als auch V-Modell sehen klar abgegrenzte und in einer bestimmten Reihenfolge aufeinander folgende Aktivitäten vor. Bei vielen Softwareentwicklungen führt dies jedoch zu Problemen, da kein „Spielraum“ für kreative Lösungen durch Experimente vorgesehen ist. Das Prototypen-Modell führt diesen zusätzlichen Freiheitsgrad in die Vorgehensmodelle ein.

Ein Softwareprototyp dient dazu, nur bestimmte Teile einer gesamten Software zu demonstrieren. Bei der Variante des horizontalen Prototyps werden nur bestimmte Ebenen einer Software realisiert. Bei einem Chipkarten-Betriebssystem wäre beispielsweise ein horizontaler Prototyp die vollständige Realisierung aller Übertragungsprotokolle – ohne ausgearbeitete Kommandos, die in diesem Falle nur Dummy-Antworten zurücksenden. Mit diesem horizontalen Prototyp lässt sich die komplette Kommunikation zwischen Terminal und Karte austesten. Diese Art von Prototyp könnte Verwendung finden, um experimentell zu zeigen, dass eine hohe Datenübertragungsgeschwindigkeit per Software ohne Hardwareunterstützung durch einen UART (*universal asynchronous receiver transmitter*) möglich ist.

Das Gegenteil des horizontalen Prototyps ist folglich der vertikale Prototyp, bei dem nur bestimmte Teile der Software durch alle Ebenen hindurch realisiert sind. Angelehnt an das vorherige Beispiel wäre dies bei einem Chipkarten-Betriebssystem die lückenlose Realisierung eines Kommandos, z. B. INTERNAL AUTHENTICATE, und eines einzigen, evtl. sogar nur rudimentär vorhandenen, Übertragungsprotokolls. Mit diesem Prototyp könnte man dann das Zeitverhalten der Authentisierung der Chipkarte inklusive der gesamten Datenübertragung vermessen. Diese Vorgehensweise ist im Übrigen für eine derartige Aufgabenstellung die einzig mögliche, da sich exakte und zuverlässige Werte für das Zeitverhalten nur im Experiment, und nicht rechnerisch, bestimmen lassen.

Ein Prototyp bei der Softwareentwicklung zeigt bestimmte Eigenschaften der Software im praktischen Einsatz, um vor der Gesamtrealisierung alternative Lösungsvarianten experimentell zu erproben. Auf der Grundlage des Prototypen wird dann entweder der Entwurf vervollständigt oder der Prototyp wird dahingehend weiterentwickelt, dass er als Teil der zu entwickelnden Software weiterverwendet werden kann.

Da sich kritische Anforderungen anhand der entstehenden Prototypen vorab im Experiment überprüfen lassen, eignet sich das Prototypen-Vorgehensmodell gut für Entwicklungen

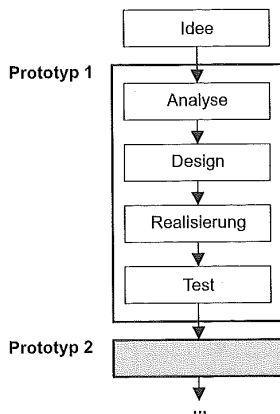


Bild 15.19 Überblickhafte Darstellung des auf die Belange der Software-Entwicklung für Chipkarten angepassten Prototypen-Modells.

mit ungenauen Vorgaben. Als weitere positive Eigenschaft dieses Vorgehensmodells kann der Auftraggeber die Prototypen begutachten und anschließend seine – eventuell kargen – Anforderungen verfeinern, sodass es zu einer besseren Abstimmung mit den Entwicklern kommt. Die für Experimente zur Verfügung stehenden Prototypen und die verbesserte Abstimmung führen infolgedessen zu einer Risikominimierung bei der Entwicklung.

Diesen Vorteilen stehen auch einige nicht zu unterschätzende Nachteile gegenüber. Das schrittweise Vorantasten von Prototyp zu Prototyp ist sowohl zeit- als auch kostenaufwendig, da in der Regel etliche Entwicklungsschritte auf der Suche nach der richtigen Lösung mehrmals durchlaufen werden müssen. Die größte Gefahr für die Softwareentwicklung nach dem Prototypen-Modell ist jedoch weniger technischer als organisatorischer Natur. Es besteht das Risiko, dass ein Prototyp durch Zeit- oder Kostendruck als fertiges Produkt umdeklariert und somit viel zu früh ausgeliefert wird. Das Ergebnis ist dann eine Software, die weder die geforderte Funktionalität aufweist noch vollständig ausgetestet ist und somit wahrscheinlich noch zahlreiche Fehler enthält. Vielfach kommt es bei dieser Art der Fertigstellung noch dazu, dass die Dokumentation entweder unvollständig oder im schlimmsten Fall überhaupt nicht vorhanden ist.

Als Bemerkung am Rande sei hier eingefügt, dass die Vorgehensweise nach dem Prototypen-Modell dem Verhalten vieler Hobby-Programmierer entspricht. Sie beginnen, in der Regel ohne Definition und Entwurf, mit einem relativ kleinen Kernprogramm, erweitern dieses schrittweise, prüfen es dabei laufend und entwickeln so eine immer größere Software. Viele Programme, gerade aus dem Public Domain- und Freeware-Bereich, zeigen leider oft recht anschaulich das so entstandene Ergebnis. Eine anspruchsvolle Kernfunktionalität wird vollständig ausprogrammiert und getestet, Nebenfunktionalitäten sind nur teilweise vorhanden und die Dokumentation beschränkt sich in vielen Fällen auf die ursprünglichen Kernfunktionen.

Auf den Bereich der Softwareentwicklung für Chipkarten bezogen, eignet sich das Prototypen-Modell sehr gut für Studien und Machbarkeitsanalysen im Bereich Übertragungsprotokolle, Kryptoalgorithmen und Dateisysteme. Der erstellte Prototyp sollte jedoch nicht

1:1 in das Endprodukt übernommen werden, sondern muss auf den gleichen Qualitätsstandard wie die restliche Software-Entwicklung gebracht werden. Es können dabei jedoch durchaus größere Sourcecode-Abschnitte in das endgültige Produkt einfließen, sofern diese entsprechend den Vorgaben für die Gesamtrealisierung adaptiert wurden.

Die Entwicklung vom Prototypen sollte sich nur auf die risikobehafteten Programmcode Teile beschränken, um auf diese Weise Entwicklungskosten und Zeitaufwand gering zu halten. Um die Gefahr der Umdeklarierung eines Prototypen zu einem Produkt gering zu halten, realisieren erfahrene Entwickler üblicherweise immer nur Teilsysteme als Prototyp und niemals Gesamtsysteme.

15.7.1.4 Evolutionäres-/Inkrementelles Modell

Die bisher beschriebenen Vorgehensmodelle erfordern relativ genaue Anforderungsanalysen und Vorgaben. Diese können jedoch in bestimmten Fällen, etwa bei gänzlich neuen Softwarekonzepten nicht vollständig oder nur sehr unklar beim Beginn der Entwicklung ermittelt werden. Zudem ergeben sich beim Einsatz der Software neue Anforderungen und Wünsche seitens der Benutzer. Das evolutionäre bzw. inkrementelle Vorgehensmodell ermöglicht es, diese Art von Anforderungen für eine Software-Entwicklung zu berücksichtigen.

Bei beiden Vorgehensmodellen wird die Software in ihrer ganzen Breite und Tiefe in einer Generationenfolge entwickelt, beide sind also Code-getrieben. Zwischen den Generationen erfolgt jeweils eine Analyse der Benutzungserfahrung, welche dann als Vorgabe in die nächste Generation einfließt. Der Unterschied zwischen evolutionärem und inkrementellem Modell besteht darin, dass beim evolutionären Modell bei jeder Generation eine vollständige Analyse durchgeführt wird, beim inkrementellen Modell hingegen nur zu Beginn des gesamten Entwicklungsprozesses. Beide Modelle eignen sich sehr gut für unbekannte Vorgaben, da man die Funktionalität der Software schrittweise über die Generationenfolge an die notwendigen Anforderungen adaptieren kann. Beide Modelle sind Programmcode-getrieben, was bedeutet, dass das Produkt zwischen den Generationen im Echtbetrieb genutzt wird, anders als beim Prototypen-Modell.

Sowohl evolutionäres wie auch inkrementelles Modell führen zu einer Minimierung des Entwicklungsrisikos. Auch ist es möglich, die Entwicklungsrichtung innerhalb gewisser Grenzen während der Entwicklung zu korrigieren. Diese Vorteile erkaufte man sich jedoch durch höhere Entwicklungskosten und der Gefahr, dass durch die unvollständige Analyse am Anfang der Entwicklung bei späteren Generationen aufwendige Änderungen der Softwarearchitektur notwendig werden. Insbesondere die evolutionäre Vorgehensweise vergrößert dieses Problem zusätzlich, da bei jeder Generation eine neue Analyse durchgeführt wird. Beim inkrementellen Modell wird am Anfang der gesamten Entwicklung analysiert und anschließend werden zwischen den Generationen lediglich Differenzen zur Ausgangsanalyse erstellt. Die Änderungen von Generation zu Generation sollten deshalb keine gravierenden Auswirkungen auf die Software-Architektur haben.

Das evolutionäre/inkrementelle Modell lässt sich vorteilhaft bei vollständigen Neuentwicklungen mit Forschungscharakter einsetzen. Dabei kann über eine vorzugsweise häufige Generationenfolge schnell eine hohe Abdeckung der Anforderungen erreicht werden. Die beiden Vorgehensmodelle werden deshalb auch üblicherweise bei Neuimplementierungen von Chipkarten-Betriebssystemen verwendet. Ein anderes Haupteinsatzgebiet sind

neue, noch nicht vollständig fixierte Chipkartenanwendungen, die durch eine iterative Generationenfolge an die endgültigen Anforderungen angepasst werden.

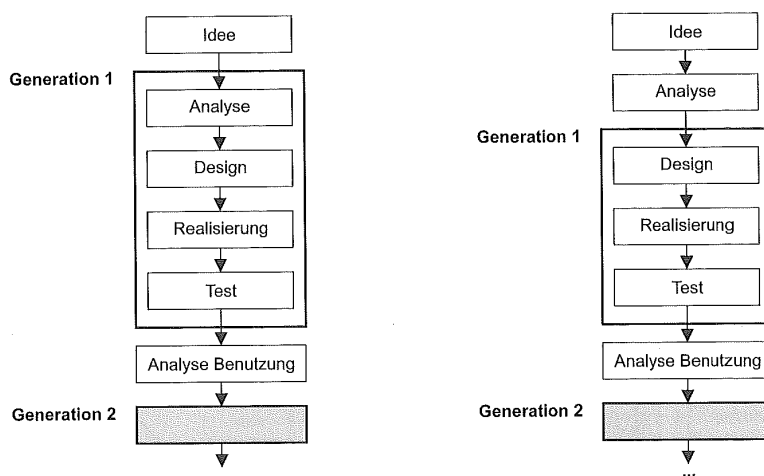


Bild 15.20 Auf der linken Seite ist im Überblick das auf die Belange der Software-Entwicklung für Chipkarten angepasste evolutionäre Prozess-Modell dargestellt. Die rechte Seite zeigt im Vergleich dazu überblickshaft das inkrementelle Prozess-Modell.

15.7.1.5 Spiralmodell

Das Spiralmodell ist ein Metamodell, d. h. ein Modell, in das die vorhergehenden Vorgehensmodelle integriert werden können, sodass es möglich wird, für die einzelnen Versionen einer Software jeweils das dafür am besten geeignete Vorgehensmodell zu benutzen. Das Spiralmodell hat seinen Namen von der spiralförmig auftragbaren Vorgehensweise, bei dem die Fläche innerhalb der Spirale den summierten Entwicklungskosten entspricht.

Der Entwicklungsablauf im Spiralmodell gliedert sich in vier Schritte: Im ersten Schritt werden die Ziele für die zu entwickelnde Software festgelegt, im darauffolgenden Schritt werden durch Risikoanalyse potenzielle Alternativen zum Erreichen der Ziele ermittelt. Im dritten Schritt wird die Software nach dem am besten geeigneten Vorgehensmodell entwickelt. Ist die gesamte Entwicklung damit noch nicht beendet, so wird im folgenden vierten Schritt der nächste Zyklus auf der Grundlage der vorangegangenen Schritte geplant. Anschließend folgt wieder der erste Schritt, d. h. die Festlegung der Ziele.

Grundsätzlich ist das Spiralmodell vor allem für größere Entwicklungsprojekte geeignet, da es aufgrund des komplexen Ablaufs einen verhältnismäßig hohen Koordinationsaufwand erfordert. Allerdings ist es dafür sehr flexibel an unterschiedliche Aufgaben anpassbar und für Entwicklungen über mehrere Versionen hinweg sehr gut geeignet.

Die Entwicklung eines Chipkarten-Betriebssystems nach dem Spiralmodell könnte beispielsweise folgendermaßen aussehen: Die Version 1, 2 und 3 des Betriebssystems wird mit dem evolutionären Modell entwickelt. Das Ergebnis sind drei Forschungsversionen, die zu Studien und Experimentierzwecken eingesetzt werden. Das Betriebssystem wird dabei schrittweise verfeinert und an die notwendigen Anforderungen angepasst. Die nächste Versi-

on mit der Nummer 4 wird dann nach dem V-Modell als auslieferbare Kundenversion entwickelt. Das V-Modell wurde hier deshalb gewählt, weil eine seiner Eigenschaften die sehr hohe Softwarequalität ist. Kundenversionen von Chipkarten-Betriebssystemen werden manchmal evaluiert (ITSEC oder CC) und die dafür benötigten Dokumente müssen größtenteils bei korrekter Benutzung des V-Modells ohnehin erstellt werden. In unserem Beispiel wäre dann die nächste Aufgabe der Software-Entwicklung die Portierung des Betriebssystems auf einen anderen Mikrocontroller. Da diese Aufgabe sehr geringe Risiken einschließt und auch wenig schöpferische Komponenten enthält, wird sie vorzugsweise nach dem Wasserfallmodell gelöst. Dieses Beispiel zeigt deutlich den Einsatz des Spiralmodells als Meta-Modell, mit dem jeweils optimalen Einsatz von entsprechenden Vorgehensmodellen, um die genannten fünf Versionen eines Chipkarten-Betriebssystems zu entwickeln.

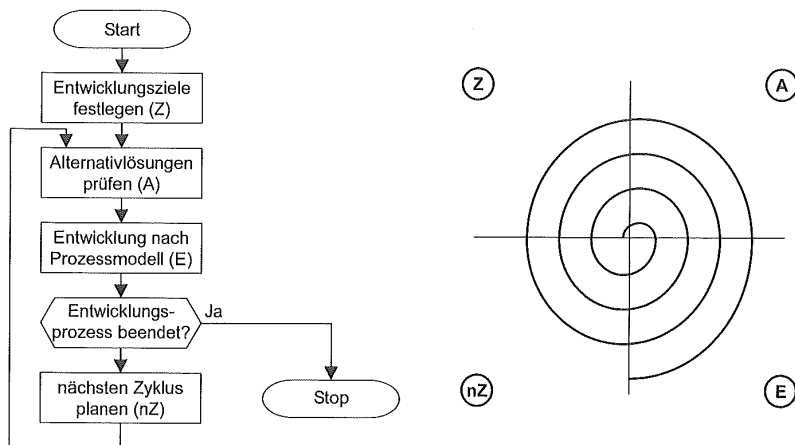


Bild 15.21 Überblickhafte Darstellung des auf die Belange der Software-Entwicklung für Chipkarten angepassten Spiralmodells. Die Fläche der Spirale ist in etwa äquivalent zum Entwicklungsaufwand.

Beispielsweise verwenden wir bei der Erstellung einer neuen Auflage des „Handbuchs der Chipkarten“ das Wasserfallmodell. Als Erstes erstellen wir eine Analyse – das Exposé –, welche Themen zu überarbeiten bzw. zu ergänzen sind und welche Kapitel gänzlich neu erstellt werden müssen. Anschließend entscheiden wir gemeinsam mit dem Verlag, wie die Änderungen und Ergänzungen durchgeführt werden. Dann folgt die Zeit des Textens, Zeichnens und Layoutens, und anschließend prüfen Lektor und Herstellung im Verlag, ob alles korrekt erstellt wurde. Die einzelnen Auflagen selbst sind wieder in einen evolutionären Prozess eingebettet, da bei jeder neuen Auflage eine vollständige Analyse durchgeführt wird.

15.7.2 Prozessreifegrad (*process maturity*)

Zur Bewertung der Qualität von Softwareentwicklungsprozessen in Organisationen existieren mehrere Verfahren. Das zurzeit bekannteste Verfahren dazu ist das Reifegradmodell nach CMM (*capability maturity model*). Das amerikanische Software Engineering Institu-

te (SEI) begann mit der Arbeit daran Ende 1986 und es wurde 1991 in einer ersten Version veröffentlicht. Die aktuell gültige Version hat die Versionsnummer 1.1 und stammt aus dem Jahre 1993 [CMM 93]. Neben CMM existiert noch ein relativ neues Bewertungsverfahren zu Prozessreifegraden, basierend auf der internationalen Norm ISO 15 504, das sich bisher in der Praxis nicht etabliert hat. Im Folgenden wird deshalb anhand von CMM überblickshaft das Thema erläutert.

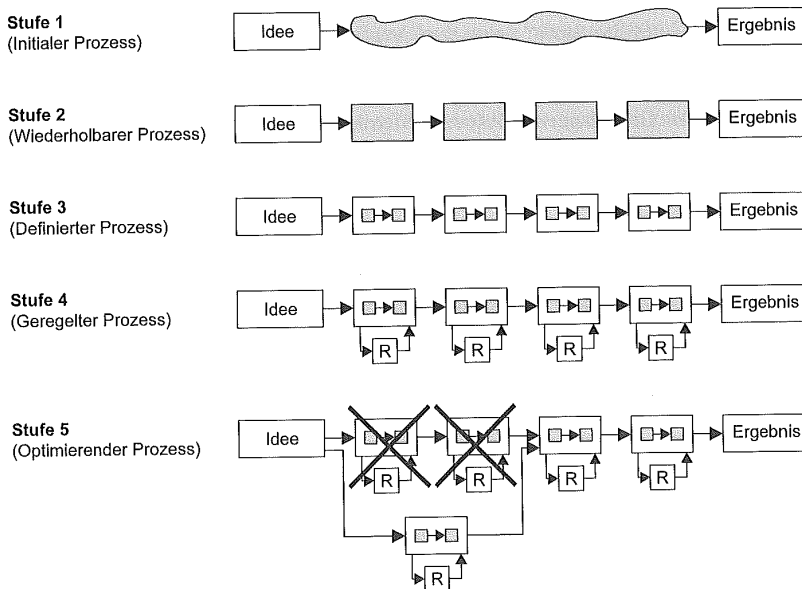


Bild 15.22 Die fünf Reifegradstufen von CMM im Überblick. Das eingekreiste „R“ symbolisiert einen Regler.

Im CMM existieren fünf Stufen der Prozessqualität, wobei Stufe 1 die qualitativ schlechteste und Stufe 5 die höchste Prozessqualität kategorisiert. Bei der ersten Stufe, genannt „Initiale Stufe“ (*initial level*), wird der Entwicklungseinheit eine Aufgabe gestellt und nach schwer vorhersagbaren Zeitaufwendungen, ungewissen Kosten und mit ebenfalls unvorhersehbarer Qualität ein Produkt fertig gestellt. Die einzelnen Schritte im Entwicklungsprozess sind nicht definiert und die Entwickler arbeiten mehr als Künstler denn als Handwerker oder Industriearbeiter. Durch großes Engagement einzelner Personen werden die zu entwickelnden Produkte in dem durchaus chaotisch zu nennenden Entwicklungsprozess manchmal noch knapp innerhalb der Grenzen von Zeit, Termin und Qualität fertig gestellt.

Bei Stufe 2, die „Wiederholbare Stufe“ (*repeatable level*) genannt wird, sind die einzelnen Aktivitäten in ihrem Rahmen festgelegt. Dies wären beispielsweise Analyse, Design, Realisierung und Test. Was im Detail innerhalb dieser generellen Tätigkeiten durchgeführt wird ist nicht festgelegt. Dies ist bei der nächsten Stufe, in der „Definierten Stufe“ (*defined level*) der Fall. Dort sind alle Tätigkeiten bei den Entwicklungsaktivitäten ausreichend genau vereinbart, sodass sie jederzeit nachvollzogen werden können. Die Qualität des Ent-

wicklungsprozesses ist somit deutlich höher und die Schwankungsbreite bei Kosten- und Termineinhaltung von Entwicklungsprojekten geringer als bei den vorherigen Stufen. Der Prozess bei Stufe 3 ist auch dadurch gekennzeichnet, dass er unabhängig von Individuen ist, da alle notwendigen Tätigkeiten festgelegt sind. Bei den Stufen eins und zwei hingegen ist eine große Personenabhängigkeit gegeben. Bei Stufe drei ist die Qualität der entwickelten Software noch unvorhersehbar, da der Prozess starr vorgegeben und nicht auf die verschiedenen Anforderungen hinsichtlich der Entwicklung anpassbar ist.

Stufe 4, „Geregelte Stufe“ (*managed level*), integriert einen Regelkreis innerhalb der einzelnen Aktivitäten, sodass der Entwicklungsprozess an die unterschiedlichen Forderungen von Entwicklungsprojekten anpassbar ist. Messdatenerfassung beim Entwicklungsprozess auf der Grundlage von Software-Metriken werden zur Regelung der einzelnen Prozessschritte verwendet.

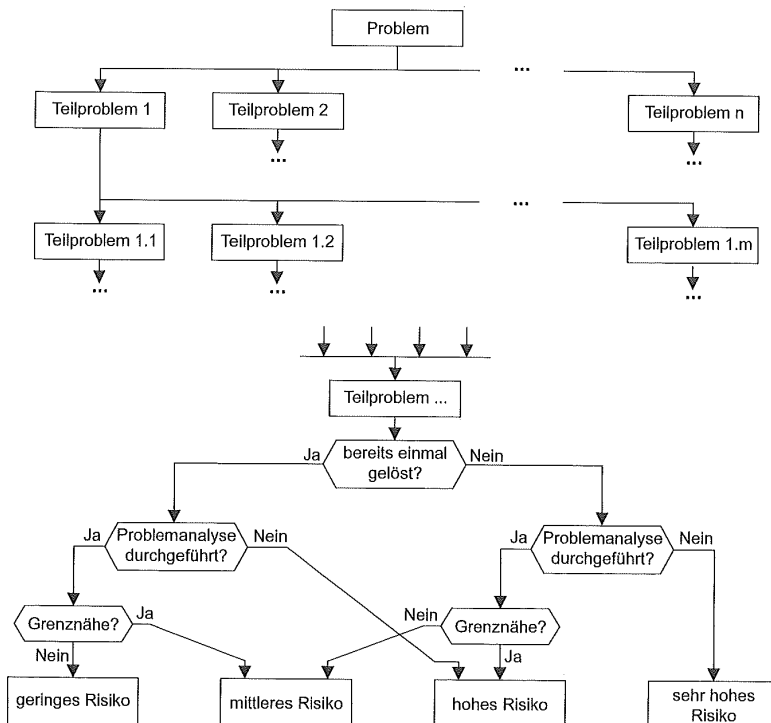


Bild 15.23 Flussdiagramm zur Aufteilung und Einstufung eines Problems in Risikoklassen. Dieses Flussdiagramm eignet sich sehr gut als Grundlage für Zeit- und Risikoabschätzungen von Entwicklungen. Idealerweise sollte man die Teilprobleme in eine Bearbeitungsdauer von zirka einer Woche aufspalten und innerhalb dieser Teilprobleme eine Zeit- und Risikoabschätzung durchführen.

Die höchste Qualitätsstufe, Stufe 5 – „Optimierende Stufe“ (*optimizing level*) – sieht nicht nur eine Regelung über die Einzelaktivitäten vor, sondern über mehrere Aktivitäten

hinweg. Diese können bei Bedarf durch neue, verbesserte, ersetzt werden, sodass der gesamte Entwicklungsprozess an unterschiedliche Anforderungen immer optimal angepasst werden kann. Durch die während der Entwicklungsprozesse protokollierten Messgrößen können die drei Kardinalkriterien Termin, Kosten und Qualität in angemessener Genauigkeit bei Entwicklungen eingehalten werden.

Das CMM mit seinen fünf Reifegraden ist selbstverständlich eine höchst abstrakte Darstellung von Entwicklungsprozessen. Die Beurteilung eines existierenden Prozesses (*assessment*) findet durch einen anonym zu beantwortenden Katalog mit etwa 100 Ja-/Nein-Fragen bei den Prozessbeteiligten statt. Dies soll zu einer ungeschönten Beantwortung und somit zu einer objektiven Beurteilung des Entwicklungsprozesses bei einer Organisation führen.

Die Prozessqualität bei den meisten Firmen liegt bei Stufe eins oder zwei. Stufe fünf über alle Entwicklungsaktivitäten haben weltweit nur einige wenige Organisationen, die dann meist auch in den typischen High-end-Softwarebereichen wie Luftfahrt, Kern- oder Militärtechnik tätig sind.

Unabhängig von der Prozessqualität kann jedoch das Softwareprodukt eines Entwicklungsprozesses mit schlechter Qualität durchaus äußerst innovativ und wirtschaftlich höchst erfolgreich sein. Die Wahrscheinlichkeit jedoch, dass Kostenziele, Qualitätsniveaus und Fertigstellungstermine eingehalten werden, ist jedoch umso niedriger, je schlechter die Prozessqualität ist. Im Extremfall kann es so weit kommen, dass das Entwicklungsprojekt abgebrochen werden muss, da entweder das veranschlagte Budget erschöpft ist, der Fertigstellungszeitpunkt inakzeptabel geworden ist oder große Teile der Software aufgrund zu hoher Komplexität niemals zufriedenstellend funktionieren werden. Diese Risiken sind umso geringer, je höher die Qualität des Entwicklungsprozesses ist.

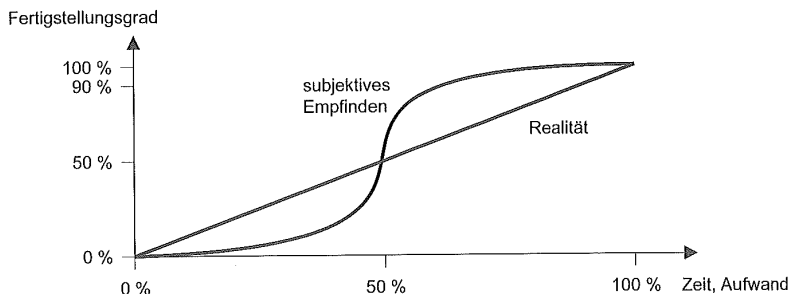


Bild 15.24 Realer, d. h. gemessener Verlauf der Fertigstellung einer Software im Vergleich zum üblichen subjektiven Empfinden der beteiligten Entwickler. Der Entwicklungsaufwand in der Grafik wird für die gesamte Entwicklungszeit als konstant angenommen.

15.8 Ablauf eines Chipkarten-Projekts

Aufgrund der aktuellen Anbietersituation bei Chipkarten entwickeln die Kartenhersteller auch die dazugehörige Software für den Mikrocontroller. Somit führt die erste Phase (A) und die letzte Phase (E) die gleiche Firma durch. Danach erfolgen die Phasen B und C bei einem Halbleiterhersteller. Der Einbau der Dice in Module und das Bonden kann dann entweder noch direkt beim Halbleiterhersteller oder schon beim Kartenhersteller gemacht

werden. Die gesamte Phase E, also Initialisierung, Personalisierung usw. erfolgt auf jeden Fall beim Kartenhersteller. Dieser leitet auch üblicherweise die Chipkarten-Projekte; die anderen Firmen sind nur mehr oder weniger Unterauftragnehmer.

Der Ablauf enthält sehr wenige Informationen über den Zeitbedarf der jeweiligen Fertigungsschritte. Dieser ist jedoch nicht zu unterschätzen, da mehrere unterschiedliche Firmen zusammenarbeiten und teilweise Prozesse durchlaufen werden müssen, die sich über viele Wochen erstrecken. Aus diesem Grund sind in der folgenden Tabelle einige typische Werte für die Durchlaufzeit der wichtigsten Produktionsschritte aufgeführt.

Für das Beispiel gelten die folgenden Rahmenparameter. Angenommen sei die fiktive Fertigung von 50000 Chipkarten. Das Betriebssystem ist zum Großteil auf der Grundlage von vorhandenen Bibliotheken neu aufzubauen. Alle beteiligten Firmen haben mittlere Kapazitäten, und die jeweiligen Fertigungsprozesse können unmittelbar nach Erhalt der notwendigen Teile beginnen.

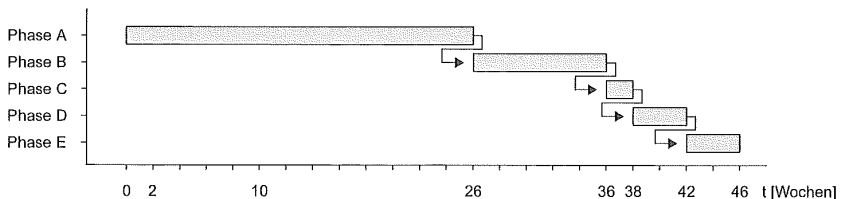


Bild 15.25 Gantt-Diagramm für den Ablauf eines typischen Chipkartenprojekts:

Phase A	Maskenerstellung	6	Monate
Phase B	Halbleiterherstellung	10	Wochen
Phase C	Modulherstellung	2	Wochen
Phase D	Kartenkörperherstellung	4	Wochen
Phase E	Initialisierung, Personalisierung	4	Wochen

15.9 Beispiele für die Konzeption von Chipkarten-Anwendungen

Es gibt zwei grundsätzliche Arten, Anwendungen auf einer Chipkarte zu realisieren: Einerseits Anwendungen, die auf Dateien und festgelegten Zugriffsbedingungen aufbauen. Die dazu notwendigen Kommandos entsprechen dabei in aller Regel den üblichen Chipkarten-Normen wie ISO/IEC 7816-4. Die andere Art der Anwendungsrealisierung basiert auf Programmcode, der in der Chipkarte aufgeführt wird, wobei es hier mehrere Varianten gibt. Der Programmcode kann in der Chipkarte entweder direkt vom Prozessor ausgeführt (nativer Code) oder interpretiert werden. Bei der Interpretation kann man noch zwischen Mikrobrowsern (z. B.: XML-Derivate) und Virtual Machines (z. B.: Java) unterscheiden. Details sowie Vor- und Nachteile der einzelnen Realisierungsmöglichkeiten für Anwendungen finden sich im Kapitel 5 Chipkarten-Betriebssysteme.

In den folgenden Beispielen werden drei typische Chipkarten-Anwendungen aufgezeigt. Diese sind im mittleren Bereich der EDV angesiedelt und verlangen keine großen Systemkonzepte. Sie könnten etwa in mittelständischen Unternehmen zum Einsatz kommen. Das dabei verwendete Hintergrundsystem kann ein in einer sicheren Umgebung aufgestellter

PC sein. Die Installationskosten und auch die laufenden Kosten sind damit ebenfalls im unteren Bereich angesiedelt.

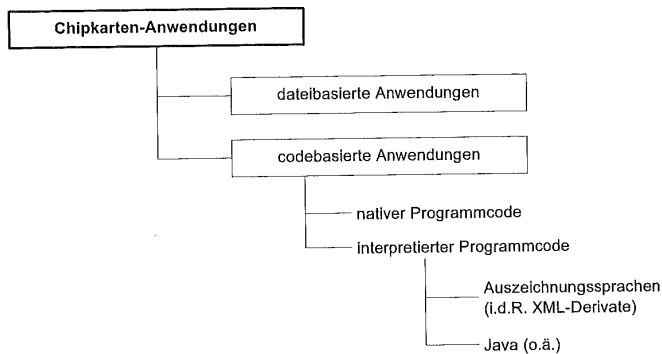


Bild 15.26 Klassifizierungsbaum der prinzipiellen Möglichkeiten zur Realisation von Chipkarten-Anwendungen.

Anhand dieser einfachen Beispiele kann sehr gut gezeigt werden, wie man eine typische Chipkarten-Anwendung aufbaut. Dies ist Schritt für Schritt erklärt und der stufenweise Aufbau hin bis zur fertigen Anwendung in der Chipkarte demonstriert. Die Seite der Terminals und des Hintergrundsystems ist jeweils immer nur im Überblick abgehandelt, kann jedoch aus den vorhandenen Informationen abgeleitet werden.

Die Grundlage dieser Beispiele ist die Verteilung von Informationen auf viele individuelle Einzelsysteme. Dies steht oft im Gegensatz zu zentralistisch aufgebauten Großrechnerlösungen, bei denen sich alle Informationen lediglich an einem Ort befinden. Auf Chipkarten-Anwendungen übertragen, hieße dies, dass die Karte nur eine Art Identitätsnachweis ist und alle Informationen in einem allwissenden Hintergrundsystem gespeichert sind. Die Erweiterung einer solchen Anwendung führt dann regelmäßig dazu, dass das alles steuernde Hintergrundsystem mit hohem Kosten- und Zeitaufwand ebenfalls ausgebaut werden muss.

Es wurde hier versucht, einen anderen Weg zu gehen. Das Hintergrundsystem ist nur für die Verwaltung und die Konsistenz des Gesamtsystems verantwortlich. Alle weiteren Informationen sind lokal in den Chipkarten abgelegt. Natürlich benötigt man eine globale Datenbasis, um das System zu verwalten, also verlorene oder kaputte Chipkarten mit den enthaltenen Daten neu herstellen zu können. Doch diese Datenbasis darf zum Betreiben des Systems nicht benötigt werden.

Man kann sich ein verzweigtes Chipkarten-System als einen großen Baum vorstellen. Dieser Baum bezieht, wie alle Bäume, seine Energie aus der Photosynthese in den vielen Blättern. Die Energiegewinnung ist also in die vielen Blätter des Baumes verlagert, wo sie parallel stattfindet. Eine effektive und gute Chipkarten-Anwendung verhält sich im übertragenen Sinn genauso. Die Informationen sind dezentral in den Chipkarten gespeichert und somit sicher gegen jegliche Angriffe. Damit belastet diese Informationsmenge auch nicht das Hintergrundsystem, das zentral nur die anfallenden Verwaltungsfunktionen übernimmt. Die eigentlichen Systemabläufe finden, genau wie in den Blättern eines Baumes, parallel und dezentral in den Terminals und Chipkarten vor Ort statt. Damit kann das Ge-

samtsystem durch zusätzliche Terminals und Chipkarten sehr einfach erweitert werden, ohne dass größere Auswirkungen auf den oder die Hintergrundrechner zu befürchten sind.

Das Gegenteil zu diesem skizzierten System ist, die gesamte anfallende Arbeit zentralistisch im Hintergrundsystem zu erledigen. In übertragenem Sinne würde man also die Photosynthese von den Blättern des Baumes zum Stamm hin verlagern, was zu einem sehr mächtigen Stamm führen würde. Das Gesamtsystem wäre damit also nicht nur sehr groß und teuer, sondern auch noch extrem anfällig gegenüber Störungen am Hintergrundrechner. Dies sollte man aber soweit wie möglich vermeiden.

Viele angehende Systembetreiber begehen den Fehler, dass sie in Unkenntnis der Eigenschaften von Chipkarten ein Gesamtsystem von oben nach unten, also top down, entwerfen. In den sicherheitssensiblen Bereichen, nämlich bei den Terminals und den Chipkarten, setzen sie dann oft voraus, dass diese auf mehr oder minder unbekannte Weise sicher gemacht werden können.

Die hier vorgestellten Entwürfe vertreten in dieser Hinsicht eine Bottom-up-Strategie, bei der vom untersten Objekt, der Chipkarte, nach oben das System und die erforderlichen Eigenschaften des Systems definiert werden. Die Gefahr von Sicherheitslücken lässt sich auf diese Weise sehr gut minimieren, da von der kleinsten Einheit ausgehend das System sicher nach oben hin aufgebaut wird.

15.9.1 Börse für Spielautomat

Ausgangssituation

Die geforderte Funktionalität dieser Chipkarten-Anwendung soll es ermöglichen, Zahlungen von kleineren Beträgen an Spielautomaten zu tätigen. Das dabei üblicherweise verwendete Münzgeld soll durch Chipkarten ersetzt werden, um so die Betriebskosten zu senken.

Es existieren zwei Arten von Terminals. Das Ladeterminale verfügt über einen Münzeinwurf mit Prüfer und kann elektronische Geldeinheiten in die Chipkarte laden. Am weitgehend autark arbeitenden Abbucheterminale werden die Geldeinheiten von der Chipkarte abgebucht.

Vorgaben

Das gesamte System soll anonym sein, wobei eine Überwachung aller Zahlungsströme möglich sein muss. Bei Verdacht auf Betrug müssen Karten identifiziert und selektiv gesperrt werden können.

Aufgrund der geforderten Anwendung im Zahlungsverkehrsbereich und der Benutzung an vollautomatischen und nicht von Menschen überwachten Geräten sollte ein mittleres Sicherheitsniveau angestrebt werden.

Lösungsvorschlag

Die Lösung basiert auf einer einfachen geschlossenen Börse, welche speziell für diesen einen Anwendungszweck gestaltet ist. Sie könnte natürlich auch für ähnliche Anwendungsfälle mit leicht geänderten Dateien und Abläufen benutzt werden. Der Einsatz einer elektronischen Geldbörse nach CEN EN 1546 wurde hier vermieden, da dies für den Anwendungsbetreiber aufwendiger als die vorgeschlagene Lösung ist. Außerdem soll das Prinzip einer einfachen geschlossenen Börse demonstriert werden.

An zentraler Stelle wird ein Automat aufgestellt, an dem vollautomatisch sowohl Münzen als auch Geldscheine einbezahlt und dann als elektronische Geldeinheiten auf die Chipkarte geladen werden können. Dazu sind weder PIN- noch andere Benutzereingaben erforderlich, da die elektronische Geldbörse anonym ist. Lediglich anhand einer im System einzigartigen Kartennummer findet eine zusätzliche Buchführung der geladenen Beträge statt.

Tabelle 15.11 Dateibaum für die Beispielanwendung „Spielautomat“

Datei	FID	Struktur	Beschreibung und Dateinhalt
MF	'3F00'	---	Wurzelverzeichnis für die Chipkarte
DF	---	---	Verzeichnis für die Anwendung „Spielautomat“
DF.EF 1	'0001'	transparent	Ausgabedatum Kartennummer
DF.EF 2	'0002'	cyclic	Betrag
DF.EF 3	'0003'	linear fixed	Schlüssel 1 Schlüssel 2 Schlüssel 3 Schlüssel 4

Die Verwaltung aller Daten und der Zahlungsströme übernimmt ein PC. Auf diesem befindet sich auch eine Datenbank, in der alle herausgegebenen Chipkarten mit ihren Rahmendaten gespeichert sind. Durch tägliche oder wöchentliche Bilanzierungsrechnung kann geprüft werden, ob die Geldströme des Systems noch geschlossen sind.

Am Bezahlungsterminal werden die geladenen Einheiten von der elektronischen Geldbörse in der Chipkarte abgebucht. Über ein Display an diesem Terminal erhält der Benutzer Informationen über die abgebuchten Beträge. Um den Preis des Terminals möglichst niedrig zu halten, benutzt man zur Sicherung der Datenübertragung keinen Shutter, sondern Secure Messaging. Jedes Terminal besitzt ein Sicherheitsmodul für die geheimen Schlüssel und die Buchführung der nach Kartennummern sortierten Beträge. In regelmäßigen Zeitabständen werden die dabei ermittelten Daten über eine Leitung oder über eine spezielle Transferkarte zum Verwaltungs-PC übertragen, welcher dann die Auswertung übernimmt. Der Lösungsvorschlag sieht den folgenden Dateibaum in der Chipkarte vor. Die Anwendung benötigt im Übrigen je nach Chipkarten-Betriebssystem etwa 100 Byte im EEPROM.

Da in regelmäßigen Zeitabständen ein Datenaustausch zwischen Bezahlungsterminal und Verwaltungsrechner stattfindet, kann damit gegebenenfalls auch eine Sperrliste im Terminal auf dem neuesten Stand gehalten werden. Stellt das Terminal fest, dass sich eine gesteckte Chipkarte auf der Sperrliste befindet, blockiert es die Datei EF 2 mit dem elektronischen Geldbetrag. Somit ist ein Bezahlen mit der Chipkarte nicht mehr möglich. Der Benutzer muss an einem Verwaltungsterminal seine Chipkarte entsperren lassen. Bei dieser Gelegenheit kann auch überprüft werden, warum diese auf die Sperrliste gesetzt wurde.

Um die Anwendung zu betreiben, sind die folgenden Schlüssel notwendig, wobei auf die Verwendung von abgeleiteten und somit kartenindividuellen Schlüsseln zugunsten eines einfacheren Gesamtsystems verzichtet wurde.

Tabelle 15.12 Notwendige Schlüssel für die Beispielsanwendung „Spielautomat“

Schlüssel	Verwendung	Funktion	Zustands- übergang
Schlüssel 1	MUTUAL AUTHENTICATION	gegenseitige Authentisierung von Terminal und Chipkarte – Bezahlen mit der Börse – Sperren der Börse	$x \rightarrow 1$
Schlüssel 2	MUTUAL AUTHENTICATION	gegenseitige Authentisierung von Terminal und Chipkarte – Entsperren der Börse – Verwaltung der Karte	$x \rightarrow 2$
Schlüssel 3	MUTUAL AUTHENTICATION	gegenseitige Authentisierung von Terminal und Chipkarte – Aufladen der Börse	$x \rightarrow 3$
Schlüssel 4	Secure Messaging	Sicherung der Datenübertragung – Authentic-Modus	---

Die vorgeschlagene Lösung lässt sich sehr gut für die Bezahlung erhaltener Leistungen an Automaten einsetzen. Eine menschliche Überwachung ist dazu nicht notwendig. Der Geldbetrag muss aber nicht unbedingt vollautomatisch an einem speziellen Automaten in die Chipkarte geladen werden, dies kann auch an einer Kasse nach Bezahlung des Aufladebetrags manuell geschehen.

Das hier skizzierte System ließe sich mit geringfügigen Anpassungen auch in einem Waschsalon oder in einer Kantine einsetzen.

Tabelle 15.13 Zugriffsbedingungen für die Anwendung „Spielautomat“
(≥ 0 : immer, < 0 : nie, SM: Secure Messaging).

Datei	Lesen	Schreiben	Sperren	Entsperren	Erhöhen des Betrags	Erniedrigen des Betrags
EF 1	≥ 0	$= 2$	< 0	< 0	< 0	< 0
EF 2	$\geq 0 \wedge \text{SM}$	< 0	$= 1$	$= 2$	$= 3$	$= 1$
EF 3	< 0	< 0	$= 1$	$= 2$	< 0	< 0

Beim Laden von elektronischen Geldeinheiten findet der im Folgenden grob skizzierte Ablauf statt:

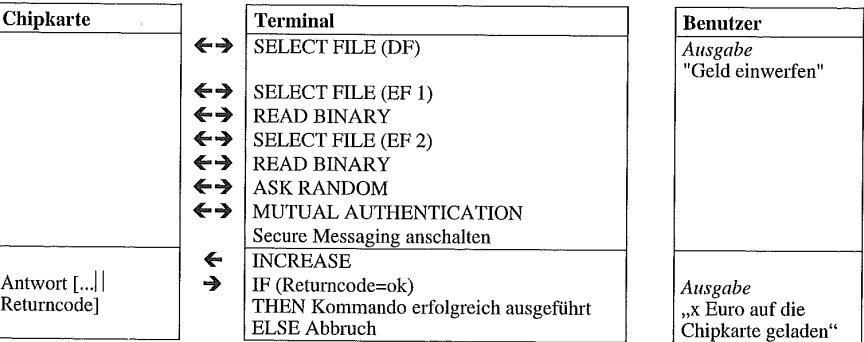


Bild 15.27 Prinzipieller Ablauf der Kommandos beim Laden von elektronischen Geldeinheiten in der Anwendung „Spielautomat“.

Chipkarte		Terminal		Benutzer
	↔	SELECT FILE (DF)		
	↔	SELECT FILE (EF 1)		
	↔	READ BINARY		
	↔	SELECT FILE (EF 2)		
	↔	READ BINARY		
	↔	ASK RANDOM		
	↔	MUTUAL AUTHENTICATION		
		Secure Messaging anschalten		
Antwort [... Returncode]	←	DECREASE	→	Ausgabe "x Euro auf der Chip- karte abgebucht"
		IF (Returncode = ok) THEN Kommando erfolgreich ausgeführt ELSE Abbruch		

Bild 15.28 Prinzipieller Ablauf der Kommandos beim Abbuchen von elektronischen Geldeinheiten in der Anwendung „Spielautomat“.

15.9.2 Zugangskontrolle

Ausgangssituation

Der Zweck dieser Anwendung ist es, eine abgestufte Zugangskontrolle für verschiedene Räume und Rechnersysteme auf der Grundlage von Chipkarten zu schaffen. Es befinden sich also an bestimmten Türen und Rechnern Terminals, die die entsprechenden Türen oder Rechner nach einer Kommunikation mit einer Karte freischalten können. Dabei ist es wichtig, dass die Möglichkeit besteht, Sicherheitsniveaus zu definieren, zu denen nur ein bestimmter Personenkreis Zutritt erlangen darf. Der Zutritt wird auf der Grundlage einer Authentisierung und eines Identitätsnachweises des Benutzers gewährt. Dieser Nachweis wird zum einen durch den Besitz einer echten Karte und zum anderen durch die Kenntnis der dazugehörigen PIN geführt. Sind diese beiden Kriterien erfüllt, so ist ein Zugang erlaubt. Die Terminals müssen einfache Sperrlisten führen können, um den Zutritt durch „verlorene“ Karten zu verhindern und diese Karten dann gegebenenfalls irreversibel zu sperren.

Vorgaben

Um die Akzeptanz der Lösung bei den Benutzern möglichst groß zu halten, darf der Kommunikationsablauf zwischen Terminal und Chipkarte und die darauf folgende Freigabe die Zeitdauer von einer Sekunde nicht signifikant überschreiten. Jeder längere Ablauf würde über kurz oder lang die Akzeptanz der Lösung erheblich stören und die Benutzer dazu verleiten, mit etwaigen Tricks (Keil in der Tür) die Sicherheit zu umgehen. Um auf die Karte geschriebene PINs als Gedächtnisstütze zu vermeiden, muss die PIN vom Benutzer änderbar sein.

Das System sollte auf ein unteres Sicherheitsniveau ausgelegt sein, da mit hoher Wahrscheinlichkeit keine aufwendigen Angriffe oder Analysen gemacht werden. Die Kosten für die Einführung und den Betrieb dürfen nicht wesentlich über einem guten Schlüsselsystem liegen, da dieses sonst eine wirtschaftlich günstigere Alternative darstellen würde. Das

System und die Chipkarte muss für die spätere Einbindung der Zeitabrechnung und Kantenabrechnung ausgelegt sein.

Lösungsvorschlag

An den entsprechenden Türen und Rechnern werden einfache Terminals mit einer Zehn-Ziffern-Tastatur fest montiert. Die Terminals können autark arbeiten und sind mit einem kostengünstigen und austauschbaren Sicherheitsmodul (z. B. Chipkarte im Plug-in-Format) ausgestattet und können bei Berechtigung die jeweilige Tür oder den Computer freischalten. Alle Terminals an zentralen und sicherheitssensiblen Stellen können bei Bedarf über eine Zweidrahtleitung selbstständig zu dem als Zentralrechner dienenden PC in Online-Verbindung geschaltet werden. Durch diesen einfachen Aufbau wird die Vorgabe der niedrigen Betriebskosten erfüllt.

Der Zentralrechner ist mit einem einfachen Multitasking-Betriebssystem ausgestattet, sodass er mehrere Aufgaben parallel erledigen kann. Er ist mit einem zusätzlichen Terminal verbunden, das für die gesamte Verwaltung des Systems zuständig ist.

Die verwendete Chipkarte muss über ein Betriebssystem verfügen, das mehrere Anwendungen verwalten kann und in der Lage ist, nachträglich Dateien (DFs und EFs) im Dateibaum der Chipkarte zu erzeugen, um damit bei Bedarf weitere Anwendungen nachzuladen. Der für die aktuelle Anwendung und die zukünftig projektierten Anwendungen notwendige EEPROM-Speicher wird 1 kByte nicht überschreiten. Die Chipkarten werden von einem Hersteller komplett mit Betriebssystem bezogen und auf dem Verwaltungsrechner entsprechend eingerichtet.

Alle Karten sind ursprünglich mit einer einheitlichen leicht zu merkenden PIN vorbelegt. Die PIN "0000" ist für diesen Fall am einfachsten. Damit entfällt jeglicher Aufwand der PIN-Generierung und der Zustellung von PIN-Briefen. Jeder Benutzer muss diese Einheits-PIN am Verwaltungsterminal bei Erhalt der Karte ändern, denn die Terminals lassen eine PIN-Eingabe von "0000" nicht zu.

Vergisst der Benutzer seine PIN oder der Fehlbedienungszähler läuft ab, dann kann am Systemterminal nach einer Authentisierung wieder eine neue PIN mit einem auf null zurückgesetzten Fehlbedienungszähler eingetragen werden.

Da das geforderte Sicherheitsniveau im mittleren Bereich liegen und die Verwaltung nicht zu aufwendig sein soll, empfiehlt sich ein stark reduziertes Schlüsselmanagement. Abgeleitete Schlüssel oder mehrere Schlüsselgenerationen finden deshalb keine Verwendung. Die Schlüssel sollten lediglich nach ihrer Funktion getrennt sein, was zur folgenden Aufteilung führt:

Tabelle 15.14 Notwendige Schlüssel für die Anwendung „Zugangskontrolle“

Schlüssel	Verwendung	Funktion	Zustandsübergang
Schlüssel 1	MUTUAL AUTHENTICATION	Verwaltung der Anwendung • Erzeugung neuer Dateien • Schreiben in Dateien • Entsperren der Anwendung	x → 1
Schlüssel 2	MUTUAL AUTHENTICATION	gegenseitige Authentisierung von Terminal und Chipkarte • Zugangsberechtigung • Sperren der Anwendung	x → 2
PIN	VERIFY CHV	Identifizierung des Benutzers	2 → 3

Der folgende Dateibaum muss in der Karte vorhanden sein:

Tabelle 15.15 Dateibaum für die Beispielanwendung „Zugangskontrolle“

Datei	FID	Struktur	Beschreibung und Dateinhalt
MF	'3F00'	---	Wurzelverzeichnis für die Chipkarte
DF	---	---	Verzeichnis für die Anwendung „Zugangskontrolle“
DF.EF 1	'0001'	transparent	Name, Vorname, Abteilung
DF.EF 2	'0002'	linear fixed	Berechtigungen
DF.EF 3	'0003'	linear fixed	Schlüssel 1 Schlüssel 2 PIN

Für den in der Chipkarte enthaltenen Dateibaum sind die folgenden Zugriffsbedingungen definiert:

Tabelle 15.16 Zugriffsbedingungen für die Beispielanwendung „Zugangskontrolle“
(≥ 0: immer, < 0: nie)

Datei	Lesen	Schreiben	Sperren	Entsperren	Erzeugung von DFs und EFs
DF	---	---	---	---	= 1
EF 1	≥ 0	= 1	< 0	= 1	---
EF 2	= 3	= 1	= 2	= 1	---
EF 3	< 0	< 0	= 2	= 1	---

Die Datei EF 2 mit den Berechtigungsdaten für Räume und Rechner ist Recordorientiert aufgebaut. Alle Records haben eine einheitliche Länge (linear fixed). Bei jedem Record befindet sich ein Eintrag, der angibt, zu welchen Räumen der Kartenbenutzer Zutritt hat. Damit hier nicht jeder einzelne Raum explizit angegeben werden muss, können auch Sicherheitsniveaus definiert werden. Mit diesen lässt sich der Zutritt dann pauschal auf einen bestimmten Bereich beschränken.

Da erfahrungsgemäß gerade bei der Zugangskontrolle oft Änderungen und Umstrukturierungen notwendig sind, sollten die Dateiinhalte der Records konsequent über eine TLV-Struktur verfügen. Damit können Erweiterungen oder Änderungen technisch sehr elegant vorgenommen werden.

Für diese Karte werden nur Standardkommandos eines auf dem Markt erhältlichen Betriebssystem nach ISO oder ETSI verwendet. Dabei muss in der Chipkarte nichts programmiert werden, was die Anschaffungskosten erheblich senkt. Die für die Karte notwendigen Kommandos sind wie folgt:

```
ASK RANDOM
CHANGE CHV
CREATE
INVALIDATE
MUTUAL AUTHENTICATION
READ BINARY
```

REHABILITATE
SELECT FILE
UNBLOCK CHV
VERIFY CHV
WRITE BINARY

Der typische Ablauf bei der Benutzung der Chipkarte für die Zugangskontrolle funktioniert gemäß dem folgenden Bild:

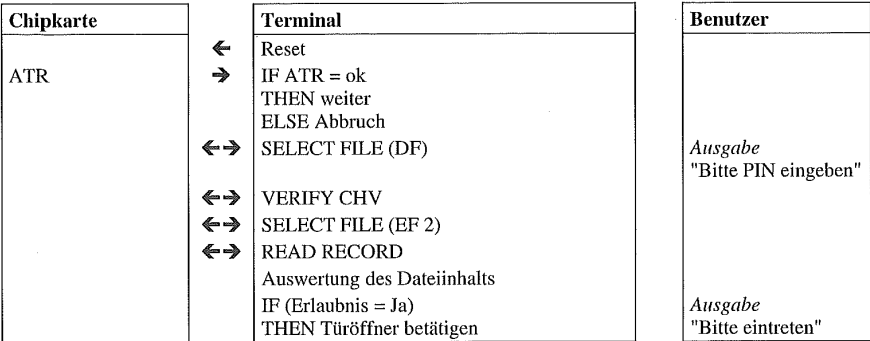


Bild 15.29 Ablauf der Kommandos bei der Anwendung „Zugangskontrolle“.

Wenn nötig, kann das Terminal direkt nach dem ATR den Benutzernamen mit READ BINARY aus der Datei lesen und mit einer Sperrliste vergleichen. Ist der Name darin enthalten, besteht die Möglichkeit, durch Sperrung aller EFs mit INVALIDATE die weitere Benutzung der Chipkarte für die Zugangskontrolle zu unterbinden. Am Verwaltungsterminal kann dann bei Bedarf nach gegenseitiger Authentisierung mit REHABILITATE die Anwendung reaktiviert werden.

Falls sich der Betreiber des Systems entschließt, die Kantinenabrechnung ebenfalls über die Chipkarte durchzuführen, muss eine neue Anwendung mit DF und EFs erzeugt werden. Dies ist auf zweierlei Art möglich: Entweder müssen sich alle Mitarbeiter mit ihrer Chipkarte zum Verwaltungsterminal begeben, oder die Dateien werden automatisch bei einer Zugangskontrolle nachgeladen. Der zweite Weg ist sicher der benutzerfreundlichere und kostengünstigere, da kein zusätzlicher Verwaltungsaufwand anfällt.

Die ebenfalls geforderte Kontrolle des Zugangs zu Computersystemen kann völlig analog zu dem oben beschriebenen ablaufen. Der einzige Unterschied besteht darin, dass kein Türöffner betätigt wird, sondern ein Signal dem Computer mitteilt, dass der Zugang zu gewähren ist.

15.9.3 Prüfung auf Echtheit eines Terminals

Ausgangssituation

Es gibt Szenarien, wonach der Kartenbenutzer die Möglichkeit haben sollte, die Authentizität des Terminals zu prüfen. Als Beispiel sei hier ein Terminal an einer Supermarktkasse angeführt, in das nach dem Stecken der Chipkarte die PIN eingegeben wird. Ein gefälschtes

Terminal könnte nun dazu benutzt werden, um die geheime PIN⁸ auszuspähen. Anschließend wird die Chipkarte gestohlen, und der Dieb kann nun aufgrund seiner Kenntnis der PIN damit bezahlen oder sich sogar Bargeld am Geldausgabeautomaten auszahlen lassen.

Im Sommer 1997 wurden mit einem gefälschten Geldausgabeautomaten auf dem Münchner Marienplatz und bei einer vergleichbaren Vorgehensweise Magnetstreifendaten mit dazugehöriger PIN ausgespäht. Mit Chipkarten kann durch eine entsprechende Anwendungsgestaltung ein guter Schutz gegen diese Art von Angriffen erreicht werden.

Vorgaben

Es ist ein Teil einer Chipkarten-Anwendung zu konzipieren, der es einem Kartenbenutzer ermöglicht, ein gefälschtes – kein manipuliertes Chipkarten-Terminal – zu erkennen. Der Benutzer darf dabei zur Prüfung des Terminals keine zusätzlichen technischen Einrichtungen benötigen.

Lösungsvorschlag

Der im Folgenden aufgeführte Lösungsvorschlag sieht vor, dass die Chipkarte ein nur dem Kartenbenutzer bekanntes Erkennungswort in einer Datei enthält. Diese Datei ist für das Terminal nur dann lesbar, wenn es sich erfolgreich mit einem geheimen Schlüssel gegenüber der Chipkarte authentisiert hat.

Das Terminal darf nach diesem Authentisierungsvorgang das Erkennungswort aus der Datei lesen und auf dem Display darstellen. Sobald der Benutzer das nur ihm bekannte Erkennungswort sieht und es korrekt ist, kann er davon ausgehen, dass es sich um ein echtes Terminal handelt. Erst dann wird er seine geheime PIN eingeben und somit den weiteren Transaktionsablauf ermöglichen.

Eine wichtige Einschränkung ist jedoch bei dieser Lösung anzuführen. Auf diese Weise kann nur ein gefälschtes Chipkarten-Terminal und nicht ein manipuliertes erkannt werden. Ist es möglich, die Terminalsoftware ohne Verlust der geheimen Schlüssel zu verändern, dann ist auch im Manipulationsfall eine korrekte Authentisierung des Terminals gegenüber der Chipkarte möglich, und auch die Kennwortdatei darf anschließend gelesen werden. Diese Restriktion der Lösung ist im jeweiligen Anwendungsfall zu bedenken. Sie ist jedoch im Grunde genommen unproblematisch, da eine derartig große Manipulationsmöglichkeit des Terminals noch wesentlich weitergehende Angriffsvarianten zulässt als die bloße Ausspähung einer PIN.

Das oben beschriebene Verfahren wird beispielsweise auch in der DIN-Spezifikation für die deutsche Signaturkarte vorgeschlagen, um die Echtheit öffentlicher Signaturterminals für den Kartenbenutzer feststellbar zu machen.⁹

Der nachfolgend mit konkreten Dateien und den dazugehörigen Zugriffsbedingungen ausgeführte Lösungsvorschlag ist keine vollständige Chipkarten-Anwendung, sondern ein Rumpf im Sinne eines Design-Patterns, der sich in beliebige Anwendungen einflechten lässt. Deshalb sind FIDs, Zustandsübergänge und der aufgeführte Ablauf je nach Bedarf auf verschiedene Werte oder Abfolgen adaptierbar. Das Beispiel soll vor allem die grundlegende Idee der „Prüfung auf Echtheit eines Terminals“ vermitteln und keine konkrete Anwendung.

⁸ Siehe auch Abschnitt 8.1.1 „Prüfung einer Geheimzahl“.

⁹ Siehe auch Abschnitt 14.4 „Digitale Signatur“.

Tabelle 15.17 Notwendige Schlüssel für die „Prüfung auf Echtheit eines Terminals“

Schlüssel	Verwendung	Funktion	Zustandsübergang
Schlüssel 1	EXTERNAL AUTHENTICATION	Authentisierung des Terminals durch die Chipkarte	x → 1
PIN	VERIFY CHV	Identifizierung des Benutzers	1 → 2

Nachdem der Benutzer das Terminal als echt erkannt hat und auch seine PIN von der Chipkarte als die Richtige erkannt wurde, sollte man die Möglichkeit schaffen, das Kennwort zu ändern. Dies kann durch entsprechendes Setzen der Zugriffsbedingungen für den schreibenden Zugriff auf die Kennwortdatei problemlos realisiert werden. Der folgende Dateibaum muss in der Karte vorhanden sein:

Tabelle 15.18 Dateibaum und Zugriffsbedingungen für die „Prüfung auf Echtheit eines Terminals“ (≥ 0: immer, < 0: nie)

Datei	FID	Struktur	Lesen	Schreiben	Dateinhalt
EF 1	'0001'	transparent	= 1	= 2	Kennwort
EF 2	'0002'	linear fixed	< 0	< 0	Schlüssel 1, PIN

Der typische Ablauf bei der Benutzung der Chipkarte für die Zugangskontrolle funktioniert nach dem folgenden Bild:

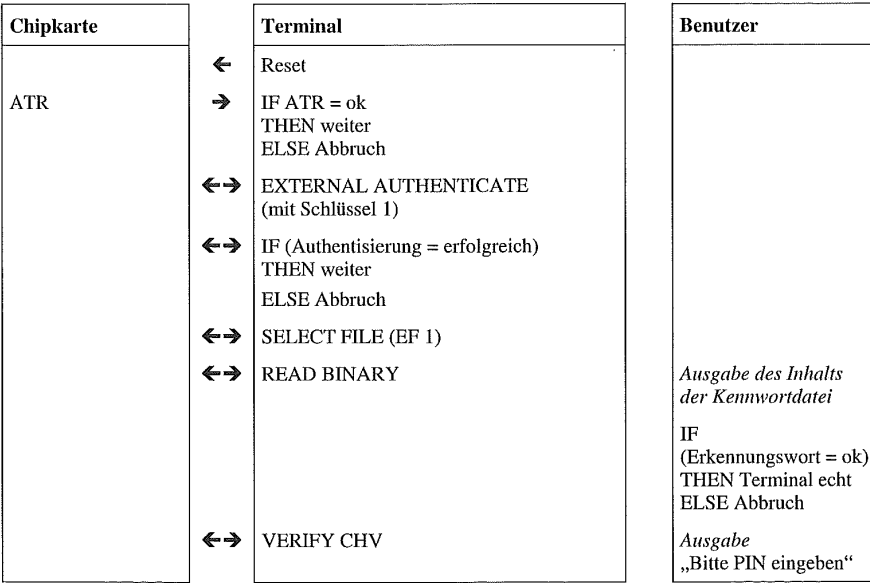


Bild 15.30 Ablauf der Kommandos bei der „Prüfung auf Echtheit eines Terminals“

16 Anhang

16.1	Glossar	905
16.2	Übersetzung von Fachwörtern	955
16.2.1	Übersetzungsliste Deutsch – Englisch	956
16.2.2	Übersetzungsliste Englisch – Deutsch	963
16.3	Weiterführende Literatur	970
16.4	Literatur	970
16.5	Kommentiertes Normen- und Spezifikationsverzeichnis	977
16.6	Kodierung von Datenobjekten	1002
16.6.1	Datenobjekte nach ISO/IEC 7816-4	1002
16.6.2	Datenobjekte nach ISO/IEC 7816-6	1003
16.6.3	Datenobjekt für Chiphersteller nach ISO/IEC 7816-6	1004
16.7	Registrierungsstellen für RID	1004
16.8	Ausgewählte RIDs	1005
16.9	Veranstaltungen	1005
16.10	World-Wide-Web-Adressen	1006
16.11	Kennwerte und Tabellen	1019
16.11.1	Zeitbereich für den ATR	1019
16.11.2	Umrechnungstabelle für Datenelemente des ATR	1019
16.11.3	Tabelle zur Ermittlung der Übertragungsgeschwindigkeit	1020
16.11.4	Tabelle mit Abtastzeitpunkten	1021
16.11.5	Tabelle der wichtigsten Chipkarten-Kommandos	1022
16.11.6	Übersicht über die verwendeten Instruction-Bytes	1027
16.11.7	Codierung von Chipkarten-Kommandos	1028
16.11.8	Chipkarten-Returncodes	1031
16.11.9	Ausgewählte Chips für Speicherkarten	1033
16.11.10	Ausgewählte Mikrocontroller für Chipkarten	1035

16.1 Glossar

Die folgenden Seiten enthalten eine Aufstellung von typischen Begriffen aus der Chipkartenvelt. Ausführliche und exakte Begriffsdefinitionen über Chipkarten finden sich zusätzlich in der ISO/IEC 7816 Normenreihe. Das Äquivalent dazu für den Sektor der elektronischen Geldbörsen ist die EN 1546, die ausführlich und prägnant alle dazugehörigen Fachbegriffe erläutert und definiert. Je nach üblichem Sprachgebrauch sind die Begriffe als Abkürzung bzw. in ausgeschriebener Form aufgeführt.

Der Pfeil → vor einem Begriff verweist auf einen weiteren Eintrag im Glossar, in dem das betreffende (*kursiv gesetzte*) Stichwort erläutert wird.

Größere Sammlungen von allgemeinen Begriffen aus der Informationstechnik finden sich in großer Zahl in der DIN 44300 und auch in den zahlreichen fachspezifischen EDV-Lexika, wie beispielsweise [Pfaffenberger 97, Dictionary of Computing 91].

μ P-Karte	andere Bezeichnung für $\rightarrow$ Mikroprozessorkarte
0-PIN	Eine für alle neu ausgegebenen $\rightarrow$ Chipkarten einheitliche und bekannte $\rightarrow$ PIN, mit der jedoch kein Zugriff auf die eigentlichen Benutzerfunktionen möglich ist. Sie ist damit eine $\rightarrow$ Trivial-PIN. Die 0-PIN muss bei der erstmaligen Benutzung der Karte in die vom Benutzer gewünschte PIN mit den üblichen Mechanismen (i. d. R. CHANGE CHV) abgeändert werden, wobei die 0-PIN als neue gültige PIN nicht erlaubt ist. Zweck der 0-PIN ist, dass der Benutzer beim Erhalt der Karte zweifelsfrei feststellen kann, ob sich diese noch in ihrem ursprünglichen Ausgabezustand befindet oder unberechtigt auf dem Zustellungsweg bereits benutzt wurde. Der Name 0-PIN stammt daher, weil oft der Wert "0000" für diese Art von PIN verwendet wird.
1 μ m, 0,8 μ m, ...-Technologie	Bei der Herstellung von Halbleiterchips wird traditionell die Leistung der verwendeten Technologie durch die Längenangabe für die kleinste mögliche Transistorstruktur auf dem Halbleiter beschrieben. Dies ist meist die Breite des Gateoxids bei Transistoren. Die momentan minimal möglichen Strukturbreiten liegen in der Größenordnung von 0,25 μ m und 0,13 μ m. Größere Strukturen auf dem Chip sind selbstverständlich immer möglich.
1, 2, 4, 8, ..., x k-Chip	Die Bezeichnung x k-Chip (mit x als ganzzahliger positiver Zahl) ist die oftmals im Sprachgebrauch vereinfachende Typenbezeichnung für $\rightarrow$ Mikrocontroller mit einer bestimmten EEPROM-Größe ($\rightarrow$ EEPROM) in kByte. Ein 32 k-Chip ist demnach ein Chipkarten-Mikrocontroller mit 32 kByte EEPROM. Für übersichtsmäßige Vergleiche von üblichen Chipkarten-Mikrocontrollern ist die Angabe der EEPROM-Größe durchaus ausreichend.
1G (erste Generation)	Bezeichnet die erste Generation von Mobilfunknetzen. Diese arbeiten analog und sind zellulär aufgebaut. Typische Beispiele für 1G-Systeme sind AMPS und das deutsche C-Netz.
2G (zweite Generation)	Bezeichnet die zweite Generation von Mobilfunknetzen. Diese arbeiten digital und sind zellulär aufgebaut. Typische Beispiele für 2G-Systeme sind $\rightarrow$ GSM und $\rightarrow$ CDMA.
3DES	$\rightarrow$ Triple-DES
3G (dritte Generation)	Bezeichnet die dritte Generation von Mobilfunknetzen. Diese arbeiten digital und sind zellulär aufgebaut. Das typische Beispiel für ein 3G-System ist $\rightarrow$ UMTS. UMTS gehört wiederum zur Familie von $\rightarrow$ IMT-2000.
3GPP (<i>third generation partnership project</i>) [3GPP]	Das <i>third generation partnership project</i> , gegründet von den fünf Normungsinstituten ANSI T1 (USA), ARIB (Japan), ETSI (Europa), TTA (Korea) und TTC (Japan), hat zur Aufgabe, weltweit verwendbare technische Spezifikationen für ein Mobilfunksystem der dritten Generation ($\rightarrow$ 3G) auf der Grundlage eines weiterentwickelten GSM-Kernsystems ($\rightarrow$ GSM) zu erstellen. Die betroffenen Normungsinstanzen setzen diese Spezifikationen anschließend in entsprechende Normen um. 3GPP wurde im Dezember 1998 in Kopenhagen von den wesentlichen internationalen Telekommunikations-Normungsinstanzen ins Leben gerufen. Ähnliche Aufgaben wie 3GPP hat 3GPP2 (<i>third generation partnership project 2</i>), hier ist jedoch der Fokus auf die Weiterentwicklung von Nicht-GSM-Systemen in Richtung dritter Generation wie beispielsweise CDMA-Systemen.
3GPP2	$\rightarrow$ 3GPP
4G (vierte Generation)	Bezeichnet die vierte Generation von Mobilfunknetzen. Diese stehen jedoch am Anfang der Konzeptionierung.
8, 16, 32 bit CPU (<i>central processing unit</i>)	Eine wichtige Kenngröße für die Leistungsfähigkeit eines $\rightarrow$ Mikroprozessors ist die Breite der Register für die zu verarbeitenden Daten im Rechenwerk. Sie wird in der Anzahl der Bits angegeben.

A2C (<i>administration to customer</i>)	Bezeichnet die Abwicklung von Geschäften des → <i>E-Commerce</i> zwischen öffentlicher Verwaltung und Endverbrauchern.
A3 (<i>algorithm 3</i>)	Bezeichnung für eine kryptografische Funktion bei → <i>GSM</i> für die Authentisierung des SIM durch das Hintergrundsystem mittels eines Challenge-Response-Verfahrens. A3 wird vom Netzbetreiber gewählt, ist also nicht einheitlich für das gesamte GSM-System.
A5 (<i>algorithm 5</i>)	Bezeichnung für eine kryptografische Funktion bei → <i>GSM</i> für die Verschlüsselung der Sprachdaten auf der Luftschnittstelle zwischen Mobile Station und Base Station bzw. Hintergrundsystem. A5 ist einheitlich für das gesamte GSM-System.
A8 (<i>algorithm 8</i>)	Bezeichnung für eine kryptografische Funktion bei → <i>GSM</i> zur Erzeugung des Sitzungsschlüssels Kc für die Sprachdatenverschlüsselung auf der Luftschnittstelle. A8 wird vom Netzbetreiber gewählt, ist also nicht einheitlich für das gesamte GSM-System.
Ablaufsteuerung (<i>Sequence Control</i>)	Legt verbindlich eine bestimmte Reihenfolge von Aktivitäten fest. Beispielsweise wird bei der gegenseitigen Authentisierung von → <i>Chipkarte</i> und Hintergrundsystem die korrekte Reihenfolge der → <i>Kommandos</i> durch eine Ablaufsteuerung in der Chipkarte realisiert. Dazu besitzt das → <i>Chipkarten-Betriebssystem</i> einen Zustandsautomaten (<i>state machine</i>), mit dem durch festlegbare Zustände (<i>states</i>) und Zustandsübergänge (<i>state transition</i>) bestimmte einzuhaltende Kommandosequenzen definierbar sind. ¹
Abschaltsequenz	Legt für einen → <i>Chipkarten-Mikrocontroller</i> die Deaktivierungsreihenfolge der elektrischen Signale beim Ausschalten einer → <i>Chipkarte</i> fest. Sie sagt nichts über die Reihenfolge der mechanischen Dekontaktierung aus. Der Zweck der Abschaltsequenz ist, die gegenüber Ladungen und Spannungen an den Kontakten empfindlichen Chipkarten-Mikrocontroller zu schützen (→ <i>Anschaltsequenz</i>).
Acquirer (Sammelbeauftragter)	Instanz, welche die Errichtung und Verwaltung der datentechnischen Verbindungen und des Datenaustauschs zwischen dem Betreiber eines Zahlungssystems und den einzelnen Service-Anbietern betreibt. Der Acquirer kann die erhaltenen einzelnen Transaktionen zusammenfassen, so dass der Betreiber des Zahlungssystems nur mehr gesammelte Zertifikate erhält.
AES (<i>advanced encryption standard</i>)	Symmetrischer → <i>Kryptoalgorithmus</i> . Er wurde ursprünglich von Joan Daemen und Vincent Rijmen entwickelt und als Rijndael publiziert. Das → <i>NIST</i> hat ihn nach einer Ausschreibung und einem Bewertungsverfahren im Jahr 2000 als Nachfolger des DES ausgewählt und 2001 als US-Norm (FIPS 197) publiziert. ²
AFNOR (<i>Association Française de Normalisation</i>)	Französische Normungsorganisation mit Sitz in Paris.
AID (<i>application identifier</i>)	Kennzeichen für eine → <i>Anwendung</i> auf einer → <i>Chipkarte</i> ; in der ISO/IEC 7816-5 definiert. Ein Teil des AID kann national oder international registriert werden und ist dann für die registrierte Anwendung reserviert und weltweit eindeutig. Der AID besteht aus den beiden Datenelementen RID (<i>registered identifier</i>) und PIX (<i>proprietary identifier</i>). ³
AMPS (<i>advanced mobile phone system</i>)	Zellularer Mobiltelefonstandard, der vor allem in USA, Lateinamerika, Australien und in Teilen von Asien verbreitet ist. AMPS arbeitet analog im 800-MHz-Bereich. AMPS-Endgeräte besitzen keine → <i>Chipkarte</i> und wurden u. a. auch aus diesem Grund oft erfolgreich angegriffen. Die Weiterentwicklung vom AMPS ist D-AMPS, ein digitaler Standard, der ebenfalls den 800-MHz-Bereich benutzt.

¹ Siehe auch Abschnitt 5.8 „Ablaufsteuerung“.

² Siehe auch Abschnitt 4.7.1 „Symmetrische Kryptoalgorithmen“.

³ Siehe auch Abschnitt 5.6.1 „Dateitypen“.

analog	Mit diesem Begriff bezeichnet man Systeme, bei denen ein Signal eine unbegrenzte Anzahl von Werten annehmen kann.
Analyse (<i>analysis</i>)	Im Sinne der Softwareentwicklung die Ermittlung der Anforderungen des Auftragsgebers an ein informationstechnisches System sowie deren eindeutige und vollständige Beschreibung. Vereinfacht ausgedrückt, ist das Ergebnis der Analyse eine Beschreibung, „was“ gemacht werden soll. Der nächste Schritt bei einer sequentiellen Softwareentwicklung ist das → <i>Design</i> .
Anonymisierung	Veränderung von personenbezogenen Daten in einer Weise, dass es nicht mehr möglich ist, diese veränderten Daten der ursprünglichen Person zuzuordnen (→ <i>Pseudonymisierung</i>).
Anschaltsequenz	Legt für einen → <i>Chipkarten-Mikrocontroller</i> die Aktivierungsreihenfolge der elektrischen Signale beim Einschalten einer → <i>Chipkarte</i> fest. Sie sagt nichts über die Reihenfolge der mechanischen Kontaktierung aus. Zweck der Anschaltsequenz ist, die gegenüber Ladungen und Spannungen an den Kontakten empfindlichen Chipkarten-Mikrocontroller zu schützen (→ <i>Abschaltsequenz</i>).
ANSI (<i>American National Standards Institute</i>) [ANSI]	US-amerikanische Normungsorganisation mit Sitz in New York.
Antikollisionsverfahren	Technisches Verfahren, das den störungsfreien Vielfachzugriff auf kontaktlose Karten ermöglicht.
Antwort (<i>response</i>)	→ <i>Kommando</i>
Antwort-APDU (<i>response-APDU</i>)	Antwort (→ <i>Kommando</i>) der → <i>Chipkarte</i> auf eine vom Terminal abgesendete → <i>Kommando-APDU</i> . Sie setzt sich aus den optionalen Antwortdaten und den je 1 Byte langen obligatorischen Statuswörtern SW1 und SW2 zusammen (→ <i>APDU</i>). ⁴
Anwendung (<i>application</i>)	Alle Daten, Dateien, → <i>Kommandos</i> , Abläufe, Zustände, Mechanismen, Algorithmen und Programme innerhalb einer → <i>Chipkarte</i> , um sie im Rahmen eines bestimmten Systems zu betreiben. Eine Anwendung mit den dazugehörigen Daten befindet sich in der Regel in einem eigenen DF direkt unterhalb des MFs. Dies wird oft auch als Oncard-Anwendung bezeichnet. Das Gegenstück ist die Offcard-Anwendung, die alle Programme und Daten einschließt, die sich nicht auf der Chipkarte befinden, jedoch für die Nutzung der Oncard-Anwendung auf der Chipkarte notwendig sind.
Anwendungsbetreiber	Eine Instanz, die eine → <i>Anwendung</i> auf → <i>Chipkarten</i> betreibt. Im Allgemeinen ist der Anwendungsbetreiber identisch mit dem Anwendungsanbieter.
APDU (<i>application protocol data unit</i>)	Softwaretechnischer Datencontainer, in den die Daten einer → <i>Anwendung</i> verpackt werden, um sie zwischen Terminal und → <i>Chipkarte</i> auszutauschen. Eine APDU wird vom Übertragungsprotokoll in eine TPDU (<i>transport protocol data unit</i>) umgewandelt und dann über die serielle Schnittstelle vom Terminal bzw. der Chipkarte verschickt. APDUs lassen sich in → <i>Kommando-APDU</i> und → <i>Antwort-APDU</i> einteilen. ⁵
API (<i>application programming interface</i>)	Detailliert spezifizierte Softwareschnittstelle für den Zugriff auf bestimmte Funktionen eines Programms.
Applet-Entwickler (<i>applet developer</i>)	Person oder Instanz, die ein → <i>Applet</i> entwickelt.

⁴ Siehe auch Abschnitt 6.5 „Struktur der Nachrichten – APDUs“.

⁵ Siehe auch Abschnitt 6.5 „Struktur der Nachrichten – APDUs“.

Applet	In der Programmiersprache Java erstelltes Programm, das von der virtuellen Maschine eines Rechners ausgeführt wird. Die Funktionalität eines Applets ist aus Sicherheitsgründen auf die vorher festgelegte Programmumgebung beschränkt. Ein Applet wird im Bereich von → <i>Chipkarten</i> manchmal auch Cardlet genannt und entspricht in der Regel einer Chipkarten-Anwendung (→ <i>Anwendung</i>).
ASK (<i>Amplitude Shift Keying</i> – Amplitudentastung)	Verfahren, bei dem die Amplitude der Trägerschwingung zwischen zwei Zuständen umgeschaltet wird.
ASN.1 (<i>abstract syntax notation one</i>)	Beschreibungssprache (Syntax und Grammatik) für Daten, mit der sich Daten und Datentypen unabhängig vom benutzenden Computersystem eindeutig definieren und darstellen lassen. Mit den → <i>BER</i> (<i>basic encoding rules</i>) und den → <i>DER</i> (<i>distinguished encoding rules</i>) werden die entsprechenden Daten dann konkret codiert. ASN.1 ist durch ISO/IEC 8824 und ISO/IEC 8825 definiert. ⁶
Assembler	Programm, das Assemblerprogramme in von einem Prozessor ausführbare Maschinensprache übersetzt. Nach dem Assembliervorgang muss das Programm üblicherweise noch mit einem Linker gebunden (<i>linking</i>) werden. Der Begriff Assembler wird jedoch oft auch als Kurzform von Assemblerprogrammcode benutzt.
asymmetrischer Kryptoalgorithmus	→ <i>Kryptoalgorithmus</i>
asynchrone Datenübertragung	Hier werden die Daten unabhängig von einem vorgegebenen Zeitraster übertragen (→ <i>synchrone Datenübertragung</i>).
atomarer Ablauf (<i>atomic operation</i>)	Eine oder mehrere Operationen in einem Programm, die entweder vollständig oder überhaupt nicht ausgeführt werden. Bei → <i>Chipkarten</i> werden atomare Abläufe oft im Zusammenhang mit EEPROM-Schreibroutinen eingesetzt. Damit wird sichergestellt, dass Dateninhalte zu jedem Zeitpunkt konsistent sind. ⁷
ATR (<i>answer to reset</i>)	Sequenz von Bytes, welche eine → <i>Chipkarte</i> als Antwort auf den (Hardware-) Reset aussendet. Der ATR beinhaltet u.a. diverse Parameter für das Übertragungsprotokoll zur Chipkarte. ⁸
Attribut (<i>attribute</i>)	Im Sinne der → <i>objektorientierten</i> Programmierung die Datencontainer (im prozeduralen Sinne die Variablen), die ein → <i>Objekt</i> enthält. Die Attributwerte können durch → <i>Methoden</i> gelesen oder geändert werden.
Auswurfleser	Terminal, das eine gesteckte Karte durch ein elektrisches oder mechanisches Signal automatisch auswerfen kann.
Authentifizierung	anderer Ausdruck für → <i>Authentisierung</i>
Authentisierung	Vorgang des Nachweises der Echtheit einer Instanz (z. B. einer Chipkarte) durch kryptografische Verfahren. Vereinfacht ausgedrückt, stellt man bei der Authentisierung durch ein festgelegtes Verfahren fest, ob jemand wirklich derjenige ist, der er vorgibt zu sein.
Authentizität	Echtheit und Unverändertheit einer Instanz oder Nachricht.
Automat	In der Informationstechnik Teil eines Programms, das einen Ablauf auf der Grundlage eines vorher definierten Zustandsdiagramms (d. h. Zustände mit Zustandsübergängen) bestimmt.

⁶ Siehe auch Abschnitt 4.1 „Strukturierung von Daten“.

⁷ Siehe auch Abschnitt 5.10 „Atomare Abläufe“.

⁸ Siehe auch Abschnitt 6.2 „Answer to Reset – ATR“.

Autorisierung	Prüfung, ob eine bestimmte Aktion ausgeführt werden darf, d. h. jemand wird zu etwas ermächtigt. Wird beispielsweise eine Kreditkartentransaktion durch den Kreditkarten-Herausgeber autorisiert, so sind die Kartendaten auf Korrektheit der Daten, Einhaltung der erlaubten Betragsgrenzen und ähnliche Kriterien geprüft worden und die beabsichtigte Zahlung wird daraufhin zugelassen. Eine Autorisierung kann durch die Authentisierung der betreffenden Instanz (z. B. einer Chipkarte) zustande kommen. Vereinfacht ausgedrückt, erteilt man bei der Autorisierung jemandem die Erlaubnis, etwas Bestimmtes zu tun.
B2A (<i>business to administration</i>)	Bezeichnet die Abwicklung von Geschäften des → <i>E-Commerce</i> zwischen Unternehmen und öffentlicher Verwaltung.
B2B (<i>business to business</i>)	Bezeichnet die Abwicklung von Geschäften des → <i>E-Commerce</i> zwischen Unternehmen.
B2C (<i>business to customer</i>)	Bezeichnet die Abwicklung von Geschäften des → <i>E-Commerce</i> zwischen Unternehmen und Endverbrauchern.
Bad Day Szenario	anderer Ausdruck für → <i>Schlechtfall</i>
baud	Benennt während einer Datenübertragung die Anzahl von Statusänderungen pro Sekunde. Je Statusänderung können jedoch abhängig vom Übertragungsverfahren ein oder mehrere Informationsbits übertragen werden. Aus diesem Grund kann Baud nur in dem Sonderfall, dass bei jeder Statusänderung lediglich ein Bit übertragen wird, mit der Übertragungsrateneinheit bit/s gleichgesetzt werden.
Bearer (Träger)	Bezeichnet den Trägerdienst, mit dessen Hilfe Daten an ein Endgerät übertragen werden. Ein möglicher Bearer für WAP ist beispielsweise SMS.
Bellcore-Angriff (<i>Bellcore-Attack</i>)	→ <i>differentielle Fehleranalyse</i>
Benutzer (<i>user</i>)	Person, die eine → <i>Chipkarte</i> verwendet. Sie muss nicht unbedingt der → <i>Karteninhaber</i> sein.
BER (<i>basic encoding rules</i>)	Die in → <i>ASN.1</i> festgelegten grundlegenden Codierungsregeln BER (<i>basic encoding rules</i>) dienen dazu, Daten als Datenobjekte zu codieren. BER-codierte Datenobjekte besitzen ein Kennzeichen (<i>tag</i>), eine Längenangabe (<i>length</i>), den eigentlichen Datenteil (<i>value</i>) und optional eine Endekennung. Aus diesem Grund werden sie auch als TLV-codierte Daten bezeichnet. Das BER-Format lässt auch verschachtelte Datenobjekte zu. Eine Untermenge der grundlegenden Codierungsregeln BER sind die DER (<i>distinguished encoding rules</i>). Diese geben unter anderem an, wie die Längenangabe der jeweiligen Datenobjekte zu codieren ist (1, 2 oder 3 Byte lang). ⁹
Betriebssystem (<i>OS – operating system</i>)	Umfasst alle Programme eines digitalen Rechnersystems, die zusammen mit den Eigenschaften der Rechneranlage die Grundlage der möglichen Betriebsarten bilden und insbesondere die Abwicklung von Programmen steuern und überwachen. ¹⁰
Betriebssystemhersteller (<i>operating system manufacturer</i>)	Instanz, die die Programmierung und den Test eines → <i>Betriebssystems</i> durchführt.
big Endian	→ <i>Endianmess</i>
binärkompatibler Programmcode	Programm, das direkt vom → <i>Mikroprozessor</i> ohne Übersetzung o. Ä. ausgeführt werden kann (→ <i>Programmcode</i>).

⁹ Siehe auch Abschnitt 4.1 „Strukturierung von Daten“.

¹⁰ Text in Anlehnung an die deutsche Norm DIN 44300.

Blacklist	anderer Ausdruck für → <i>Sperrliste</i>
Blackbox-Test	Bei diesem Test geht man davon aus, dass die testende Instanz keine Kenntnisse von den internen Abläufen, Funktionen und Mechanismen der zu prüfenden Software hat.
Bluetooth [Bluetooth]	Eine für den Nahbereich (<100 m) konzipierte drahtlose Netzwerktechnologie im 2,4 GHz Frequenzbereich mit einer maximalen Bruttoübertragungsrate in der Größenordnung von 1 MBit/s. Die Firma Ericsson als Initiator dieser Technologie wollte mit der Namensgebung an den vor ca. 1000 Jahren lebenden dänischen König Harald II. erinnern, der den Beinamen Bluetooth trug. Sein Verdienst war die Zusammenfassung vieler einzelner Gebiete zu einem einheitlichen Königreich.
Bond-out-Chip	Ein in einem vielpoligen Keramikgehäuse untergebrachter Mikrocontroller, bei dem alle chipinternen Busse für Speicher frei zugänglich sind, sodass das üblicherweise maskenprogrammierte → <i>ROM</i> durch einen chipexternen Speicher ersetzt werden kann. Zweck eines Bond-out-Chips ist die Schaffung der Möglichkeit, ohne → <i>ROM-Maske</i> Tests von Software auf einer Zielhardware durchzuführen.
Bootloader (Urlader)	Einfaches kleines Programm, dessen einzige Funktion darin besteht, weitere und umfangreichere Programme beispielsweise über eine serielle Schnittstelle in einen Speicher nachzuladen und dort zu starten (→ <i>Loader</i>). Der Bootloader wird in der Regel dazu benutzt, den eigentlichen Programmcode in einen neuen Chip oder in ein neues elektronisches Gerät zu laden. Oft kann der Bootload-Vorgang nur ein einziges Mal ausgeführt werden.
Börsenanbieter (<i>purse provider</i>)	Die Organisation, die für die Gesamtfunktionalität und Sicherheit eines Börsensystems verantwortlich ist. Er ist im Regelfall auch der Herausgeber des elektronischen Kartengeldes und garantiert für die Einlösung.
Börseninhaber (<i>purse holder</i>)	Die Person, die die → <i>Chipkarte</i> mit der elektronischen Geldbörse besitzt.
BPSK (<i>Binary Phase Shift Keying</i>)	Phasenumtastung von 180°, wodurch sich zwei Phasenzustände ergeben.
Browser	Programm zur Betrachtung von Hypertextdokumenten, zur Navigation zwischen diesen Dokumenten und zur Ausführung von in die Hypertextdokumente eingebettetem → <i>Programmcode</i> . Einfach aufgebaute, wenig Speicher und Rechenleistung benötigende Browser werden oft auch als Microbrowser bezeichnet. Diese können beispielsweise als → <i>Anwendung</i> innerhalb eines → <i>Chipkarten-Betriebssystems</i> laufen (z. B.: SIM Alliance-Browser, auch S@T-Browser genannt) oder in die Software eines Mobiltelefons integriert sein (z. B. WAP-Browser). Browser können durch nachträglich ladbare Softwarekomponenten, so genannte Browser-Plug-Ins, in ihrem Funktionsumfang erweitert werden.
Brute-force-Angriff (<i>brute force attack</i>)	Angriff auf ein kryptografisches System durch die Berechnung aller Möglichkeiten eines Schlüssels.
BSI (Bundesamt für Sicherheit in der Informationstechnik) [BSI]	Das deutsche Bundesamt für Sicherheit in der Informationstechnik (BSI) wurde 1991 gegründet und ist der Nachfolger der deutschen Zentralstelle für das Chiffrierwesen. Das BSI hat folgende Aufgaben: Untersuchung von Sicherheitsrisiken bei IT-Anwendungen, Prüfung und Bewertung der Sicherheit von IT-Systemen, Zulassung von IT-Systemen für Behörden und Unterstützung der Strafverfolgungs- und Verfassungsschutzbehörden. Außerdem bietet das BSI Beratung von Herstellern, Vertreibern und Anwendern in Fragen der IT-Sicherheit an und legt insofern oft auch die Rahmenbedingungen für Kryptografieanwendungen in Deutschland fest.

buffern	Typischer Angriff bei Magnetstreifenkarten. Dazu wird der Magnetstreifen zunächst gelesen und gespeichert. Nachdem ein Terminal Daten (z. B. den Fehlbedienungsanzähler) auf dem Magnetstreifen verändert hat, werden die ursprünglichen Daten wieder auf den Magnetstreifen zurückgeschrieben.
bug fix	In der Softwareentwicklung die Ausbesserung eines bekannten Fehlers (<i>bug</i>) durch zusätzlichen → <i>Programmcode</i> . Der Fehler als solcher wird durch einen bug fix beseitigt, im Gegensatz zum → <i>work around</i> .
Burst	→ <i>Signalburst</i>
Bytecode (<i>bytecode</i>)	Dieser Begriff ist mehrfach besetzt und kann nur in seinem verwendeten Kontext korrekt interpretiert werden. Eine weit verbreitete Verwendung des Begriffs steht im Zusammenhang mit Java. Unter Bytecode wird dabei der von einem Java-Compiler aus dem Source-Code erzeugte (d.h. compilierte) Zwischencode für die Java Virtual Machine (JVM) bezeichnet. Der Bytecode ist von der Firma Sun standardisiert und wird vom Interpreter der Java Virtual Machine ausgeführt. Eine weitere Verwendung des Begriffs Bytecode gibt es im Umfeld von Mikrobrowsern (→ Browser). Unter „Bytecode“ wird hier das von einem so genannten Bytecode-Converter übersetzte Hypertextdokument verstanden. Das Ergebnis der Übersetzung ist der Bytecode, der dann im nächsten Schritt vom Mikrobrowser interpretiert wird.
CAD (<i>card acceptance device</i>)	Im Bereich des elektronischen Zahlungsverkehrs wird statt der ISO-Abkürzung → <i>IFD (interface device)</i> oft die Bezeichnung CAD für Chipkarten-Terminal benutzt.
CAMEL (<i>customized applications for mobile enhanced logic</i>)	Zusätzlich mögliches Charakteristikum von → <i>GSM</i> , um die Funktionalität von so genannten intelligenten Netzen (IN – <i>Intelligent Network</i>) zu unterstützen. Mit CAMEL können beispielsweise Rufnummern während des Rufaufbaus vom Netzwerk geändert werden. Damit lassen sich dann auf einfache Weise Anwendungen wie weltweites → <i>Roaming</i> mit vorbezahlten Karten oder weltweit verfügbare einheitliche Servicenummern realisieren.
CAP-Datei (<i>card application file – CAP-file</i>)	Datenaustauschformat zwischen der Java Offcard Virtual Machine und der Java Oncard Virtual Machine.
Card Modelling Language (CML)	Abstrakte und betriebssystemunabhängige Beschreibungssprache für die Definition von Chipkarten-Anwendungen (→ <i>Anwendung</i>).
Cardholder Verification Method (CVM)	Verfahren zur → <i>Identifizierung</i> von Personen. Üblicherweise ist dies die PIN-Prüfung, es kann jedoch auch bei höher entwickelten Systemen eine biometrische Benutzeridentifizierung sein.
Cardlet	→ <i>Applet</i>
CCITT (<i>Comité Consultatif International Télégraphique et Téléphonique</i>)	Einst ein internationaler Ausschuss für Telefon- und Telegraphendienste mit Sitz in Genf. Mittlerweile ist die CCITT unter Erweiterung der Aufgaben in → <i>ITU</i> umgetauft worden.
CCS (<i>cryptographic checksum</i>)	Kryptografische Prüfsumme über Daten, mit der Manipulationen dieser Daten während der Speicherung erkannt werden können. Werden Daten während ihrer Übertragung mit einer CCS geschützt, so spricht man von einem → <i>MAC (message authentication code)</i> .
CDMA (<i>code division multiple access – Codevielfachzugriff</i>)	Vielfachzugriffsverfahren zur parallelen Datenübertragung von mehreren Sendern zu einem Empfänger innerhalb eines Frequenzspektrums. Dazu wird im Sender das schmalbandige Funksignal durch eine für diesen Sender spezifische Abbildungsvorschrift auf ein breitbandiges Funksignal abgebildet, d. h. ge-

spreizt. Bei bekannter senderindividueller Abbildungsvorschrift kann dann im Empfänger aus dem empfangenen breitbandigen Funksignal das schmalbandige Funksignal wiederhergestellt werden. CDMA wird bei UMTS auf der Luft-schnittstelle zwischen Mobiltelefon und Basisstation verwendet.¹¹ Bei WCDMA (*wideband code division multiple access*) findet → *Uplink* und → *Downlink* in zwei getrennten Frequenzspektren statt, weshalb dieses CDMA-Verfahren oft auch als FD/CDMA (*frequency division/code division multiple access*) bezeichnet wird. Bei TD/CDMA (*time division/code division multiple access*) wird die Trennung zwischen Uplink und Downlink durch die Benutzung unterschiedlicher Zeitschlitzte realisiert.

CEN (<i>Comité Européen de Normalisation</i>) [CEN]	Die europäische Normungsorganisation CEN in Brüssel (Belgien) setzt sich aus den nationalen Normungsorganisationen aller europäischen Länder zusammen und ist die offizielle Institution der EU für europäische Normung.
CEPS (<i>common electronic purse specifications</i>) [CEPSCO]	Eine Spezifikation für → <i>elektronische Geldbörsen</i> mit dem Fokus auf weltweite Interoperabilität (→ <i>interoperabel</i>) und Einbeziehung aller Komponenten, die für den Betrieb eines elektronischen Geldbörsensystems notwendig sind. CEPS wurde 1999 in ihrer ersten Version von CEPSCO veröffentlicht und baut auf vielen Prinzipien der europäischen Norm für elektronische Geldbörsen EN 1546 auf.
CEPT (<i>Conférence Européenne des Postes et Télécommunications</i>)	Europäische Normungsorganisation der nationalen Telekommunikationsgesellschaften.
Challenge-Response-Verfahren	Das im Chipkartenbereich übliche Authentisierungsverfahren. Ein geheimer Schlüssel eines Kryptoalgorithmus ist das gemeinsame Geheimnis der beiden Kommunikationspartner. Ein Kommunikationspartner sendet dem anderen eine Zufallszahl (<i>challenge</i>), dieser verschlüsselt diese mit einem Kryptoalgorithmus und sendet das Ergebnis an den Fragesteller zurück (<i>response</i>). Der Fragesteller wendet nun auf die erhaltene verschlüsselte Zufallszahl die Umkehrfunktion des Kryptoalgorithmus an und vergleicht das Ergebnis dieser Operation mit der ursprünglich gesendeten Zufallszahl. Stimmen beide überein, so weiß der Fragesteller, dass der andere Kommunikationsteilnehmer ebenfalls den geheimen Schlüssel kennt und schließt daraus, dass dieser authentisch ist. ¹²
Chinesischer Restklassensatz (<i>chinese remainder theorem</i>)	Verfahren, um den → <i>RSA</i> -Algorithmus zu beschleunigen. Dazu müssen jedoch die beiden geheimen Primzahlen p und q bekannt sein, weshalb das Verfahren nur zur Entschlüsselung bzw. zum Signieren benutzt werden kann.
Chipgröße (<i>chip size</i>)	Fläche eines Chips, üblicherweise in Quadratmillimeter gemessen. Sie ist weitgehend direkt proportional zum Chip-Preis. Die maximale Chipgröße bei Chipkarten-Mikrocontrollern liegt aufgrund der heute üblichen Module bei ca. 25 mm ² .
Chipkarte (<i>chip card</i>)	Der allgemeine Begriff für eine Karte, meist aus Kunststoff, die ein oder mehrere Halbleiterchips enthält. Eine Chipkarte kann entweder eine → <i>Speicherkarte</i> oder eine → <i>Mikroprozessorkarte</i> sein.
Chipkarten-Anwendung (<i>smart card application</i>)	→ <i>Anwendung</i>
Chipkarten-Betriebssystem (<i>smart card operating system</i>)	(Oft auch COS (<i>card operating system</i>) genannt) Eine auf die Belange von Chipkarten spezialisierte Form eines → <i>Betriebssystems</i> ; umschließt alle Programme auf einem → <i>Chipkarten-Mikrocontroller</i> , die den Betrieb und die Ver-

¹¹ Siehe auch Abschnitt 13.1.1 „Vielfachzugriffsverfahren“.

¹² Siehe auch Abschnitt 4.11.2 „Gegenseitige symmetrische Authentisierung“.

	Verwaltung von Chipkarten-Anwendungen (→ <i>Anwendung</i>) ermöglichen. Dazu müssen die für eine oder mehrere Chipkarten-Anwendungen benötigten Daten, Dateien, → <i>Kommandos</i> , Abläufe, Zustände, Mechanismen, Algorithmen und Programme in geeigneter Weise unterstützt werden. Bietet ein Chipkarten-Betriebssystem die Möglichkeit des Betriebs mehrerer parallel existierender Anwendungen, so spricht man von multiapplikationsfähig. Die Entwicklungstendenz von Chipkarten-Betriebssystemen geht in Richtung → <i>offener Chipkarten-Betriebssysteme</i> . Typische Chipkarten-Betriebssysteme sind → <i>Multos</i> , → <i>Java Card</i> , → <i>Windows for Smart Cards</i> und → <i>STARCOS</i> .
Chipkarten-Mikrocontroller (<i>smart card microcontroller</i>)	Speziell für die Anforderungen von Chipkarten optimierte → <i>Mikrocontroller</i> . Die Verbesserungen betreffen vor allem Aspekte der Chipsicherheit (z. B. Schutzschichten, Detektoren), Chipgröße und besondere Funktionseinheiten für chipkartenspezifische Anforderungen (z. B. UART für Kommunikation).
Chipmodul (<i>chip modul</i>)	Der Träger und die Halterung für ein Die mit darauf angeordneten Kontaktelementen. Die häufig gebrauchte Kurzform für Chipmodul ist Modul.
Chip-on-Tape (COT)	Beim Chip-on-Tape sind die → <i>Chipmodule</i> auf ein flexibles dünnes Band mit typischerweise 35 mm Breite paarig nebeneinander gesetzt.
CHV	→ <i>PIN</i>
CICC (<i>contactless integrated chip card</i>)	Nach ISO die normgerechte Bezeichnung für eine Karte mit Chip, bei der die Energie- und Datenübertragung berührungslos durch elektromagnetische Felder erfolgt. Es kann sich dabei sowohl um einen Speicherchip als auch einen Mikrocontrollerchip handeln.
Class-Datei (<i>classfile</i>)	Die kompilierten, d. h. in Bytecode übersetzten und mit zusätzlichen Informationen versehenen Java-Programme werden in einer so genannten Class-Datei abgespeichert. Nach dem Laden werden sie von der Java Virtual Machine ausgeführt.
Clearing	Funktion der Abrechnung im elektronischen Zahlungsverkehr zwischen dem Akzeptanten einer elektronischen Zahlung (in der Regel ein Händler) und seiner Bank.
Clearingsystem	Rechnergestütztes Hintergrundsystem, das die zentrale Abrechnung im Rahmen einer Anwendung für elektronischen Zahlungsverkehr übernimmt.
Clean Room VM	→ <i>Java Card Virtual Machine</i>
CLIP	Markenname von Europay für technologisch unterschiedliche elektronische Geldbörsensysteme mit Chipkarten.
CMDA 2000 (<i>code division multiple access 2000</i>)	Mobilfunksystem der dritten Generation (→ <i>3G</i>) im 2000 MHz-Frequenzbereich mit ähnlichen Leistungsmerkmalen wie → <i>UMTS</i> . Die vorgesehene Chipkarte für CDMA 2000 Endgeräte ist optional und hat den Namen → <i>R-UTM</i> .
CMM (<i>capability maturity model</i>)	International verwendetes Modell zur Bestimmung des Reifegrades einer Softwareentwicklung. Der Reifegrad wird mit einem standardisierten Fragenkatalog bestimmt und hat 5 Stufen. Reifegrad 1 kennzeichnet eine mehr oder minder nach chaotischen Prinzipien ablaufende Entwicklung, Reifegrad 5 kennzeichnet eine geordnete und sich kontinuierlich selbst verbessernde Entwicklung als höchsten möglichen Reifegrad. ¹³
CODEC (<i>compressor/decompressor</i> oder <i>coder/decoder</i>)	Hardwarechip oder Algorithmus, der entweder für die Kompression/Dekompression oder die Verschlüsselung/Entschlüsselung von Daten bestimmt ist.
Combikarte	Eingetragenes Warenzeichen der Firma ADE, das eine → <i>Dual-Interface-Karte</i> bezeichnet.

¹³ Siehe auch Abschnitt 15.7 „Vorgehensmodelle“.

Common Criteria (CC) [CC]	Kriterienkatalog zur Entwicklung und → <i>Evaluierung</i> von informationstechnischen Systemen und soll zukünftig nationale und internationale Kriterienkataloge wie → <i>TCSEC</i> und → <i>ITSEC</i> ersetzen. Die Common Criteria wurden 1996 in der Version 1.0 von → <i>NIST</i> veröffentlicht und sind mittlerweile auch als ISO 15408 international genormt. Aktuell gültig ist die Version 2.0 von 1998.
Compiler	Programm, das eine Programmiersprache wie BASIC oder C in von einem Mikroprozessor direkt ausführbare Maschinensprache übersetzt. Nach dem Kompilationsvorgang muss das Programm üblicherweise noch mit einem Linker gebunden (linking) werden.
COS (<i>card operating system</i>)	Übliche Bezeichnung für → <i>Chipkarten-Betriebssysteme</i> . Sie ist auch oft ein Teil des Produktnamens des Betriebssystems (z. B. STARCOS).
CP8	Markenname eines → <i>Multiapplication-Chipkarten-Betriebssystems</i> von Bull [Bull], das in verschiedenen Versionen erhältlich ist.
CPU (<i>central processing unit</i>)	→ <i>Mikroprozessor</i>
CRC (<i>cyclic redundancy check</i> – zyklische Redundanzprüfung)	Einfacher und weit verbreiteter Fehlererkennungscode (→ <i>EDC</i>) zur Sicherung von Daten. Vor seiner Anwendung muss der CRC durch einen Startwert und ein Teilerpolynom festgelegt werden.
CT-API (<i>chipcard terminal – application programming interface</i>)	Eine in Deutschland weit verbreitete anwendungsunabhängige Schnittstellen-Spezifikation für die Anbindung von → <i>MKT-Terminals</i> an PCs. Der Herausgeber der Spezifikation ist Teletrust Deutschland.
D-AMPS	→ <i>AMPS</i>
Dateibody (<i>file body</i>)	→ <i>Dateiheader</i>
Dateiheader (<i>file header</i>)	Dateien in Chipkarten sind üblicherweise immer in zwei getrennte Teile aufgespalten. Der Dateiheader genannte Teil enthält Informationen über die → <i>Dateistruktur</i> und die → <i>Zugriffsbedingungen</i> , während in dem mit einem Zeiger verbundenen Dateibody die veränderbaren Nutzdaten gespeichert sind.
Dateistruktur (<i>file structure</i>)	Der von außerhalb der Chipkarte sichtbare Aufbau eines → <i>EFs</i> . Dateistrukturen dienen der logisch strukturierten und speicherplatzminimierten Ablage von Nutzdaten. Die üblichen Dateistrukturen nach ISO/IEC 7816-4 sind transparent, linear fixed, linear variable und cyclic. ¹⁴
Dateityp (<i>file type</i>)	Kennzeichnet bei Dateiverwaltungen in Chipkarten die Art einer Datei, d. h. ob es sich um eine Verzeichnisdatei (MF, DF) handelt oder um eine Datei für Nutzdaten (EF).
DEA (<i>data encryption algorithm</i>)	anderer Name für → <i>DES</i>
Debitkarte (<i>debit card</i>)	Karte mit oder ohne Chip, die einen Verfügungsrahmen aufweist und bei der die Bezahlung zeitgleich nach Erhalt des Gutes oder der Dienstleistung stattfindet. Dazu ist die Debitkarte mit einem Bankkonto verbunden, so dass unmittelbar bei der Bezahlung der entsprechende Betrag transferiert werden kann. Diese Bezahlung wird oft als „pay now“ bezeichnet. Das typische Beispiel für Debitkarten sind ec-Karten.
Debugging	Fehlersuche und -beseitigung mit dem Zweck, möglichst viele Fehler einer Software zu erkennen und zu korrigieren. Das Debugging wird in der Regel vom

¹⁴ Siehe auch Abschnitt 5.6.4 „Dateistrukturen von Efs“.

	Softwareentwickler im Zuge der → <i>Implementierung</i> durchgeführt und ist nicht identisch mit dem Testing (→ <i>Test</i>).
DECT (<i>digital enhanced cordless telecommunications</i> , früher: <i>digital European cordless telecommunications</i>)	Ein von → <i>ETSI</i> herausgegebener Standard für schnurlose Telefone in → <i>Zellulartechnik</i> mit digitaler Datenübertragung im 1,9 GHz-Bereich. Chipkarten sind bei DECT im Mobilteil des Telefons vorgesehen, jedoch nach Spezifikation optional, was dazu führte, dass sie nicht verwendet werden.
Defragmentierung (<i>defragmentation</i>)	Eine Defragmentierung hat die Aufgabe, an unterschiedlicher physikalischer Position im Speicher abgelegte Daten so zu verschieben, dass diese Daten zusammenhängend abgelegt sind. Die wesentlichen Teile einer Defragmentierung müssen atomar ablaufen, da es sonst beim Abbruch des Vorgangs zu Speicherinkonsistenzen kommen kann.
Delamination (<i>delamination</i>)	Unerwünschtes Auseinanderlösen von durch Druck und Hitze miteinander verbundenen (d. h. laminierten) Folien. Die Delamination einer Karte kann beispielsweise durch zu großflächige Aufdrucke mit nicht thermoplastischen Druckfarben (typische Farben für Offsetdruck) zwischen Kern- und Deckfolien verursacht werden.
Depersonalisierung (<i>depersonalisation</i>)	Rückgängigmachung der elektrischen → <i>Personalisierung</i> einer Chipkarte. Falls ein → <i>Chipkarten-Betriebssystem</i> eine Depersonalisierung zulässt, kann diese beispielsweise durch ein spezielles → <i>Kommando</i> nach vorheriger Authentisierung ausgeführt werden. Depersonalisierungen werden unter anderem bei falsch personalisierten Chipkarten ausgeführt, um diese wiederverwerten zu können.
DER (<i>distinguished encoding rules</i>)	→ <i>BER</i>
DES (<i>data encryption standard</i>)	Der bekannteste und verbreitetste symmetrische kryptografische Algorithmus (→ <i>Kryptoalgorithmus</i>). Er wurde von IBM zusammen mit dem → <i>NBS</i> entwickelt und 1977 als US-Norm (FIPS 46) als DEA (<i>data encryption algorithm</i>) publiziert. ¹⁵ Offizieller Nachfolger des DES ist der → <i>AES</i> .
Design (<i>design</i>)	Das Design, oft auch Entwurf genannt, im Sinne der Softwareentwicklung ist der Aufbau einer Softwarearchitektur anhand der bei der → <i>Analyse</i> ermittelten Anforderungen. Vereinfacht ausgedrückt, ist das Ergebnis des Designs eine Beschreibung, „wie“ die Anforderungen in der Software umgesetzt werden. Der nächste Schritt bei einer sequentiellen Softwareentwicklung ist die → <i>Implementierung</i> .
deterministisch	Bezeichnet ein Verfahren oder einen Algorithmus, das bzw. der bei identischen Ausgangsbedingungen immer zum gleichen Ergebnis kommt. Das Gegenteil ist → <i>probabilistisch</i> .
DF (<i>dedicated file</i>)	Verzeichnis im Dateisystem einer Chipkarte. Ein Sonderfall eines DFs ist das Root-Verzeichnis MF.
DF Name	Neben dem FID (<i>file identifier</i>) ein weiteres Merkmal für ein DF, mit einer Länge zwischen 1 und 16 Byte. Er wird zur Selektion des DFs benutzt und kann einen registrierten 5 bis 16 Byte langen AID (<i>application identifier</i>) enthalten, der das DF weltweit eindeutig macht. ¹⁶
Die, Dice	Siliziumkristall in Form eines Plättchens, auf dem sich ein einzelner halbleitertechnisch aufgebauter elektronischer Schaltkreis (z. B. Mikrocontroller) befindet.

¹⁵ Siehe auch Abschnitt 4.7.1 „Symmetrische Kryptoalgorithmen“.

¹⁶ Siehe auch Abschnitt 16.7 „Registrierungsstellen für RID“.

Diensteanbieter (<i>service provider</i>)	Instanz, welche gegen Bezahlung einen Dienst (z. B. Verkauf von Gütern oder Dienstleistungen) anbietet.
differentielle Fehleranalyse (DFA) (<i>differential fault analysis</i>)	Das Prinzip der differentiellen Fehleranalyse wurde 1996 von Dan Boneh, Richard A. DeMillo und Richard J. Lipton, die alle drei bei Bellcore angestellt waren, veröffentlicht [Boneh 96]. Das Verfahren basiert darauf, durch die bewusste Einstreuung von Fehlern während der kryptografischen Berechnung den geheimen Schlüssel zu ermitteln. Im ursprünglichem Verfahren wurden nur Public-Key-Algorithmen genannt. Diese Angriffsmethode wurde jedoch innerhalb einiger Monate sehr schnell weiterentwickelt [Anderson 96a], sodass mit der differentiellen Fehleranalyse prinzipiell alle Kryptoalgorithmen angegriffen werden können, sofern sie keine besonderen Schutzmaßnahmen aufweisen.
differentielle Kryptoanalyse (<i>differential crypto analysis</i>)	Hier werden Klartext-Paare mit bestimmten Unterschieden und gleichem Schlüssel benutzt, um durch die Analyse der Entwicklung dieser Unterschiede über die einzelnen DES-Runden hinweg den geheimen Schlüssel zu errechnen. Diese Analyse-methode wurde von Eli Biham und Adi Shamir 1990 veröffentlicht.
digital	Bezeichnet Systeme, bei denen ein Signal nur eine begrenzte Anzahl von Werten annehmen kann.
digitale Signatur (<i>digital signature</i>)	Digitale Signaturen werden zur Feststellung der Authentizität von elektronischen Nachrichten oder Dokumenten verwendet. Digitale Signaturen beruhen in der Regel auf asymmetrischen Kryptoalgorithmen, wie beispielsweise dem RSA-Algorithmus. Die Rechtswirksamkeit einer digitalen Signatur wird in vielen Ländern durch Gesetz geregelt. In Deutschland beispielsweise durch das → <i>Signaturgesetz</i> . Digitale Signaturen werden manchmal auch elektronische Unterschriften genannt.
digitaler Fingerabdruck (<i>digital fingerprint</i>)	Häufige Bezeichnung für den Hash-Wert einer Nachricht (z. B. mit SHA-1 erstellt).
digitales Wasserzeichen (<i>digital watermark</i>)	Eine idealerweise unsichtbare bzw. unhörbare, nicht rückgängig zu machende Markierung in Bild- oder Tondateien zur Realisierung von Urheberschutz. Bei Bedarf können dann mit Analyseprogrammen entsprechende Bild- oder Tondateien auf ein digitales Wasserzeichen überprüft werden. Zur Erzeugung von digitalen Wasserzeichen werden oftmals Verfahren der → <i>Steganografie</i> benutzt.
Downlink	Verbindung von einem übergeordneten System (z. B. Basisstation) zu einem untergeordneten System (z. B. Mobiltelefon). Das Gegenteil ist der → <i>Uplink</i> .
Download	Übertragen von Daten von einem übergeordneten System (z. B. Hintergrundsystem, Host) an ein untergeordnetes System (z. B. Terminal). Das Gegenteil ist der → <i>Upload</i> .
DPA (<i>differential power analysis</i>)	Die differentielle Leistungsanalyse ist eine Angriffsmethode auf → <i>Chipkarten</i> und stellt eine Verbesserung der einfachen Leistungsanalyse (→ <i>SPA</i>) dar. Dabei wird wiederholt mit hoher zeitlicher Auflösung der Stromverbrauch eines Mikrocontrollers für eine bestimmte Operation mit bekannten Daten gemessen und durch Mittelwertbildung das Rauschen eliminiert. Dann wird der Stromverbrauch mit unbekannten Daten gemessen und anschließend durch Differenzbildung der Messwerte der bekannten und der unbekannten Daten auf die unbekannten Daten geschlossen. Bekannt wurde die DPA durch eine Veröffentlichung von Paul Kocher, Joshua Jaffe und Benjamin Jun im Juni 1998 [Kocher 98]. ¹⁷
DRAM (<i>dynamic random access memory</i>)	RAM-Speicher in dynamischer Bauweise; benötigt zum Erhalt des Speicherinhalts eine konstante Stromversorgung sowie eine zyklische Wiederauffrischung des Inhalts. DRAM-Speicher sind aus Kondensatoren aufgebaut. Sie be-

¹⁷ Siehe auch Abschnitt 8.2.4.1 „Angriffe auf der physikalischen Ebene“.

	nötigen weniger Platz auf dem Chip als SRAM-Speicher und sind deshalb billiger. Allerdings ist die Zugriffszeit auf SRAMs geringer.
Dual-Band-Mobiltelefon (<i>dual band mobile</i>)	Ein Dual-Band-Mobiltelefon kann in zwei Frequenzbändern (z. B. 900 MHz und 1800 MHz) arbeiten.
Dual-Interface-Karte (<i>dual interface card</i>)	Bezeichnung für eine → <i>Chipkarte</i> mit kontaktbehafteter und kontaktloser Schnittstelle für die Datenübertragung von und zur Chipkarte.
Dual-Mode-Mobiltelefon (<i>dual mode mobile</i>)	Mobiltelefon, das in zwei verschiedenen Mobilfunksystemen (z. B. GSM und AMPS) arbeiten kann.
Dual-Slot-Handy (<i>dual slot mobile</i>)	Bezeichnung für ein Mobiltelefon, das neben der Benutzerkarte (z. B. SIM) noch eine von außen zugängliche Kartenkontaktiereinheit für üblicherweise ID-1-Chipkarten (→ <i>Chipkarte</i>) besitzt. Mit Dual Slot Handys lassen sich beispielsweise Zahlungen über das Mobilfunknetz mit bestehenden elektronischen Geldbörsen auf Chipkarten abwickeln.
Dual-Slot-Lösung (<i>dual slot solution</i>)	Chipkarten-Anwendung, die auf der Nutzung der zweiten Kartenkontaktiereinheit eines Dual-Slot-Handies beruht.
duplizieren	Das Übertragen von echten Daten auf eine zweite Karte zum Zwecke der Herstellung einer oder mehrerer identischer (d. h. geklonter) Karten. Der Begriff ist in der Regel identisch mit → „klonen“.
Dynamic STK (<i>dynamic SIM application toolkit</i>)	Veralteter Ausdruck für Microbrowser-Lösungen (→ Browser) nach der → <i>SIM Alliance</i> Spezifikation.
ECBS (European Committee for Banking Standards) [ECBS]	Eine im Jahr 1992 gegründete europäische Organisation zur Entwicklung technischer Lösungen und Standards für eine europaweite → <i>interoperable</i> Infrastruktur für Zahlungsverkehrssysteme.
ECC (<i>elliptic curve cryptosystem</i>)	Die Bezeichnung für Kryptosysteme (d. h. in der Regel Kryptoalgorithmen) auf der Grundlage von elliptischen Kurven.
ECC (<i>error correction code</i>)	Prüfsumme über Daten. Mit einem ECC können Fehler in den Daten mit einer bestimmten Wahrscheinlichkeit erkannt und ggf. auch fehlerfrei korrigiert werden.
E-Commerce (<i>Electronic Commerce</i> , elektronischer Geschäftsverkehr)	Unter diesem Begriff versteht man alle Formen von Dienstleistung, Handel und dazugehörigem Zahlungsverkehr in offenen Netzen, d. h. vor allem im Internet. Werden bei E-Commerce mobile Endgeräte verwendet so spricht man von → <i>M-Commerce</i> .
EDC (<i>error detection code</i>)	Prüfsumme über Daten. Mit einem EDC können Fehler in den Daten mit einer bestimmten Wahrscheinlichkeit erkannt werden. Ein typisches Beispiel für einen EDC ist die XOR- oder CRC-Prüfsumme bei verschiedenen Datenübertragungsprotokollen.
EDGE (<i>enhanced data rates for GSM and TDM evolution</i>)	EDGE ist als letzte Evolutionsstufe für GSM-Netzwerke vorgesehen. Unter Beibehaltung der existierenden Netzinfrastruktur können durch ein anderes Modulationsverfahren auf der Luftschnittstelle GSM-Mobiltelefone nach EDGE-Standard mit einer Datenübertragungsrate von bis zu 384 kBit/s an Basisstationen angebunden werden.
EEPROM (<i>electrical erasable read only memory</i>)	Eine nichtflüchtige Speicherart, die in → <i>Chipkarten</i> Verwendung findet. EEPROMs sind in Speicherseiten, Pages genannt, aufgeteilt. Die Seitengröße wird auch als → <i>Granularität</i> bezeichnet. Der Inhalt der Speicherseiten kann verändert und als Ganzes gelöscht werden, wobei es aber eine physikalisch be-

	dingte Obergrenze der Anzahl der schreibenden bzw. löschenden Zugriffe gibt. ¹⁸ Der Schreibvorgang bei EEPROM-Zellen findet durch den Fowler-Nordheim-Effekt statt und nicht durch Hot-Electron-Injection wie bei → <i>Flash-EEPROM</i> . EEPROM-Zellen haben eine typische Schreibzeit von 3 ms pro Speicherseite.
EF (<i>elementary file</i>)	Die eigentlichen Datenspeicher im Dateibaum einer Chipkarte. Sie können entweder die Eigenschaft „working“ (d. h. für den Gebrauch durch das Terminal) oder „internal“ (d. h. für den Gebrauch durch das Betriebssystem der Chipkarte) haben und besitzen eine interne Struktur (transparent, linear fixed, linear variable, cyclic, ...). ¹⁹
Einlagenkarte (<i>mono layer card</i>)	Karte, die sich aus einer einzigen Schicht Kunststoffolie zusammensetzt (→ <i>Mehrlagenkarte</i>).
Einwegfunktion (<i>one-way function</i>)	Mathematische Funktion, die sich einfach berechnen lässt, deren Umkehrfunktion aber einen sehr großen Rechenaufwand erfordert.
elektronische Geldbörse (<i>electronic purse</i> , <i>e-purse</i>)	Karte mit Chip, die vor der Bezahlung mit einem Geldbetrag aufgeladen werden muss. Diese Bezahlung wird oft als „pay before“ bezeichnet. Die typischen Beispiele für elektronische Geldbörsen sind die deutsche Geldkarte, die österreichische Quick Börse, Visa Cash, Proton oder Mondex. Elektronische Geldbörsen können die Eigenschaft von Purse-to-Purse-Transaktionen (siehe auch Purse-to-Purse-Transaktion) haben. ²⁰
elektronischer Scheck (<i>electronic cheque</i>)	Variante einer → <i>elektronischen Geldbörse</i> , die mit festen und nicht stückelbaren Beträgen arbeitet. Diese Bezahlung wird oft als „pay before“ bezeichnet. ²¹
Embossing (<i>embossing</i>)	→ <i>Hochprägung</i>
Emulator (<i>emulator</i>)	Gerät (d. h. eine Hardware), das die Funktionsweise eines anderen Gerätes (des Zielsystems) nachahmt. Die Nachahmung mittels Software bezeichnet man hingegen als → <i>Simulator</i> . Emulatoren werden oft zur Entwicklung von Software für noch nicht existierende Zielsysteme eingesetzt. Ein Chipkarten-Emulator ist somit eine Hardwareschaltung, die eine echte Chipkarte auf elektrischer und logischer Ebene vollständig nachbildet. Aufgrund der Realisierung eines Großteils der Funktionalität in Hardware sind Emulatoren in der Regel schneller, d. h. echtzeitnäher, als Simulatoren.
EMV (Europay, Mastercard, Visa) [EMV]	Eine gemeinsame Spezifikation für Zahlungsverkehrskarten mit Chip sowie dazugehörige Terminals der Firmen Europay, Master Card, Visa und American Express. Diese Spezifikationen sind zum weltweiten Industriestandard für Kredit, Debit- und Börsenkarten avanciert und somit das Pendant des Zahlungsverkehrs zur Telekommunikationsnorm GSM 11.11.
EMV-Spezifikation	→ <i>EMV</i>
Ende-zu-Ende- Verbindung (<i>end-to- end-connection</i>)	Die direkte Kommunikation zweier Instanzen unter Zuhilfenahme von Kommunikationspfaden einer oder mehrerer weiterer Instanzen, die jedoch den Informationsgehalt des eigentlichen Datenaustausches nicht verändern. Ist die Kommunikation der zwei Ursprungsinstanzen kryptografisch gesichert, so spricht man von → <i>tunneling</i> . Ein typisches Beispiel für eine Ende-zu-Ende-Verbindung ist die direkte Kommunikation eines Anwendungsanbieters mit einer SIM nach GSM 03.48.

¹⁸ Siehe auch Abschnitt 3.4.2 „Speicherarten“.

¹⁹ Siehe auch Abschnitt 5.6.4 „Dateistrukturen von Efs“.

²⁰ Siehe auch Abschnitt 12.1.2 „Elektronisches Geld“.

²¹ Siehe auch Abschnitt 12.1.2 „Elektronisches Geld“.

Endianness	Gibt die Reihenfolge der Bytes innerhalb eines Bytestrings an. Big Endian besagt, dass sich das höherwertigste Byte am Anfang und folglich das niederwertigste am Schluss der Kette von Bytes befindet. Bei little Endian ist die Reihenfolge umgedreht, d. h. das niedrigstwertige Byte ist am Anfang und das höchstwertige am Schluss.
enrollment	Prozess der ursprünglichen Aufnahme der biometrischen Daten eines → <i>Chipkartenbenutzers</i> und die Einbringung in die entsprechende Chipkarte. Die beim enrollment in der Chipkarte gespeicherten Daten sind dann im Folgenden die Basis für die weiteren biometrischen Benutzeridentifizierungen.
EP SCP	→ <i>SMG9</i>
EPROM (<i>erasable read only memory</i>)	Nichtflüchtige Speicherart, die früher in Chipkarten Verwendung fand und vollständig durch die → <i>EEPROM</i> -Technologie abgelöst ist. Ein EPROM kann nur durch UV-Licht gelöscht werden, weshalb es im Chipkartenbereich nur als WORM-Speicher (<i>write once, read multiple</i>) verwendet werden kann. ²²
ETS (<i>European telecommunication standard</i>)	Bezeichnung der von → <i>ETSI</i> herausgegebenen Normen, die sich in erster Linie mit europäischer Telekommunikation beschäftigen.
ETSI (<i>European telecommunications standards institute</i>) [ETSI]	Normungsinstitut der europäischen Telekommunikationsgesellschaften mit Sitz in Sophia Antipolis, Frankreich; es beschäftigt sich mit der Normung im Bereich der europäischen Telekommunikation. Die im Chipkartenbereich wichtigsten ETSI-Normen sind die Normenreihe für GSM (z. B. GSM 11.11 für die SIM) und für UMTS (z. B. TS 31.102 für die USIM-Anwendung). Die Expertentreffen der ETSI-Normungsgruppen finden in der Regel an den unterschiedlichsten (sehwerten) Orten in Europa oder, je nach Gastgeber, weltweit statt, weshalb manche Menschen der festen Überzeugung sind, dass die Abkürzung ETSI eigentlich für <i>European travel and sightseeing institute</i> steht.
etu (<i>elementary time unit</i>)	Die Dauer eines Bits bei der Datenübertragung zu Chipkarten. Die absolute Zeitdauer für ein etu ist nicht festgelegt, sondern in Abhängigkeit von der Chipkarte angelegten Takt und des Clock Rate Conversion Factors definiert.
Eurosmart [Eurosmart]	Die 1994 gegründete Interessenvertretung der europäischen Chipkartenhersteller mit Sitz in Brüssel. Die Aufgaben von Eurosmart sind die Förderung und Standardisierung (→ <i>Standard</i>) von → <i>Chipkarten</i> und Chipkarten-Systemen, die Bildung eines Forums für den Austausch von Marktdaten und technischen Daten sowie die Etablierung von Verbindungen zu nationalen und internationalen Standardisierungsgremien.
Evaluierung (<i>evaluation</i>)	Die unparteiliche, objektive, wiederholbare und reproduzierbare Bewertung von informationstechnischen Systemen (d.h. Hardware und/oder Software) durch eine vertrauenswürdige Instanz gemäß den Vorgaben eines Kriterienkatalogs. Das zu evaluierende informationstechnische System wird → <i>Target of Evaluation</i> genannt. International gebräuchliche Kriterienkataloge zur Evaluierung von → <i>Chipkarten</i> und Chipkartensystemen sind → <i>ITSEC</i> und Common Criteria (→ <i>CC</i>).
f1, f2, f3, f4, f5 (<i>function 1 ... 5</i>)	Bezeichnungen für kryptografische Funktionen bei → <i>UMTS</i> zur Authentisierung von Netzwerk und → <i>USIM</i> als auch zum Aufbau einer kryptografisch gesicherten Datenübertragung auf der Luftschnittstelle. Der Kern dieser <i>Security Functions</i> ist ein symmetrischer → <i>Kryptalgorithmus</i> , der mit zusätzlich verknüpften Eingangswerten parametrisiert werden kann. Als Beispielalgorithmus ist seitens der USIM-Spezifikation für f1 bis f5 der MILENAGE-Algorithmus vorgeschlagen, der in seinem Kern auf dem → <i>AES</i> basiert.
Fab (<i>fab</i>)	Halbleiterproduktionsstätte.

²² Siehe auch Abschnitt 3.4.2 „Speicherarten“.

face	Das <i>face</i> eines Halbleiters ist diejenige Seite des Chips, auf der sich die halbleitertechnisch erzeugten Funktionsstrukturen befinden. So bedeutet beispielsweise eine Face-to-face-Kontaktierung, dass zwei Chips in ihren Funktionsstrukturen so zusammengelegt werden, dass die Chips elektrisch miteinander verbunden sind.
Falltür (<i>trap door</i>)	Ein vorsätzlich angelegter Mechanismus in einer Software oder in einem Algorithmus, mit dem Sicherheitsfunktionen oder Schutzmechanismen umgangen werden können.
FAT (<i>file allocation table</i>)	Dateizuordnungstabelle, die eine spezielle Methode zur Dateiverwaltung bezeichnet. Dabei wird der zu verwaltende Speicher in Speicherabschnitte (so genannte Cluster) aufgeteilt und Informationen über die Belegung und die Adressen dieser Speicherabschnitte in der FAT verwaltet.
fault tree analysis	Beim Testen jene Methode, bei der zur Fehlersuche jeder mögliche Programmablauf im Programmcode durchlaufen wird.
FD/CDMA (<i>frequency division/code division multiple access</i>)	→ CDMA
FDMA (<i>frequency division multiple access</i> – Frequenzvielfachzugriff)	Ein → <i>Vielfachzugriffsverfahren</i> zur parallelen Datentübertragung von mehreren Sendern zu einem Empfänger auf verschiedenen Frequenzbändern. Dazu erhält jeder Sender aus dem ganzen zur Verfügung stehenden Frequenzbereich ein Frequenzband, auf dem er exklusiv senden darf. FDMA wird bei vielen Mobiltelefonanwendungen auf der Luftschnittstelle zwischen Mobiltelefon und Basisstation benutzt, wie beispielsweise im deutschen C-Netz. ²³
Fehlbedienungszähler (<i>error counter</i>)	Zähler, der Schlechtfälle erfasst und von dem es abhängt, ob ein bestimmtes Geheimnis (PIN oder Schlüssel) weiterhin benutzt werden kann. Erreicht der Fehlbedienungszähler den Maximalwert, so ist das Geheimnis gesperrt und kann nicht mehr verwendet werden. Der Fehlbedienungszähler wird üblicherweise auf null zurückgesetzt, wenn die Aktion erfolgreich verlaufen ist (d. h. im Gutfall).
FIB (<i>focused ion beam</i>)	Gerät zur Erzeugung eines gebündelten Ionenstrahls zum Abtragen oder Aufbau von Material auf Halbleitern.
FID (<i>file identifier</i>)	Ein zwei Byte großes Merkmal für eine Datei, das zur Selektion dieser Datei benutzt werden kann. Sowohl MF als auch DF und EF besitzen einen FID. Der FID des MFs ist immer '3F00'. ²⁴
FIPS (<i>Federal Information Processing Standard</i>)	Die von → <i>NIST</i> herausgegebenen US-amerikanischen Normen.
Firewall	Instanz (Hardware und/oder Software), die eine sicherheitstechnische Trennung zwischen bestimmten → <i>Anwendungen</i> oder Instanzen realisiert. Eine Firewall kann beispielsweise zwei Anwendungen auf einer Chipkarte in einer Art und Weise separieren, dass diese nicht über die Firewall hinweg unberechtigterweise auf Daten zugreifen können. Die deutsche Übersetzung Brandschutzmauer, welche sich zwischen zusammenhängenden Gebäuden befindet, beschreibt plastisch den Zweck und die Funktionsweise einer Firewall.
Flash	Gebräuchliche Kurzform für einen → <i>Flash-EEPROM-Speicher</i> .

²³ Siehe auch Abschnitt 13.1.1 „Vielfachzugriffsverfahren“.

²⁴ Siehe auch Abschnitt 5.6 „Dateien in der Chipkarte“.

Flash-EEPROM (<i>flash electrical erasable read only memory</i>)	Nichtflüchtige Speicherart, die in Chipkarten in Zukunft Verwendung finden wird. Ein Flash-EEPROM ähnelt in Funktionalität und halbleitertechnischem Aufbau dem → <i>EEPROM</i> , doch findet bei Flash-EEPROM-Zellen der Schreibvorgang durch eine so genannte Hot-Electron-Injection statt und nicht durch den Fowler-Nordheim-Effekt wie beim EEPROM. Bei der Hot-Electron-Injection werden durch eine hohe Potenzialdifferenz zwischen Source und Drain schnelle Elektronen erzeugt, von denen ein Teil die Tunnel-Oxidschicht durchdringt und im Floating Gate gespeichert wird. Dieser Effekt reduziert die Schreibzeit auf ca. 10 µs. Flash-EEPROMs eignen sich aufgrund ihrer großen Speicherseiten (z. Zt. typisch 128 Byte) sehr gut als Ersatz für maskenprogrammiertes → <i>ROM</i> . ²⁵
Floorlimit	Das Floorlimit gibt an, ob eine Zahlung von einer dritten Instanz autorisiert (→ <i>Autorisierung</i>) werden muss. Unterhalb dieser Grenze ist eine Autorisierung nicht notwendig, oberhalb dieser Grenze muss eine Autorisierung vorgenommen werden, da sonst die Zahlung nicht möglich oder garantiert ist.
flüchtiger Speicher (<i>volatile memory</i>)	Speicherart (z. B. RAM), die ihren Inhalt nur bei dauernder Stromzufuhr behält.
Footprint	exakter: Memory Footprint, bezeichnet die Aufteilung eines Speichers für bestimmte Zwecke.
Foundry	Halbleiterproduktionsstätte, die auf Vertragsbasis von Dritten entwickelte Halbleiter herstellt.
FPLMTS (<i>future public land mobile telecommunication service</i>)	→ <i>IMT-2000</i>
FRAM (<i>ferroelectric read access memory</i>)	Nichtflüchtige Speicherart, die in → <i>Chipkarten</i> sehr selten Verwendung findet. FRAMs sind in Speicherseiten, so genannte Pages, aufgeteilt. Die Seitengröße wird auch als → <i>Granularität</i> bezeichnet. Für diese Speicherart benutzt man zur Speicherung von Informationen die Eigenschaften von ferroelektrischen Substanzen, die zwischen Control Gate und Floating Gate eingebracht werden. FRAM-Zellen haben eine typische Schreibzeit von 100 ns pro Speicherseite und benötigen keine spezielle Löschspannung. Allerdings ist bei FRAM-Speicher die Anzahl der Lesezyklen limitiert, und die Herstellung beinhaltet schwer beherrschbare Prozessschritte, weshalb diese Speichertechnologie bei Chipkarten-Mikrocontrollern bisher wenig Anwendung findet.
Frame	Folge von Datenbits und optional Fehlererkennungsbits mit Rahmenbegrenzungen am Anfang und Ende. Frames für die kontaktlose Datenübertragung bei Chipkarten sind in ISO/IEC 14443 definiert.
Garbage Collection	Sammelt den von einer → <i>Anwendung</i> nicht mehr benutzten Speicher und stellt ihn wieder als Freispeicher zur Verfügung. Die Garbage Collection wurde früher mittels Interrupt zum normalen Programmablauf realisiert. In modernen Computersystemen ist die Garbage Collection ein Thread niedriger Priorität, welcher ständig den Speicher auf nicht mehr benötigte Bereiche durchsucht und ihn dann wieder freigibt.
Geldkarte	Markenname der seit 1996 in Deutschland eingeführten elektronischen Geldbörse. Die Geldkarte ist sowohl die Bezeichnung für die Anwendung auf einer → <i>Multiapplication-Chipkarte</i> als auch für Chipkarte an sich. Das Chipkarten-Betriebssystem für die Geldkarte bzw. für die Debit-Funktionalität ist → <i>SEC-COS</i> (<i>security card operating system</i>).
geschlossene Anwendung (<i>closed application</i>)	Eine → <i>Anwendung</i> auf einer Chipkarte, die nur dem Anwendungsbetreiber zur Verfügung steht und nicht allgemein verwendet werden kann.

²⁵ Siehe auch Abschnitt 3.4.2 „Speicherarten“.

geschlossene Börse (<i>closed purse</i>)	Realisation einer geschlossenen → <i>Anwendung</i> für eine elektronische Geldbörse. Sie kann nur in dem vom Anwendungsbetreiber freigegebenen Rahmen und nicht für allgemeine Zahlungstransaktionen verwendet werden.
Glitch	Sehr kurzer Spannungseinbruch oder eine sehr kurze Spannungserhöhung.
Global Platform [Global Platform]	Ein im Jahr 1999 gegründetes, international agierendes Gremium verschiedener Chipkartenfirmen zur Standardisierung von Technologien für → <i>Multiapplication-Chipkarten</i> . Die wichtigste von Global Platform herausgegebene Spezifikation ist die Open Platform (→ <i>OP</i>)-Spezifikation.
GPRS (<i>general packet radio system</i>)	Eine von → <i>ETSI</i> standardisierte Erweiterung von → <i>GSM</i> , um höhere Übertragungsraten für Daten bei mobilen Endgeräten zu realisieren. GPRS bietet eine → <i>paketorientierte</i> Verbindung mit einer Datenübertragungsrate bis zu 115,2 kBit/s durch Bündelung der 8 vorhandenen Zeitschlitz zu je 14400 Bit/s. Ein Mobiltelefon mit GPRS-Technologie ist hinsichtlich der Datenübertragung ständig im Netz eingebucht und damit zu jedem Zeitpunkt für Datenübertragungen verfügbar. Die Datenübertragungsrate wird dem aktuell notwendigen Kapazitätsbedarf dynamisch angepasst, so dass immer nur die aktuell benötigte Kapazität belegt wird. Aus diesem Grund eignet sich GPRS auch sehr gut für diskontinuierliche Datenübertragungen.
Granularität (<i>granularity</i>)	Bei → <i>EEPROM</i> -Speichern ein häufig benutzter alternativer Ausdruck für die Pagegröße. Beispielsweise hat ein EEPROM mit der Granularität 32 die Pagegröße von 32 Byte.
Greylist (<i>greylist</i>)	Liste in einer Datenbank, auf der alle → <i>Chipkarten</i> oder Geräte vermerkt sind, die unter Beobachtung stehen (→ <i>Sperrliste</i> , <i>Hotlist</i> , <i>Whitelist</i>).
Greybox Test (<i>greybox test</i>)	Mischform zwischen Whitebox Test und Blackbox Test. Die testende Instanz kennt dabei teilweise die internen Abläufe, Funktionen und Mechanismen der zu prüfenden Software.
GSM (<i>global system for mobile communications</i>)	Digitales, zellulares, betreiberübergreifendes, länderübergreifendes und bodengebundenes Mobilfunksystem der zweiten Generation (→ <i>2G</i>). Die für dieses Mobilfunksystem zugewiesenen Frequenzbereiche sind bei 900 MHz (GSM 900), 1800 MHz (GSM 1800) und 1900 MHz (GSM 1900). Das GSM-System ist durch eine Reihe von Spezifikationen definiert, deren Herausgeber → <i>ETSI</i> ist. Der Zusammenschluss der wesentlichen Netzbetreiber und Hersteller bei GSM ist die → <i>GSM Association</i> . Ursprünglich war GSM nur für einige Länder in Zentraleuropa als Nachfolger der länderspezifischen analogen Mobilfunksysteme geplant. GSM hat sich jedoch zu dem Weltstandard für Mobilfunk entwickelt. Die niedrigen Datenübertragungsraten im GSM-System von 9600 Bit/s bzw. 14400 Bit/s haben die Notwendigkeit der Verbesserung des Systems zu Tage gefördert. Der Evolutionspfad des GSM-Systems hinsichtlich Übertragungskapazität sieht deshalb als nächste Entwicklungsstufen das verbindungsorientierte → <i>HSCSD</i> und das paketorientierte → <i>GPRS</i> vor. Anschließend kann die Datenübertragungsrate bei GSM mit → <i>EDGE</i> nochmals erhöht werden. Der designierte Nachfolger von GSM wird → <i>UMTS</i> sein. ²⁶
GSM Association [GSM Association]	Weltweit agierendes Gremium mit Sitz in Dublin und London zur Abstimmung von Mobilfunksystemen. Es wurde 1987 in Kopenhagen gegründet und ist verantwortlich für die Entwicklung und den Einsatz der → <i>GSM</i> -Normen. Die GSM Association repräsentiert über 500 Netzbetreiber, Hersteller und Zulieferer der GSM-Industrie.
Guillochen	Die meist runden oder ovalen geschlossenen und miteinander verwobenen Linienfelder, die sich auch auf vielen verschiedenen Geldscheinen oder Aktien befinden. Diese Muster lassen sich aufgrund ihrer feinen Struktur hochqualitativ nur auf drucktechnischem Wege erzeugen und sind deshalb nur schwer kopierbar.

²⁶ Siehe auch Abschnitt 13.3 „Das UMTS-System“.

Gutfall (<i>good case</i>)	Bei einer logischen Entscheidung der Fall, der zum günstigeren oder beabsichtigten Ergebnis führt.
HAL (<i>hardware abstraction layer</i>)	Zwischenschicht in einem Betriebssystem zur Verbergung aller Hardware-spezifika der Zielplattform vor dem Rest des Betriebssystems. Ziel ist es, eine Portierung des Betriebssystems stark zu vereinfachen, indem beim Wechsel der Hardwareplattform lediglich Anpassungen innerhalb des HAL notwendig sind. ²⁷
Halbbyte (<i>half byte</i>)	→ <i>Nibble</i>
halbduplex (<i>half duplex</i>)	Datenübertragung, bei der miteinander kommunizierende Geräte zu einem Zeitpunkt entweder senden oder empfangen können. Das gleichzeitige Senden und Empfangen von Daten ist mit einer Halbduplex-Verbindung nicht möglich, dazu wird eine Vollduplex-Verbindung benötigt (→ <i>vollduplex</i>).
Händlerkarte (<i>merchant card</i>)	→ <i>Chipkarte</i> , die sich bei einem elektronischen Zahlungsverkehrssystem im Terminal des Händlers als Sicherheitsmodul befindet.
handover	Bezeichnet bei Mobilfunknetzen die unterbrechungsfreie Weitergabe eines Mobiltelefons von einer Zelle zur nächsten. Bei GSM wird ein handover immer vom Netz initiiert.
Happy-Day-Szenario (<i>happy day scenario</i>)	anderer Ausdruck für → <i>Gutfall</i>
Hardmaske (<i>hard mask</i>)	Dieser Begriff bedeutet, dass sich der gesamte → <i>Programcode</i> weitgehend im ROM befindet (→ <i>ROM-Maske</i>). Dies spart gegenüber einer Softmaske Platz, da ROM-Zellen wesentlich kleiner als EEPROM-Zellen sind. Dies hat aber den Nachteil, dass die volle Zeitspanne einer kundenindividuellen Halbleiterproduktion anfällt. Die Durchlaufzeit bei einer Hardmaske ist aus diesem Grund erheblich größer als bei einer Softmaske. Hardmasken werden üblicherweise für große Stückzahlen bei weitgehend einheitlicher Funktionalität der Chipkarten verwendet. Das Gegenteil einer Hardmaske ist die → <i>Softmaske</i> , bei der wesentliche Funktionen im EEPROM sind.
Hash-Funktion	Verfahren zur Komprimierung von Daten mittels einer Einwegfunktion, sodass die ursprünglichen Daten nicht rückrechenbar sind. Die Hash-Funktion liefert für einen Eingabewert beliebiger Länge einen Ausgabewert fester Länge und ist so beschaffen, dass eine Änderung der Eingangsdaten mit sehr hoher Wahrscheinlichkeit Auswirkungen auf den berechneten Hash-Wert (d. h. den Ausgabewert) hat. Ein typischer Vertreter der Hash-Algorithmen ist der SHA-1. Das Ergebnis einer Hash-Funktion ist der Hash-Wert, der oft auch als digitaler Fingerabdruck bezeichnet wird. ²⁸
HBCI (<i>home banking computer interface</i>)	Ein von der deutschen Kreditwirtschaft festgelegter Standard zur Realisierung von Homebanking in Deutschland unter optionaler Zuhilfenahme von Chipkarten.
Heimatnetz (<i>home net</i>)	Im Mobilfunkbereich jenes Mobilfunknetz, bei dessen Betreiber der Teilnehmer Kunde ist.
Hintergrundsystem (<i>background system</i>)	Hintergrundsysteme sind alle Computersysteme, die die Verarbeitung und Verwaltung von Daten ab der Hierarchie der Terminals übernehmen.
Hochprägung (<i>embossing</i>)	Teil der physikalischen Personalisierung, bei der Zeichen in einen Kartenkörper aus Kunststoff in einer solchen Weise geprägt werden, dass sie erhaben sind. Die Hochprägung wird in der Fachsprache auch Embossing genannt.
Hologramm	Fotografische Aufnahme bei der Holographie. Sie ist ein dreidimensionales Bild des fotografierten Objekts. Je nach Betrachtungswinkel des Beobachters wird das Objekt auf dem Hologramm auch unter verschiedenen Winkeln gesehen. Die

²⁷ Siehe auch Abschnitt 5.2 „Grundlagen“.

²⁸ Siehe auch Abschnitt 4.9 „Hash-Funktionen“.

	bei Karten üblicherweise verwendeten Hologramme sind Prägehologramme, bei denen auch bei alltäglichen Lichtverhältnissen ein halbwegs passables dreidimensionales Bild sichtbar ist.
Homezone	Bei Mobilfunknetzen die Bezeichnung für einen standortbezogenen Dienst (→ <i>Location Based Service</i>), bei dem in einem bestimmten Gebiet (üblicherweise der Nahbereich um die eigene Wohnung) die Gesprächsgebühren zu einem deutlich niedrigeren Tarif (i. d. R. Festnetztarif) abgerechnet werden. Ein Festnetzanschluss für Sprachverbindungen kann dadurch überflüssig werden.
horizontaler Prototyp	→ <i>Prototyp</i>
Hotlist	Liste in einer Datenbank, auf der alle → <i>Chipkarten</i> oder Geräte vermerkt sind, die wahrscheinlich manipuliert sind und keinesfalls akzeptiert werden dürfen (→ <i>Sperrliste, Redlist, Greylist, Whitelist</i>).
HSCSD (<i>high speed circuit switched data</i>)	Die verbindungsorientierte HSCSD-Technologie ist eine Ergänzung zum GSM-Standard, um durch zusätzliche Nutzung vorhandener Zeitslots auf der Luftschnittstelle durch Kanalbündelung eine theoretische Erhöhung der Datenübertragungsrate von bis zu $8 \cdot 9600 \text{ bit/s}$ (76 800 bit/s) (Uplink und Downlink) zu erreichen. Bestehende GSM-Netze lassen sich mit relativ geringem Aufwand durch Erweiterungen in den Basisstationen sowie spezielle Mobiltelefone auf HSCSD erweitern. Der Nachteil ist jedoch, dass der Bedarf an Übertragungskanälen maximal bis auf das 8-fache steigt.
HSM (<i>hardware security module</i> oder <i>host security module</i>)	→ <i>Sicherheitsmodul</i>
HTML (<i>hypertext markup language</i>)	Auf XML aufbauende logische Auszeichnungssprache für Hypertext-Dokumente im WWW (→ <i>WML, WWW, XML, Hypertext</i>).
Hybridkarte	Karte mit zwei unterschiedlichen Kartentechnologien. Typische Beispiele sind eine Karte mit Magnetstreifen und zusätzlichem Chip oder eine → <i>Chipkarte</i> mit optischem Speicher an der Kartenoberfläche.
Hypertext (<i>hypertext</i>)	Hypertext besitzt gegenüber normalem Text zusätzliche Querverweise (Hyperlinks) zu weiteren Stellen im Text oder anderen Dokumenten. Diese können durch eine entsprechende Benutzeraktion (i. d. R. durch Anklicken) aufgerufen werden. Gegenüber dem üblichen linear aufgebauten Fließtext wie beispielsweise in Büchern bietet Hypertext somit eine beliebige Vernetzung von Texten durch Querverweise. → <i>HTML</i> und → <i>WML</i> sind zwei typische Vertreter von Auszeichnungssprachen für Hypertext-Dokumente.
ICC (<i>integrated chip card</i>)	Nach ISO die normgerechte Bezeichnung für eine Karte mit Chip. Es kann sich dabei sowohl um einen Speicherchip als auch einen Mikrocontrollerchip handeln.
ID-1 Karte (<i>ID-1 card</i>)	Nach ISO 7810 das Standardformat für → <i>Chipkarten</i> . Sie hat eine Breite von $\approx 85,6 \text{ mm}$, eine Höhe von $\approx 54 \text{ mm}$ und eine Dicke $\approx 0,76 \text{ mm}$. ²⁹ Im Mobilfunkbereich wird mittlerweile jedoch vor allem das ID-000 Format (→ <i>Plug-In</i>) eingesetzt.
Identifizierung (<i>identification</i>)	Vorgang des Nachweises der Echtheit eines Gerätes oder einer Person durch Vergleich eines übergebenen Passwortes mit einem gespeicherten Referenzpasswort. Identifizierung kann als ein Spezialfall der → <i>Authentisierung</i> angesehen werden, wobei bei der Identifizierung eine Person authentisiert wird. Das zur Identifizierung verwendete Verfahren wird manchmal auch als → <i>cardholder verification method</i> bezeichnet.

²⁹ Siehe auch Abschnitt 3.1.1 „Formate“.

IEC (<i>International Electrotechnical Commission</i>) [IEC]	Die IEC wurde 1906 gegründet und hat ihren Sitz in Genf, Schweiz. Die Aufgabe der IEC ist die weltweite Normung im Bereich der Elektrotechnik.
IFD (<i>interface device</i>)	Nach ISO die normgerechte Bezeichnung für ein Chipkarten-Terminal.
Implanter	Fertigungsmaschine für Chipkarten, deren Aufgabe es ist, Module in die Kavität von Kartenkörpern einzufügen, in der Fachsprache Implantieren genannt.
Implementierung (<i>implementation</i>)	Im Sinne der Softwareentwicklung die Erstellung eines Programms anhand der beim → <i>Design</i> festgelegten Software-Architektur. Zur Implementierung gehört das → <i>Debugging</i> , jedoch nicht der → <i>Test</i> , der bei einer sequentiellen Softwareentwicklung der nächste und abschließende Schritt ist.
IMSI-Catcher	Gerät zum Abhören von GSM-Gesprächen durch Aufbau einer eigenen Zelle. Das Funktionsprinzip beruht darauf, dass sich der IMSI-Catcher zwischen Mobiltelefon und Basisstation dazwischenschaltet, er sich also gegenüber dem Mobiltelefon als Basisstation ausgibt und gegenüber der echten Basisstation als Mobiltelefon.
IMT-2000 (<i>international mobile telecommunication 2000</i>)	Konzept der → <i>ITU</i> für Mobilfunk-Lösungen der dritten Generation (→ <i>3G</i>) im 2000 MHz Frequenzbereich. IMT-2000 entstand 1995 als Nachfolger von FPLMTS (<i>future public land mobile telecommunication service</i>), ein 1985 begonnenes Konzept der ITU für Mobilfunk, das sich jedoch nicht in der ursprünglichen Form als weltweit einheitliches System durchsetzen ließ. Eine mögliche Realisierung von IMT-2000 ist → <i>UMTS</i> .
Individualisierung (<i>individualisation</i>)	→ <i>Personalisierung</i>
Initialisierer (<i>initializer</i>)	Instanz, die die → <i>Initialisierung</i> durchführt.
Initialisierung (<i>initialisation</i>)	Ds Laden der festen und personenunabhängigen Daten einer → <i>Anwendung</i> in das EEPROM. Ein Synonym für Initialisierung ist die Vorpersonalisierung.
Inlettfolie	Folie, die sich nach dem Zusammenlaminiieren aller Folien im Innern des Kartenkörpers befindet. Ein Synonym für Inlettfolie ist Kernfolie. In der Regel wird die Inlettfolie zwischen zwei Deckfolien einlaminiert und bildet so mit den beiden äußeren Folien die Karte. Die Inlettfolie ist oft Träger von Sicherheitsmerkmalen oder elektrischen Bauteilen, wie beispielsweise der Spule für kontaktlose Chipkarten.
Instrumentierung	Einfügen von speziellem → <i>Programmcode</i> in ein zu analysierendes Programm, um Abläufe und Aufrufe in diesem Programm für Testing analysieren zu können. ³⁰
Intelligente Speicherkarte	Speicherkarte mit erweiterter Logikschaltung für zusätzliche Sicherheitsfunktionen, die den Speicherzugriff überwachen.
interoperabel (<i>interoperability</i>)	Dieses Adjektiv wird in der Chipkartenwelt für Lösungen benutzt, die nicht auf eine spezifische Chipkarten-Anwendung (siehe auch Anwendung) oder Geräte eines bestimmten Herstellers zugeschnitten sind. → <i>Offene Chipkarten-Betriebssysteme</i> sind in der Regel interoperabel. Das Gegenteil von interoperablen Lösungen sind → <i>proprietäre</i> Lösungen. Ein Beispiel für eine → <i>interoperable</i> Chipkarte ist die → <i>SIM</i> , die in allen → <i>GSM</i> -Mobiltelefonen gleichermaßen ohne Kompatibilitätsprobleme eingesetzt werden kann.
Interpreter (<i>interpreter</i>)	Programm, das eine Programmiersprache wie BASIC oder Java zur Laufzeit in eine von einem Mikroprozessor ausführbare Maschinensprache übersetzt und auch sofort ausführt. Aufgrund des zur Laufzeit stattfindenden Übersetzungsvorgangs sind interpretierte Programme immer langsamer als kompilierter

³⁰ Siehe auch Abschnitt 9.3.3 „Dynamische Tests von Betriebssystemen und Anwendungen“.

	→ <i>Programcode</i> . Interpreter lassen jedoch wesentlich hardwareunabhängigere Programme als Compiler zu.
ISDN (<i>integrated services digital network</i>)	Bezeichnung für ein international standardisiertes digitales Fernsprechnetz, das gleichermaßen Telefongespräche wie Datenübertragung unterstützt. Ein ISDN-Anschluss besteht aus zwei Basiskanälen mit jeweils 64 kBit/s Übertragungsrate und einem Steuerkanal mit 16 kBit/s Übertragungsrate.
ISO (<i>International Standardisation Organization</i>) [ISO]	Die ISO wurde 1947 gegründet und hat ihren Sitz in Genf, Schweiz. Die Aufgabe von ISO ist, die weltweite Normung zu unterstützen, um einen ungehinderten Austausch von Gütern und Dienstleistungen zu ermöglichen. Die erste ISO-Norm wurde 1951 veröffentlicht und beschäftigt sich mit Temperaturen bei Längenmessungen.
ITSEC (<i>Information Technique System Evaluation Criteria</i>)	Kriterienkatalog zur Entwicklung und → <i>Evaluierung</i> der Sicherheit von informationstechnischen Systemen im europäischen Bereich und wurden 1991 veröffentlicht. Die Weiterentwicklung der ITSEC und Vereinheitlichung mit diversen nationalen Kriterienkatalogen sind die → <i>Common Criteria</i> .
ITU (<i>International Telecommunications Union</i>) [ITU]	Internationale Organisation zur Koordinierung, Normung und Entwicklung von globalen Telekommunikationsdiensten mit Sitz in Genf. Die Vorgängerorganisation war die CCITT.
Java Card Runtime Environment (JCRE) (Java Card Laufzeitumgebung)	Java Card Laufzeitumgebung besteht im Wesentlichen aus der → <i>Java Card Virtual Machine</i> (JVM) und dem Java Card API.
Java Card Virtual Machine (JCVM)	(→ <i>Virtual Machine</i>) Ein – üblicherweise in Software – simulierter → <i>Mikroprozessor</i> , der die Aufgabe hat, den Java Bytecode auszuführen und die Klassen und Objekte zu verwalten. Zusätzlich stellt die Java Card Virtual Machine die Trennung der → <i>Anwendungen</i> durch Firewalls sicher und ermöglicht die gemeinsame Nutzung von Daten. Im Prinzip kann sie auch als eine Art → <i>Interpreter</i> gesehen werden. Eine Clean Room VM ist die Implementation einer Java VM auf der Grundlage von öffentlich zugänglichen Informationen, d. h. ohne zusätzliche lizenzbedingte Informationen von Sun. Clean Room Implementierungen der Java VM gelten im allgemeinen als befreit von anfallenden Lizenzzahlungen an Sun.
Java Card	→ <i>Chipkarte</i> mit → <i>Mikrocontroller</i> , die unter anderem eine → <i>Java Card Virtual Machine</i> und ein → <i>Java Card Runtime Environment</i> besitzt. Java Cards sind → <i>Multiapplication-Chipkarten</i> mit der Eigenschaft, dass sie Programme in Java verwalten und ausführen können. Streng genommen ist eine Java Card kein → <i>Chipkarten-Betriebssystem</i> , da u. a. beispielsweise in den Ursprungsspezifikationen keine Dateiverwaltung festgelegt ist. In der Praxis gilt jedoch die Java Card als Urtyp eines → <i>offenen Chipkarten-Betriebssystems</i> .
Java Development Kit (JDK)	Sammlung von Softwarewerkzeugen, die die Softwareentwicklung in Java ermöglichen.
Java	Eine von der Firma Sun entwickelte hardwareunabhängige und objektorientierte Programmiersprache (→ <i>objektorientierte Programmierung</i>), die im Bereich des Internets stark verbreitet ist. Java Source Code wird mit einem Compiler in einen standardisierten Bytecode übersetzt, der dann üblicherweise mit einer so genannten virtuellen Maschine auf der jeweiligen Zielhardware (Intel, Motorola, ...) und Betriebssystemplattform (Windows, MacOS, Unix, ...) interpretiert wird. Es gibt Mikroprozessoren (Pico Java), die den Bytecode von Java direkt ausführen können.
Java Card Forum [JCF]	Ein im Jahr 1997 gegründetes und international agierendes Gremium verschiedener Chipkartenfirmen zur Förderung der Java Card Technologie und zur Entwicklung der dazugehörigen Spezifikationen (→ <i>Javacard</i>).

Kaltreset (<i>cold reset</i>)	→ <i>Reset</i>
Karte (<i>card</i>)	Allgemeiner Begriff für ein in seinen physischen Abmessungen genormtes rechteckiges Stück Material mit abgerundeten Ecken. Eine Karte kann mit verschiedenen → <i>Kartenelementen</i> ausgestattet sein und damit auch einen Halbleiterchip (→ <i>Chipkarte</i>) besitzen. Der englische Begriff <i>card</i> hat im WML-Umfeld eine völlig andere Bedeutung (→ <i>Card</i>).
Kartenakzeptant (<i>card acceptor</i>)	Eine Instanz, bei der Karten für eine bestimmte Form von Interaktion (z. B. Bezahlung) verwendet werden können. Typisches Beispiel ist ein Händler, der Kreditkarten zur Bezahlung akzeptiert.
Kartenbenutzer (<i>card holder</i>)	Person, die eine Karte benutzt. Sie ist deshalb der Kartenbesitzer, aber nicht unbedingt der Karteneigentümer.
Kartenbesitzer	Person, die die tatsächliche Verfügungsgewalt über eine Karte hat. Der Kartenbesitzer muss nicht zwangsläufig der Karteneigentümer sein.
Karteneigentümer	Natürliche oder juristische Person, die die rechtliche Herrschaft über die Karte hat und mit dieser nach Belieben verfahren kann. Bei Kredit- und Debitkarten sind die kartenherausgebenden Banken oft die Eigentümer der Karten, die kartenbenutzenden Kunden sind dann lediglich die Kartenbesitzer.
Kartenelement (<i>card component</i>)	Zusätzliche Funktionseinheiten auf einer → <i>Karte</i> , wie beispielsweise Unterschriftsstreifen, → <i>Hochprägung</i> , → <i>Magnetstreifen</i> , Chip (→ <i>Speicherkarte</i> , <i>Mikroprozessorkarte</i>) und Tastatur (→ <i>System-on-Card</i>).
Kartenemittent	andere Bezeichnung für den → <i>Kartenherausgeber</i>
Kartenherausgeber (<i>card issuer</i>)	Auch Kartenemittent; Instanz, die für die Ausgabe von Karten verantwortlich ist. Bei Monoapplikationskarten ist der Kartenherausgeber in der Regel zugleich Anwendungsanbieter, muss es aber nicht zwangsläufig sein.
Kartenhersteller (<i>card manufacturer</i>)	Instanz, die Kartenkörper herstellt, in die er Module einbettet.
Karteninhaber	Besitzer einer Karte, meistens auch der Benutzer.
Kartenkörper	Kunststoffkarte, die als Halbfertigprodukt in nachfolgenden Produktionsschritten weiterverarbeitet wird und u. U. weitere Funktionselemente enthält (z. B. implantierter Chip).
Kartenleser (<i>card reader</i>)	Mechanisch und elektrisch einfach aufgebautes Gerät, das zur Aufnahme und galvanischen Kontaktierung einer → <i>Chipkarte</i> dient. Im Gegensatz zu Terminals haben Kartenleser kein Display und keine Tastatur. Unabhängig vom Term „Kartenleser“ können Kartenleser in der Regel auch zum Schreiben von Daten in Karten verwendet werden.
Kavität (<i>cavity</i>)	Die üblicherweise gefräste Aussparung im Kartenkörper für das zu implantierende Modul.
Kennzeichen (<i>tag</i>)	→ <i>Tag</i>
Kerckhoff-Prinzip	Das nach Auguste Kerckhoff (1835–1903) benannte Prinzip besagt, dass die gesamte Sicherheit eines Kryptoalgorithmus ausschließlich auf der Geheimhaltung des Schlüssels beruhen soll und nicht auf der Geheimhaltung des Kryptoalgorithmus.
Kernel (Kern)	Zentraler Teil eines → <i>Betriebssystems</i> , der grundlegende Betriebssystemfunktionen den darüberliegenden Schichten des Betriebssystems bereitstellt.
Kernfolie (<i>core foil</i>)	anderer Ausdruck für → <i>Inletfolie</i>
Kernspannung (<i>core voltage</i>)	Die direkt auf dem Chip verwendete Betriebsspannung des Prozessors oder Mikrocontrollers. Ist die Kernspannung geringer als die an den Chip angelegte Spannung, so muss sie von einem auf dem Chip befindlichen Spannungswandler ent-

	sprechend reduziert werden. Niedrige Kernspannungen sind wegen der notwendigen Spannungsfestigkeit bei zunehmend geringeren Strukturbreiten und zur Reduktion der chipinternen Kapazitäten bei den immer höher werdenden Taktfrequenzen notwendig. Die typische Kernspannung von Mikrocontrollern in 0,13 µm-Technologie ist beispielsweise 1,8 V.
Kinegramm	Ein Kinegramm zeigt unter verschiedenen Blickwinkeln unterschiedliche Darstellungen. Das Kinegramm kann einen scheinbaren und ruckartigen Bewegungsablauf zeigen, oder es zeigt unterschiedliche Motive in Abhängigkeit vom Betrachtungswinkel. Kinegramme sind ähnlich, aber nicht identisch den Hologrammen, die ein 3-dimensionales Bild zeigen.
Kippbild, Kinegramm	Vom Aufbau her ein Hologramm, dessen Abbildung sich sprunghaft mit dem Betrachtungswinkel ändert.
Klasse (<i>class</i>)	Im Sinne der → <i>objektorientierten Programmierung</i> eine Art von abstrakter Bauleitung für ein → <i>Objekt</i> , d. h. für dessen → <i>Attribute</i> , → <i>Methoden</i> und Beziehungen zu anderen Objekten.
Klon	→ <i>klonen</i>
klonen	Das vollständige Kopieren von ROM und EEPROM eines Mikrocontrollers. Klonen ist ein typischer Angriff auf ein Chipkartensystem. Das Ergebnis ist ein Mikrocontroller mit vollends identischem Speicherinhalten und wird Klon genannt.
Kollision	Eine Kollision entsteht, wenn zwei oder mehr kontaktlose Karten, die sich im Aktionsbereich eines Terminals befinden, gleichzeitig Daten an das Terminal senden, so dass dieses die empfangenen Daten nicht decodieren oder eindeutig zuordnen kann.
Kommando (<i>command</i>)	Im Bereich von → <i>Chipkarten-Betriebssystemen</i> eine Anweisung an die Chipkarte, eine bestimmte Aktion auszuführen. Das Ergebnis eines Kommandos ist eine Antwort (<i>response</i>) der Chipkarte, die mindestens den Status, optional aber auch dazugehörige Daten des ausgeführten Kommandos rückmeldet. Kommandos werden durch → <i>Kommando-APDUs</i> an die Chipkarte übermittelt und Antworten durch → <i>Antwort-APDUs</i> .
Kommando-APDU (<i>command-APDU</i>)	→ <i>Kommando</i> vom Terminal an die Chipkarte; besteht aus dem Kommando-Header und optional aus dem Kommando-Body. Der Kommando-Header setzt sich wiederum aus Class-, Instruction-, P1- und P2-Byte zusammen (→ <i>APDU</i>). ³¹
komplettieren (<i>completion</i>)	Das Vervollständigen des Betriebssystems durch Laden der EEPROM-Teile. Dies ermöglicht nachträgliche Änderungen und Anpassungen, ohne dass eine neue ROM-Maske erstellt werden muss. Beim Komplettieren werden in jede Chipkarte identische Daten geschrieben, es ist also dem Prinzip nach eine Art Initialisierung.
Kontaktfläche	Die an der Vorderseite einer → <i>Chipkarte</i> befindlichen sechs oder acht Kontaktflächen sind die elektrische Schnittstelle zwischen dem Terminal und dem Mikrocontroller in der Chipkarte. Alle elektrischen Signale werden über diese Kontakte geführt.
kontaktlose Karte (<i>contactless card</i>)	Kurzbezeichnung für eine → <i>Chipkarte</i> , bei der die Energie- und Datenübertragung berührungslos durch elektromagnetische Felder erfolgt (→ <i>CICC</i>).
Kreditkarte (<i>credit card</i>)	Karte mit oder ohne Chip, die einen Verfügungsrahmen aufweist und bei der die Bezahlung zeitlich nach Erhalt des Gutes oder der Dienstleistung stattfindet. Diese Bezahlung wird oft als „buy now – pay later“ bezeichnet. Das typische Beispiel für die Kartenart sind hochgeprägte Kreditkarten.

³¹ Eine detaillierte Darstellung findet sich im Abschnitt 6.5.1 „Struktur der Kommando-APDUs“.

Kryptoalgorithmus (<i>cryptoalgorithm</i>)	Rechenvorschrift mit mindestens einem geheimen Parameter, dem → <i>Schlüssel</i> , um Daten zu ver- oder entschlüsseln. Es gibt symmetrische Kryptoalgorithmen (z. B. DES-Algorithmus), die zur Ver- und Entschlüsselung den gleichen Schlüssel benutzen und asymmetrische Kryptoalgorithmen (z. B. RSA-Algorithmus), die zur Verschlüsselung einen öffentlichen Schlüssel und zur Entschlüsselung einen geheimen Schlüssel verwenden.
Kryptokoprozessor	Bezeichnet im Chipkartenbereich eine zusätzliche numerische Recheneinheit auf einem Mikrocontroller. Die Funktionalität dieses Koprozessors ist optimiert zur schnellen Berechnung von Secret-Key-Algorithmen (z. B. DES) und/oder Public-Key-Algorithmen (RSA, DSA, ECC).
kryptografischer Algorithmus (<i>cryptographic algorithm</i>)	Die mittlerweile selten verwendete Langform von → <i>Kryptoalgorithmus</i> .
Kryptokarte	→ <i>Mikroprozessorkarte</i>
Kundenkarte (<i>customer card</i>)	→ <i>Chipkarte</i> , die von Kunden in einem elektronischen Zahlungsverkehrssystem an den Terminals der Händler zum Bezahlen benutzt wird.
kuvertieren	Automatisches Falten und Einstecken eines Briefes in einen Briefumschlag.
Ladebeauftragter (<i>load agent</i>)	Instanz, welche das Laden von elektronischen Geldeinheiten auf eine elektronische Geldbörse vornimmt. Er ist sozusagen das Gegenstück zum Leistungsanbieter.
Laminieren (<i>lamine</i>)	Das Verkleben von dünnen Materialschichten unter Druck und Hitze. Karten werden in der Regel aus mehreren Kunststofffolien laminiert.
Lasercutter	Gerät zum Bohren und Schneiden mit einem energiereichen Laserstrahl in einer Genauigkeit von Teilen eines Mikrometers vorzugsweise auf Halbleitern.
Lasergravur (<i>laser engraving</i>)	Verfahren zur Schwärzung von speziellen Kunststoffschichten durch Verbrennen mit einem Laser. Dieses Verfahren wird umgangssprachlich oft auch „lasern“ genannt.
Lead-Frame Modul	Kostengünstige Module, bei denen die aus einer Kupferlegierung mit galvanisierter Goldoberfläche gestanzten Kontaktflächen durch einen Moldkörper aus Kunststoff zusammengehalten werden. Auf diesem wird dann mit Pick-and-Place der Chip aufgesetzt und im Wire-Bond-Verfahren mit den Rückseiten der Kontaktflächen verbunden. Anschließend schützt man den Chip noch mit einem Überzug aus lichtundurchlässigem Epoxidharz.
Lead-Frame-Verfahren	Eines der zurzeit kostengünstigsten Verfahren zur Modulherstellung, ohne Nachteile bei der mechanischen Stabilität dafür in Kauf nehmen zu müssen.
Leadtime	Die Zeit in der Halbleiterfertigung von der Maskenabgabe (→ <i>ROM-Maske</i>) bis zu den ersten Mustern.
Lebenszyklus (<i>life cycle</i>)	Die Summe der Lebensphasen einer → <i>Chipkarte</i> , beginnend von der Chip- und Kartenherstellung über die → <i>Personalisierung</i> bis zur Benutzung und dem logischen oder physischen Lebensende der Karte. Im Lebenszyklus einer Karte werden dann die einzelnen Phasen zur Festlegung von bestimmten Sicherheitsmechanismen und Funktionalitäten verwendet. Ein Beispiel für die diesbezügliche Einteilung des Lebenszyklus einer Karte ist die → <i>Open Platform-Spezifikation</i> .
Leistungsanbieter (<i>service provider</i>)	In einem Chipkartensystem derjenige, der Leistungen anbietet, die ein Benutzer in Anspruch nimmt und bezahlt. Beim Beispiel eines Zahlungsverkehrssystems mit elektronischen Geldbörsen ist er derjenige, der vom Börseninhaber für seine Ware oder Dienstleistung Geld von einer elektronischen Geldbörse erhält.

leitungsorientiert	→ <i>verbindungsorientiert</i>
Linker	Ein Linker hat die Aufgabe, die symbolischen Speicheradressen eines kompilierten oder assemblierten → <i>Programmcodes</i> in absolute oder relative Speicheradressen umzusetzen.
little Endian	→ <i>Endianness</i>
Loader (<i>loader</i>)	Programm, mit dessen Hilfe weitere Programme, beispielsweise über eine serielle Schnittstelle, nachgeladen werden können (→ <i>Bootloader</i>).
Location based Service (standortbezogener Dienst)	Mehrwertdienste für Mobiltelefonteilnehmer, die auf dem Wissen über die aktuelle geografische Position beruhen. Beispiele dazu sind lokale Wettervorhersagen, dynamisch an den aktuellen Standort angepasste Stadtpläne und integrierte Standortangaben bei Notrufen.
logische Kanäle (<i>logical channels</i>)	Logische Kanäle dienen dazu, mit mehreren → <i>Anwendungen</i> auf einer Chipkarte parallel und unabhängig voneinander Daten auszutauschen. Die Kommunikation findet jedoch nach wie vor über die nur einmal vorhandene serielle Schnittstelle mit Chipkarte statt, durch die logischen Kanäle können jedoch die Empfängeranwendungen für die → <i>APDUs</i> auf der Chipkarte eindeutig adressiert werden. ³²
M/Chip	Name einer → <i>EMV</i> -konformen Implementierung einer chipgestützten Debit-/Kreditkarte von Europay und Mastercard. Die Variante M/Chip Select verwendet neben symmetrischen Kryptoalgorithmen auch asymmetrische und stellt eine Obermenge von M/Chip light dar, welche eine vereinfachte Variante ist und nur symmetrische Kryptoalgorithmen benutzt.
MAC (<i>message authentication code</i> – Datensicherungscode)	Kryptografische Prüfsumme über Daten, mit der Manipulationen dieser Daten während der Übertragung erkannt werden können. Werden Daten während ihrer Ablage in einem Speicher mit einem MAC geschützt, so spricht man von einer CCS (→ <i>cryptographic checksum</i>).
Magnetkarte	Oft verwendete und sachlich nicht korrekte Kurzform des Begriffs → <i>Magnetstreifenkarte</i> .
Magnetstreifenkarte	Karte mit einem Magnetstreifen, auf dem Daten geschrieben und wieder gelesen werden können. Der Magnetstreifen enthält in der Regel drei Datenspuren mit unterschiedlicher Datenaufzeichnungsdichte. Spur 1 und 2 werden nach der Ausgabe an den Kartenbenutzer nur mehr gelesen, und Spur 3 darf auch im Feld noch geschrieben werden. Die magnetische Eigenschaft des magnetisierbaren Materials kann entweder hoch- oder niederkoerzitiv sein.
Maosco	→ <i>Multos</i>
Maske	Kurzform von → <i>ROM-Maske</i> .
M-Commerce (<i>mobile Commerce</i> , mobiler Geschäftsverkehr)	Alle Formen von Dienstleistung, Handel und dazugehörigem Zahlungsverkehr mit mobilen Endgeräten (z. B. Mobiltelefon, PDA). Werden stationäre Endgeräte verwendet, spricht man von → <i>E-Commerce</i> .
Mehrlagenkarte (<i>multi layer card</i>)	Karte, die sich aus mehreren Schichten Kunststoffolie zusammensetzt. Diese werden in die äußeren Deckfolien (<i>overlay foil</i>) und die Kernfolien (<i>core foils</i>) unterteilt (→ <i>Einlagenkarte</i>).
Memory Footprint	Die Aufteilung der Speicherbelegung bei einem Rechnersystem.
Methode (<i>method</i>)	Eine Methode im Sinne der → <i>objektorientierten</i> Programmierung ist eine Funktion für die Manipulation von Attributwerten (→ <i>Attribut</i>) eines → <i>Objekts</i> , die von der → <i>Klasse</i> des Objektes zur Verfügung gestellt wird.

³² Siehe auch Abschnitt 6.7 „Logische Kanäle“.

ME _X E (<i>mobile station execution environment</i>)	Framework zur Integration einer Java Virtual Machine (JVM) in Mobiltelefone. Es können dann Java-Programme in Mobiltelefone geladen und ausgeführt werden. Damit besteht die Möglichkeit, Zusatzanwendungen direkt im Mobiltelefon und nicht (wie zur Zeit üblich) auf der SIM zu realisieren.
MF	Das Master File im Dateisystem einer Chipkarte ist ein besonderes DF. Es ist das Wurzelverzeichnis des Dateibaums und wird automatisch nach einem Reset der Chipkarte selektiert.
Microbrowser	→ <i>Browser</i>
Mikrocontroller (<i>microcontroller</i>)	Ein Mikrocontroller besitzt auf einem Chip einen → <i>Mikroprozessor</i> , flüchtigen Speicher (→ <i>RAM</i>), nichtflüchtigen Speicher (→ <i>ROM</i> , <i>EEPROM</i> , <i>Flash-EEPROM</i>) und entsprechende Schnittstellen für die Kommunikation nach chip-extern. Er ist damit ein selbstständig arbeitsfähiger Computer auf einem einzelnen Chip. Mikrocontroller werden neben Chipkarten vor allem in der Steuerungstechnik eingesetzt.
Mikroprozessor	Wichtigste Funktionseinheit eines → <i>Mikrocontrollers</i> . Er löst die im Programm festgelegten Maschinenbefehle in die elementaren Mikrobefehle auf und führt diese aus. Ein Mikroprozessor enthält alle für die Befehlsabarbeitung notwendigen Register, ein Steuerwerk und das Rechenwerk. Das eigentliche Rechenwerk eines Mikroprozessors wird manchmal auch Prozessor genannt. Der Begriff CPU (<i>central processing unit</i>) wird oft als Synonym zu Mikroprozessor gebraucht.
Mikroprozessorkarte	Karte mit Chip, welcher einen → <i>Mikrocontroller</i> mit CPU, flüchtigem (→ <i>RAM</i>) und nichtflüchtigem (→ <i>ROM</i> , <i>EEPROM</i> , ...) Speicher besitzt. Mikroprozessorkarten können noch einen numerischen Koprozessor (→ <i>Kryptokoprozessor</i>) haben, um Public-Key-Kryptoalgorithmen schnell ausführen zu können. Diese Art von Karten werden manchmal auch Kryptokarten oder Kryptocontrollerkarten genannt.
MILENAGE-Algorithmus	Der symmetrische Beispielalgorithmus für die kryptografischen Funktionen f1 bis f5 (→ <i>fI</i>) der → <i>USIM</i> . Der MILENAGE-Algorithmus basiert in seinem Kern auf dem → <i>AES</i> .
MKT (Multifunktionales Kartenterminal)	Eine in Deutschland weit verbreitete Spezifikation für multifunktionale Chipkartenterminals (→ <i>Terminal</i>) und deren Anbindung an PCs mittels der → <i>CT-API</i> -Schnittstellenspezifikation. Es werden sowohl → <i>Speicherkarten</i> als auch → <i>Mikroprozessorkarten</i> unterstützt. Der Herausgeber der Spezifikation ist Teletrust Deutschland.
Modul	→ <i>Chipmodul</i>
Modulhersteller	Instanz, die Dice in Module einbaut und eine elektrische Verbindung durch Bon- den mit den Kontaktelementen herstellt.
Mondex [Mondex]	→ <i>Elektronische Geldbörse</i> auf Chipkarten mit der Möglichkeit von → <i>Purse-to-Purse-Transaktionen</i> .
Monoapplication-Chipkarte	Dieser Begriff sagt aus, dass sich auf einer Chipkarte nur eine → <i>Anwendung</i> befindet.
monofunktionale Chipkarte	Prozessorchipkarten, deren Betriebssystem nur eine einzige → <i>Anwendung</i> unterstützt und unter Umständen sogar auf diese Anwendung hin optimiert wurde. Verwaltungsfunktionen (z. B. Generieren und Löschen von Dateien) werden von monofunktionalen Chipkarten entweder überhaupt nicht oder nur in sehr eingeschränkter Form unterstützt.
MoU (<i>memorandum of understanding</i>)	Gemeinsame rechtliche Grundlage aller GSM-Netzbetreiber. Das Gremium hinter dem MoU ist die → <i>GSM Association</i> .

Multiapplication-Chipkarte (<i>multi application smart card</i>)	Dieser Begriff sagt aus, dass sich auf einer Prozessorchipkarte mehrere → <i>Anwendungen</i> befinden, z. B. eine Bankkarte mit Telefonfunktion.
multifunktionale Chipkarte (MFC, <i>multi functional card</i>)	(Üblicherweise) Prozessorchipkarten, die mehrere → <i>Anwendungen</i> unterstützen und die entsprechende Verwaltungsfunktionen für die Anlage und das Löschen von Anwendungen und Dateien haben.
Multitasking	Computersysteme, die Multitasking unterstützen, ermöglichen es, mehrere Programme quasi gleichzeitig auszuführen. Die parallel ausgeführten Programme befinden sich üblicherweise in einem von den anderen Programmen abgegrenzten und geschützten Adressraum und können nur über spezielle Mechanismen Daten miteinander austauschen. Multitasking ist nicht dem Multithreading gleichzusetzen, da dort ein einzelnes Programm quasi gleichzeitig mehrere Aufgaben ausführt. Ein Computersystem kann sowohl Multitasking als auch Multithreading unterstützen.
Multithreading	Computersysteme, die Multithreading unterstützen, ermöglichen es einem Programm, quasi gleichzeitig mehrere Aufgaben auszuführen. Die einzelnen Threads eines Programms benutzen dabei üblicherweise einen gemeinsamen Adressraum. Multithreading ist nicht dem Multitasking gleichzusetzen, da sich dort mehrere individuelle Programme in separierten Adressräumen parallel in der Ausführung befinden. Ein Computersystem kann sowohl Multithreading als auch Multitasking unterstützen.
Multos (<i>multiapplication operating system</i>)	Markenname eines offenen Multiapplication-Chipkarten-Betriebssystems (→ <i>Chipkarten-Betriebssystem</i> , <i>offenes Chipkarten-Betriebssystem</i>). ³³ Der Herausgeber der Spezifikationen, der Lizenzgeber und Betreiber der für Multos notwendigen Zertifizierungsdienste ist das Maosco Consortium [Maosco].
Namensraum (<i>namespace</i>)	Eine Menge von Namen, in der alle Namen eindeutig sind.
Nativcode (<i>native code</i>)	Programm in dem spezifischen Maschinencode für den Mikroprozessor, auf dem er ausgeführt wird.
NBS (<i>National Bureau of Standards</i>)	Die vor 1988 gültige Bezeichnung des → <i>NIST</i> .
NCSC (<i>National Computer Security Center</i>) [NCSC]	Das US-amerikanische NCSC ist eine Unterorganisation des NSA, zuständig für die Prüfung von Sicherheitsprodukten und Herausgeber von Kriterien für sichere Computersysteme, u. a. der TCSEC.
negative file	→ <i>Blacklist</i>
Nibble	Die vier höherwertigen oder niederwertigen Bits eines Bytes. Ein anderer Ausdruck für Nibble ist Halbbyte.
Nichtabstreitbarkeit (<i>non-repudiation</i>)	(I. d. R. kryptografische) Verfahren, die sicherstellen, dass der Empfänger den Erhalt einer Nachricht nicht leugnen kann. Der Sender der Nachricht kann damit beweisen, dass sie der Empfänger bekommen hat. Nichtabstreitbarkeit ist also das Analogon zum „Einschreiben mit Rückantwort“ bei der konventionellen Briefpost.
nichtflüchtiger Speicher (<i>non-volatile memory</i>)	Speicherart (z. B. ROM, EPROM, EEPROM), die ihren Inhalt auch ohne Stromzufuhr behält.
NIST (<i>National Institute of Standards</i>)	Das US-amerikanische NIST ist eine Abteilung des amerikanischen Wirtschaftsministeriums und zuständig für die US-nationale Normung von Informa-

³³ Siehe auch Abschnitt 5.14.2 „Multos“.

and Technology) [NIST]	tionstechnik. Bis 1988 trug es die Bezeichnung NBS. Das NIST ist der Herausgeber der FIPS-Normen.
Norm (<i>standard</i>)	Dokument, das technische Beschreibungen und/oder genaue Kriterien enthält, die als Regeln und/oder Definition von Eigenschaften verwendet werden, um dadurch sicherzustellen, dass Materialien, Produkte, Prozesse oder Leistungen für ihren Zweck verwendet werden können. In diesem Buch wird der Ausdruck „Norm“ durchgehend im Zusammenhang mit einen nationalen oder internationalen Normungsgremium (z. B. ISO, CEN, ANSI, ETSI) benutzt. Der deutsche Begriff Norm ist nicht mit dem deutschen Begriff → <i>Standard</i> gleichzusetzen.
NPU (<i>numeric processing unit</i>)	→ <i>Kryptokoprozessor</i>
NSA (<i>National Security Agency</i>) [NSA]	Die US-amerikanische NSA ist die offizielle Institution für Kommunikationssicherheit der amerikanischen Regierung. Sie ist direkt dem Verteidigungsministerium untergeordnet und hat u.a. die Aufgabe, ausländische Kommunikation abzuhören und zu decodieren. Die Entwicklung neuer Kryptoalgorithmen und die Beschränkung des Einsatzes von bestehenden fällt ebenfalls in das Aufgabengebiet dieser Behörde.
Null-PIN	→ <i>0-PIN</i>
Nutzdaten	Jene Daten, die direkt für eine → <i>Anwendung</i> notwendig sind.
Nutzen	Beim Druck die Zusammenfassung von kleinen zu bedruckenden Teilen (z. B. eine Karte) auf einem großen Bogen, der nach dem Bedrucken in einzelne Teile getrennt wird. Dadurch kann der Druckvorgang fertigungstechnisch optimiert werden, da die großen Bögen in einem Arbeitsschritt gefertigt werden können, anstatt in vielen einzelnen. Ein typischer Nutzen beim Druck von Karten besteht beispielsweise aus 42 Karten auf einer großen Kunststoffolie.
Objekt (<i>object</i>)	Im Sinne der → <i>objektorientierten</i> Programmierung ist ein softwaretechnisches Konstrukt, dessen Bauanleitung eine → <i>Klasse</i> ist, das Daten enthält, d. h. → <i>Attribute</i> hat, die mit in der Klasse definierten → <i>Methoden</i> gelesen und geändert werden können.
objektorientierte Programmierung (<i>object oriented programming</i>)	Die objektorientierte Programmierung basiert darauf, dass alle Daten einer Software in → <i>Objekten</i> abgelegt sind, die auch → <i>Methoden</i> bereitstellen, um diese Daten zu lesen oder zu ändern. Die Objekte werden durch → <i>Klassen</i> definiert. Ein zentraler Aspekt der objektorientierten Programmierung ist, dass die zu verarbeitenden Daten und nicht der Programmablauf wie bei der → <i>prozeduralen Programmierung</i> im Mittelpunkt steht. Typische objektorientierte Programmiersprachen sind C++ und → <i>Java</i> .
OCF (<i>open card framework</i>) [OCF]	Die OCF-Spezifikation beschreibt eine plattformunabhängige Java-basierte Schnittstelle zur Einbindung von Chipkarten in beliebige Anwendungen auf PCs. Die Voraussetzung ist, dass für das jeweils am PC benutzte Terminal ein passender Treiber vorhanden ist und die verwendete Chipkarte OCF-kompatibel ist.
Offcard-Anwendung (<i>offcard application</i>)	→ <i>Anwendung</i>
offene Anwendung	→ <i>Anwendung</i> auf einer Chipkarte, die unterschiedlichen Leistungsanbietern (z. B. Händlern, Dienstleistern) ohne notwendige Rechtsbeziehung untereinander zur Verfügung steht.
offene Börse	Realisation einer offenen → <i>Anwendung</i> für eine elektronische Geldbörse. Mit ihr können allgemeine Zahlungstransaktionen für unterschiedliche Leistungsanbieter getätigt werden.
offene Plattform (<i>open platform</i>)	→ <i>offenes Chipkarten-Betriebssystem</i> oder → <i>OP</i>

offenes Chipkarten-Betriebssystem (<i>open smart card operating system</i>)	Ein → <i>Chipkarten-Betriebssystem</i> wird als offen bezeichnet, wenn es die Möglichkeit bietet, unabhängig vom → <i>Betriebssystemhersteller</i> dritte → <i>Anwendungen</i> und Programme auf die Chipkarte zu laden und diese dort in sicherer Umgebung auszuführen. Die drei bekanntesten offenen Chipkarten-Betriebssysteme sind → <i>Multos</i> , → <i>Java Card</i> und → <i>Windows for Smart Cards</i> . Offene Chipkarten-Betriebssysteme sind in der Regel → <i>interoperabel</i> und nicht → <i>proprietär</i> . Ein anderer Ausdruck für offenes Chipkarten-Betriebssystem ist auch offene Plattform (<i>open platform</i>), welche aber nicht mit der Schnittstelle zur Verwaltung von Applikationen auf einer Chipkarte (→ <i>OP</i>) verwechselt werden darf.
Oncard-Anwendung (<i>oncard application</i>)	→ <i>Anwendung</i>
oncard Matching	Fähigkeit einer → <i>Chipkarte</i> , oncard oder offcard gemessene biometrische Daten mit in der Chipkarte gespeicherten Referenzdaten zwecks Benutzeridentifizierung (→ <i>Identifizierung</i>) zu vergleichen.
OP (<i>Open Platform</i>)	früher Visa Open Platform (VOP); eine ursprünglich von Visa International spezifizierte Schnittstelle im → <i>Chipkarten-Betriebssystem</i> mit der Aufgabe der Verwaltung von Chipkarten-Applikationen. Die Spezifikation umfasst unter anderem das Nachladen von Chipkarten-Applikationen, die Sicherstellung der Applikationslebenszyklen (→ <i>Lebenszyklus</i>) und die Verbindung einer Chipkarten-Applikation mit dem Chipkarten-Betriebssystem. Die OP-Spezifikation ist der internationale <i>de facto</i> Standard bei → <i>Multiplication-Chipkarten</i> und zum Applikationsmanagement. Der Herausgeber von OP ist mittlerweile das → <i>Global Platform</i> Gremium).
Optische Speicherkarte (<i>optical memory card</i>)	Karte, bei der Informationen in einer reflektierenden Schicht auf der Kartenoberfläche eingebrannt sind (analog einer CD).
OTA (<i>over the air</i>)	Bezeichnung für die Möglichkeit im → <i>GSM</i> - und → <i>UMTS</i> -System, zwischen Hintergrundsystem und → <i>SIM</i> über die Luftschnittstelle von Feststation und Mobilstation eine Ende-zu-Ende-Verbindung (→ <i>End-to-End-Verbindung</i>) aufzubauen. Damit können dann beispielsweise direkt vom Hintergrundsystem aus Kommandos transparent an die SIM gesendet werden. OTA ist auch eine der Grundlagen für alle → <i>Zusatzanwendungen</i> auf der SIM, da diese über die Luftschnittstelle auch direkt und transparent Daten mit übergeordneten Systemen austauschen können. Als Übertragungsdienst (→ <i>Bearer</i>) für OTA wird oft der Kurzmitteilungsdienst → <i>SMS</i> benutzt.
Package	Bei der Javacard ein → <i>Namensraum</i> und die kleinste Einheit innerhalb der Sprache Java. Ein Package kann über Klassen und Interfaces verfügen. ³⁴
Padding	Erweiterung eines Datenstrings durch Fülldaten mit dem Zweck, diesen Datenstring auf eine bestimmte Länge zu bringen. Meist muss die neue Länge des Datenstrings ein Vielfaches einer bestimmten Blocklänge (z. B. 8 Byte) sein, um dann den Datenstring beispielsweise durch einen Kryptoalgorithmus weiterverarbeiten zu können.
paketorientiert (<i>packet-switched</i>)	Bei paketorientierten Verbindungen werden die auszutauschenden Daten beim Sender in Pakete aufgeteilt, einzeln und u. U. auch über unterschiedliche Wege übertragen und beim Empfänger wieder zu den ursprünglichen Daten zusammengesetzt. Bei paketorientierten Verbindungen wird in der Regel die übertragene Datenmenge abgerechnet und nicht die Zeitdauer der Verbindung (→ <i>verbindungsorientiert</i>). Typische Beispiele für paketorientierte Verbindungen sind X.25 und GPRS.
parallele Datenübertragung	Hier werden mehrere (z. B. 8/16/32) Datenbits gleichzeitig auf eben so vielen Datenleitungen übertragen (→ <i>serielle Datenübertragung</i>).

³⁴ Siehe auch Abschnitt 5.14.1 „Java Card“.

Paritätsbit (<i>parity bit</i>)	Der wohl bekannteste Fehlererkennungscode (→ <i>EDC</i>) ist die Verwendung eines Paritätsbits, das an das zu schützende Byte angehängt wird. Vor der Berechnung der Parität muss festgelegt werden, ob mit gerader (<i>even</i>) oder ungerader (<i>odd</i>) Parität gearbeitet wird. Bei der geraden Parität wird das Paritätsbit so gewählt, dass die Gesamtzahl der Einsen im Datenbyte und im Paritätsbit eine gerade Zahl ergibt. Bei ungerader Parität muss die Zahl der Einsen in Datenbyte und Paritätsbit ungerade sein. Mit einem Paritätsbit ist die sichere Erkennung von einem falschen Bit pro Byte möglich. Die Korrektur eines Bitfehlers hingegen ist nicht möglich, da das Paritätsbit keine Aussage über die Position des veränderten Bits erlaubt.
Passivierung	Schutzschicht auf einem Halbleiter, um ihn vor Oxidation und anderen chemischen Vorgängen zu schützen. Vor einer physischen Manipulation des Halbleiters muss sie teilweise oder vollständig entfernt werden.
Patch	In der Softwareentwicklung ein kleines Programm, manchmal auch in Maschinencode geschrieben, das die Funktionalität eines vorgegebenen Programms ergänzt oder abändert. Patches werden in der Regel zur schnellen und unkomplizierten Korrektur von Programmfehlern benutzt. Sie werden entweder als → <i>work around</i> oder als → <i>bug fix</i> realisiert.
Patent	Dokument, das einem Erfinder das Recht zur alleinigen Verwertung der Erfindung für einen beschränkten Zeitraum und für ein oder mehrere bestimmte Länder einräumt. Die maximale Laufzeit eines Patents beträgt üblicherweise 20 Jahre.
Pay before	Dieser Ausdruck bezieht sich auf den Geldfluss bei Karten im Zahlungsverkehr. Vor dem Erhalt der gewünschten Ware oder Dienstleistung fließt das „echte“ Geld des Karteninhabers. Typische Vertreter von Pay before sind die → <i>elektronischen Geldbörsen</i> , die vor dem Einkauf mit elektronischem Geld geladen werden müssen. Im Telekommunikationsbereich wird diese Art der Bezahlung als → <i>prepaid</i> bezeichnet.
Pay later	Dieser Ausdruck bezieht sich auf den Geldfluss bei Karten im Zahlungsverkehr. Erst nach dem Erhalt der gewünschten Ware oder Dienstleistung fließt das „echte“ Geld des Karteninhabers. Typische Vertreter von Pay later sind die Kreditkarten, bei denen zum Teil erst Wochen nach dem Kauf das Geld vom Konto des bezahlenden auf das Konto des Händlers transferiert wird. Im Telekommunikationsbereich wird diese Art der Bezahlung als → <i>postpaid</i> bezeichnet.
Pay now	Dieser Ausdruck bezieht sich auf den Geldfluss bei Karten im Zahlungsverkehr. Beim Erhalt der gewünschten Ware oder Dienstleistung fließt das „echte“ Geld des Karteninhabers. Typische Vertreter dafür sind alle Debitkarten, wie beispielsweise die ec-Karte, die es ermöglicht, unmittelbar beim Kauf Geld vom Konto des Bezahlenden auf das Konto des Händlers zu transferieren.
PC/SC (<i>personal computer/smart card</i>) [PC/SC]	Die PC/SC-Spezifikation beschreibt eine plattform- und programmiersprachunabhängige Schnittstelle zur Einbindung von Chipkarten in beliebige Anwendungen. Die Voraussetzung ist, dass für das jeweils am PC benutzte Terminal ein passender Treiber vorhanden ist und die verwendete Chipkarte PC/SC-kompatibel ist. Im Dezember 1997 wurde die Version 1.0 der aus acht Teilen bestehenden „Interoperability Specification for ICCs and Personal Computer Systems“ veröffentlicht. ³⁵
PCD (<i>Proximity Coupling Device</i>)	Ein PCD ist ein Kartenterminal welches die Kommunikation mit einer kontaktlosen Karte (→ <i>PICC</i>) ermöglicht.
Persistenz (<i>persistent</i>)	Persistenz ist die Fähigkeit eines Objekts, über die Ausführungszeit eines Prozesses hinaus zu existieren, das Gegenteil ist die Transistenz. Persistente Objekte überdauern deshalb sowohl eine Sitzung als auch plötzliche Spannungsausfälle, ohne dass Daten verloren gehen oder inkonsistent werden.

³⁵ Siehe auch Abschnitt 11.4.1 „PC/SC“.

Personalisierer (<i>personaliser</i>)	Instanz, die die Personalisierung durchführt.
Personalisierung (<i>personalisation</i>)	Die Personalisierung ist der Vorgang der Zuordnung einer Karte zu einer Person. Dies kann einerseits durch die physikalische Personalisierung (z. B. Hochprägung, Lasergravur) oder auch durch die elektrische Personalisierung (d. h. Laden der personenbezogenen Daten in den Speicher der Chipkarte) geschehen. Das Wort Individualisierung für diesen Vorgang wäre der exaktere Begriff, da bei der elektrischen Personalisierung nicht zwangsläufig personenbezogene Daten in den Chip geschrieben werden, wie beispielsweise bei anonymen → <i>vorbezahlten SIMs</i> .
Phase 1, Phase 2, Phase 2+	Diese Phasen kennzeichnen einzelne aufeinander aufbauende Entwicklungsschritte im → <i>GSM</i> -System. In Phase 1 wurden u. a. die Basisdienste Sprachübertragung, Rufweiterleitung, → <i>Roaming</i> und → <i>SMS</i> realisiert. In Phase 2 kamen ab 1996 zu den Diensten der Phase 1 u. a. die Dienste Konferenzschaltung, Gesprächsweitergabe, Rufnummernübermittlung und GSM im 1 800 MHz Frequenzband hinzu. Anschließend wurden diese Dienste in Phase 2+ u. a. mit der Funktionalität des → <i>SIM Application Toolkit</i> , HSCSD (<i>high speed circuit switched data</i>) und → <i>GPRS</i> ergänzt.
PICC (<i>Proximity Integrated Circuit Card</i>)	Kontaktlose Chipkarte mit einer Reichweite von ungefähr 10 cm.
PIN (<i>personal identification number</i>)	Eine üblicherweise 4-stellige numerische geheime Zahl zur → <i>Identifizierung</i> einer Person. In der Telekommunikationswelt wird für die PIN meist die Bezeichnung CHV (<i>card holder value</i>) benutzt.
PIN-Pad	Im ursprünglichen Sinn die mechanisch und kryptografisch besonders geschützte Eingabetastatur bei Terminals. Im allgemeinen Sprachgebrauch wird oft auch das ganze Terminal als PIN-Pad bezeichnet.
PKCS #1 ... 15 (<i>public key cryptography standard number 1 ... 15</i>) [RSA]	Die Public Key Cryptography Standards sind von der Firma RSA Inc. veröffentlichte Regelwerke mit Fokus auf die Anwendung von asymmetrischen Kryptoalgorithmen, wie z. B. RSA. ³⁶
PKI (<i>public key infrastructure</i>)	Alle Einrichtungen, die zum Betrieb und zur Verwaltung einer auf asymmetrischer Kryptografie aufbauenden Datenspeicherung und Datenaustausch notwendig sind. Dies ist unter anderem eine → <i>Zertifizierungsinstanz</i> , eine → <i>Registrierungsinstanz</i> , ein → <i>Verzeichnisdienst</i> für Sperrlisten (→ <i>Zertifikatswiderrufsliste</i>), ein Zeitstempeldienst (→ <i>Zeitstempel</i>) und → <i>Signaturkarten</i> .
PLMN (<i>public land mobile network</i>)	<i>terminus technicus</i> für terrestrische Mobilfunknetze.
Plug-In (Einschubkarte)	Chipkarte in kleinem Format nach GSM 11.11 und TS 102.221, die vor allem im Mobilfunkbereich und als Sicherheitsmodul bei Terminals Verwendung findet. Die offizielle Formatbezeichnung für ein Plug-In nach ISO lautet ID-000 Format im Gegensatz zum größeren ID-1 Format (→ <i>ID-1 Karte</i>) der üblichen Chipkarten. Sie hat eine Breite von ≈ 25 mm, eine Höhe von ≈ 15 mm und eine Dicke $\approx 0,76$ mm. ³⁷
polling	Das laufende programmgesteuerte Abfragen eines Eingabekanals zur Detektion einer eingehenden Nachricht. Polling benötigt je nach Wiederholrate der stattfindenden Abfragen unter Umständen große Rechenleistung, weshalb man versucht, polling zu vermeiden und stattdessen eine von der Rechnerhardware unterstützte Abfrage mittels Interrupt bevorzugt.

³⁶ Siehe auch Abschnitt 4.7.2 „Asymmetrische Kryptoalgorithmen“.

³⁷ Siehe auch Abschnitt 3.1.1 „Formate“.

POS (<i>point of sale</i>)	Die Örtlichkeit, bei der ein bestimmtes Gut oder eine bestimmte Dienstleistung verkauft wird.
postpaid	Bezieht sich auf den Geldfluss bei Karten im Telekommunikationsbereich. Erst nach dem Erhalt der Dienstleistung (i. d. R. Telefonat oder Datenübertragung) fließt das echte Geld des Karteninhabers. Postpaid-Karten sind seitens der Zahlungsfunktion mit Kreditkarten vergleichbar. Im Zahlungsverkehr wird diese Art der Bezahlung als → <i>pay later</i> bezeichnet.
Power-On-Reset	→ <i>Reset</i>
prepaid (vorbezahlt)	Bezieht sich auf den Geldfluss bei Karten im Telekommunikationsbereich. Bereits vor dem Erhalt der Dienstleistung (i. d. R. Telefonat oder Datenübertragung) fließt das echte Geld des Karteninhabers. Prepaid-Karten sind seitens der Zahlungsfunktion mit elektronischen Geldbörsen vergleichbar. Im Zahlungsverkehr wird diese Art der Bezahlung als → <i>pay before</i> bezeichnet.
prepaid SIM (vorbezahlte SIM)	Vorbezahlte und in der Regel wiederaufladbare SIM. Die gesamte Funktionalität der Abrechnung und Wiederaufladung wird im Regelfall durch das Hintergrundsystem abgedeckt, hat also keine Auswirkungen auf Datenelemente oder Funktionen in der SIM. Das Gegenteil ist eine → <i>postpaid</i> SIM.
Proaktivität	Transaktions-Mechanismus für Chipkarten, der es der Chipkarte ermöglicht, selbstständig Aktionen im Terminal zu starten. Damit wird das starre Master-Slave-Prinzip bei der Chipkarten-Kommunikation umgangen. Realisiert wird die Proaktivität durch zyklisch stattfindende Anfragen des Terminals (d. h. <i>polling</i>) an die Chipkarte. Die Zykluszeit der Anfragen kann dabei vorab durch die Chipkarte eingestellt werden. Entstanden und hauptsächlich eingesetzt wird die Proaktivität vor allem bei SIMs, um nach GSM 11.14 einen Teil der Steuerung des Mobiltelefons zu übernehmen.
probabilistisch (<i>propabilistic</i>)	Bezeichnet ein Verfahren oder Algorithmus, der bei identischen Ausgangsbedingungen zu unterschiedlichen Ergebnissen kommt. Das Gegenteil ist → <i>deterministisch</i> .
Programmcode	Bezeichnung für ein von einem → <i>Interpreter</i> oder direkt vom → <i>Mikroprozessor</i> ausführbares Programm (→ <i>Nativecode</i>).
proprietär (<i>proprietary</i>)	Dieses Adjektiv wird in der Chipkartenwelt in häufig abwertender Form für firmenspezifische Lösungen benutzt, deren Spezifikationen nicht vollständig veröffentlicht oder die Eigentum einer einzelnen Firma sind. Das Gegenteil zu proprietär sind so genannte offene Lösungen, die auch von Dritten benutzt werden können. Sowohl die Verwendung des Begriffs proprietär als auch des Begriffs der offenen Lösung ist aber keinesfalls eindeutig. Objektiv gesehen ist manches so genannte offene Chipkarten-Betriebssystem auch beachtlich proprietär und abhängig von einer bestimmten Firma. Eine proprietäre Chipkarten-Anwendung (→ <i>Anwendung</i>) wäre beispielsweise ein elektronisches Geldbörsensystem für ein abgegrenztes Gebiet, das sich nicht an die einschlägigen Spezifikationen hält und von einer bestimmten Firma als Speziallösung entwickelt wurde.
Protection Profil (PP)	Im Rahmen einer → <i>Evaluierung</i> eine implementierungsunabhängige Menge von Sicherheitsvorgaben (→ <i>Security Target</i>), angepasst an spezifische Anwendungsgebiete für bestimmte → <i>Targets of Evaluation</i> .
Proton [Proton]	Markenname eines weltweit verbreiteten elektronischen Geldbörsensystems mit ca. 50 Millionen ausgegebenen Karten (Stand Frühjahr 2002). Die Spezifikationen von Proton definieren u. a. auch ein Multiapplication-Chipkarten-Betriebssystem (→ <i>Chipkarten-Betriebssystem</i>).
Prototyp	Ein (Software-)Prototyp ist ein ablauffähiges Modell des späteren Produkts mit eingeschränkter Funktionalität. Er wird zur Erprobung von bestimmten Eigenschaften des späteren Produkts benutzt. Beim horizontalen Prototyp werden nur

	bestimmte Ebenen der Software realisiert, wogegen beim vertikalen Prototyp bestimmte Teile durch alle Ebenen der Software hindurch realisiert werden.
prozedurale Programmierung	Die prozedurale Programmierung basiert darauf, dass Programme als Folgen von Anweisungen an einen → <i>Mikroprozessor</i> formuliert werden. Der Programmablauf kann zur Vereinfachung in Funktionen zerlegt werden und die notwendigen Daten werden in Variablen gehalten. Ein zentraler Aspekt der prozeduralen Programmierung ist, dass der Programmablauf und nicht wie bei der → <i>objektorientierten Programmierung</i> die zu verarbeitenden Daten im Mittelpunkt stehen. Typische prozedurale Programmiersprachen sind Basic und C.
Prozessmodell	anderer Begriff für → <i>Vorgehensmodell</i>
Prozessor	→ <i>Mikroprozessor</i>
Prozessorkarte	Kurzform von → <i>Mikroprozessorkarte</i> .
Pseudonymisierung	Veränderung von personenbezogenen Daten mittels einer Zuordnungsvorschrift in einer Weise, dass es nicht mehr möglich ist, diese veränderten Daten ohne Kenntnis der Zuordnungsvorschrift der ursprünglichen Person zuzuordnen. Der Begriff rührt daher, dass im einfachsten Fall der ursprüngliche Name durch ein einzigartiges Pseudonym ersetzt wird. In einer separaten Zuordnungstabelle (der Zuordnungsvorschrift) wird dann die Verbindung zwischen Pseudonym und ursprünglichem Namen hergestellt (→ <i>Anonymisierung</i>).
PSTN (<i>public switched mobile network</i>)	Bezeichnung für das reguläre öffentliche leitungsgebundene Telefonnetz.
Public-Key-Algorithmus	→ <i>Kryptoalgorithmus</i>
PUK (<i>personal unblocking key</i>)	Spezielle → <i>PIN</i> zum Rücksetzen des auf dem Maximalwert stehenden Fehlbildungszählers der PIN. Eine PUK hat üblicherweise eine größere Länge (z. B. 8stellig) als die PIN, da Benutzer die PUK nur im Fall einer vergessenen PIN kennen müssen und dann in ihren Unterlagen nachsehen können. Mit der erfolgreichen Benutzung einer PUK wird gleichzeitig eine neue PIN festgelegt, da die alte PIN dem Benutzer offensichtlich nicht mehr bekannt war.
Pull-Technologie	Informationsweitergabe durch das Abholen von Information von einem übergeordneten System (z. B. Server) durch ein untergeordnetes System (z. B. Mobiltelefon). Der Gegensatz zur Pull-Technologie ist die → <i>Push-Technologie</i> .
Purse-to-Purse-Transaktion (<i>purse to purse transaction</i>)	Transaktion von elektronischen Geldeinheiten von einer elektronischen Geldbörse direkt zu einer anderen, ohne den Umweg über ein drittes, übergeordnetes System. Im Regelfall bedeutet diese Funktionalität, dass das Börsensystem anonym arbeiten muss und die elektronischen Geldbörsen für diese Funktion einen gemeinsamen Schlüssel benutzen müssen.
Push-Technologie	Informationsweitergabe durch das Versenden von Information von einem übergeordneten System (z. B. Server) an ein untergeordnetes System (z. B. Mobiltelefon). Der Gegensatz zur Push-Technologie ist die → <i>Pull-Technologie</i> .
Quick	Markenname der 1995 in Österreich landesweit eingeführten elektronischen Geldbörse. Diese basiert in ihren wesentlichen Teilen auf der europäischen Norm für branchenübergreifende elektronische Geldbörsen EN 1546. ³⁸
Radicchio [Radicchio]	Weltweite Initiative von Firmen und Organisationen zur Entwicklung von sicheren mobilen → <i>E-Commerce</i> -Lösungen mit → <i>PKI</i> .
RAM (<i>random access memory</i>)	Flüchtige Speicherart, die in Chipkarten als Arbeitsspeicher Verwendung findet. Das RAM verliert seinen Inhalt bei Stromausfall. SRAM und DRAM sind RAM-Speicher mit besonderen technischen Eigenschaften. ³⁹

³⁸ Siehe auch Abschnitt 12.3.1 „CEN-Norm EN 1546“.

³⁹ Siehe auch Abschnitt 3.4.2 „Speicherarten“.

Rauschfreiheit	Eine Eigenschaft von → <i>Kryptoalgorithmen</i> . Diese benötigen bei Rauschfreiheit unabhängig von → <i>Schlüssel</i> , Klar- und Schlüsseltext für die Ver- und Entschlüsselung immer die gleiche Zeit. Ist ein Kryptoalgorithmus nicht rauschfrei, kann durch eine Analyse der Berechnungszeit der Schlüsselraum sehr stark eingeschränkt werden. Auf diese Weise kann der Schlüssel wesentlich schneller als bei einer erschöpfenden Schlüsselsuche gefunden werden.
Record	Auch Datensatz; eine bestimmte Anzahl von Daten ähnlich einem String.
Redlist	→ <i>Hotlist</i>
regelbasierte Programmierung	Diese Programmierung basiert darauf, dass allgemeine Regeln formuliert werden, die dann auf die zu lösenden Probleme angewandt werden. Anhand dieser Regeln kann dann das Problem vom Computer selbstständig gelöst werden. Ein zentraler Aspekt der regelbasierten Programmierung besteht darin, dass keine Programmabläufe wie bei der → <i>prozeduralen Programmierung</i> oder die zu verarbeitenden Daten wie bei der → <i>objektorientierten Programmierung</i> im Mittelpunkt stehen, sondern allgemeine Regeln. Typische regelbasierte Programmiersprachen sind Lisp und Prolog.
Registrierungsinstanz (<i>registration authority</i> – RA)	Nimmt bei einer → <i>PKI</i> die Zertifizierungsanträge von Antragstellern entgegen und leitet diese nach Prüfung der Echtheit der Antragsteller an die → <i>Zertifizierungsinstanz</i> weiter. Die Registrierungsinstanz ist somit die eindeutige Zuordnung von Zertifikaten zu Personen herbeiführt.
Remote Applet Management	Die Verwaltung (Erzeugen, Löschen, ...) von → <i>Applets</i> in einer Chipkarte von einem Hintergrundsystem aus. Beispielsweise ist es in verschiedenen → <i>GSM-Systemen</i> über die Luftschnittstelle möglich, Applets in eine SIM zu laden bzw. zu löschen.
Remote File Management	Die Verwaltung (Erzeugen, Löschen, Schreiben, Lesen, Änderung der Zugriffsbedingungen, ...) von Dateien in einer Chipkarte von einem Hintergrundsystem aus. Beispielsweise ist es in verschiedenen → <i>GSM-Systemen</i> möglich, über die Luftschnittstelle neue Dateien in einer SIM zu erzeugen und in diese Daten zu schreiben.
Reset	Zurücksetzen eines Computers (hier: einer Chipkarte) auf einen klar definierten Ausgangszustand. Man spricht von einem Kaltreset oder Power-On-Reset, wenn zur Ausführung des Resets die Versorgungsspannung ab- und wieder angeschlossen wird. Ein Warmreset wird durch ein Signal auf der Resetleitung zur Chipkarte ausgeführt, die Versorgungsspannung bleibt davon unberührt.
Retikel	→ <i>ROM-Maske</i>
roaming	Die Erreichbarkeit eines Mobiltelefons in einem anderem Netz als dem → <i>Heimatnetz</i> .
roll back	Funktionalität von Betriebssystemen, Daten im Fehler- oder Abbruchfall konsistent zu halten. Bei der roll back-Funktionalität werden die bei einer fehlerhaften oder abgebrochenen Operation verwendeten Daten durch die ursprünglichen Daten wieder ersetzt. Dieser Prozess kann automatisch oder nach Anforderung angestoßen werden und ist bei Chipkarten-Betriebssystemen oft durch → <i>atomare Abläufe</i> realisiert. Eine andere Strategie der Konsistenzhaltung von Daten im Fehler- oder Abbruchfall ist die → <i>roll forward</i> -Funktionalität.
roll forward	Funktionalität von Betriebssystemen, Daten im Fehler- oder Abbruchfall konsistent zu halten. Bei der roll forward-Funktionalität wird bei einer fehlerhaften oder abgebrochenen Operation mit den vorhandenen inkonsistenten Daten die Operation in einer Art wieder aufgenommen, dass die Daten anschließend wieder konsistent sind. Dieser Prozess kann automatisch oder nach Anforderung angestoßen werden, ist jedoch bei Chipkarten-Betriebssystemen aufgrund der hohen Sicherheitsanforderungen selten realisiert. Eine andere Strategie der Konsistenzhaltung von Daten im Fehler- oder Abbruchfall ist die → <i>roll back</i> -Funktionalität.

ROM (<i>read only memory</i>)	Nichtflüchtige Speicherart, die in Chipkarten Verwendung findet. Sie dient vornehmlich zur Speicherung von Programmen und statischen Daten, da sich der Inhalt eines ROM nicht verändern lässt. ⁴⁰
romed application	Diese Bezeichnung wird für Chipkarten-Anwendungen benutzt, die sich nicht im EEPROM befinden, sondern fest im maskenprogrammierten ROM des → <i>Chipkarten-Mikrocontrollers</i> .
ROM-Maske (<i>ROM-mask</i>)	Dieser Begriff wird umgangssprachlich sehr kontextspezifisch verwendet. Die ursprüngliche Bedeutung einer ROM-Maske ist die halbleitertechnische Belichtungsmaske für die Herstellung des ROM bei der Halbleiterfertigung. Die Bezeichnung Maske wird jedoch nur verwendet, wenn bei der Belichtung des → <i>Wafers</i> gegenüber der Maske keine Verkleinerung stattfindet. Werden hingegen die Strukturen bei der Abbildung auf den Wafer verkleinert, wird die „Maske“ als Retikel bezeichnet. Der Ausdruck „Maske“ bezeichnet jedoch auch den Dateninhalt des ROMs bei → <i>Chipkarten-Mikrocontrollern</i> und ist in manchen Fällen sogar das Synonym für das ganze → <i>Chipkarten-Betriebssystem</i> (→ <i>Softmaske, Hardmaske</i>).
Round-trip Engineering	Hier wird innerhalb des Software-Entwicklungsprozesses sequentiell an Design und Implementierung gearbeitet, wobei sich beide Tätigkeiten gegenseitig beeinflussen. Die Software-Architektur und der Programmcode werden durch ein Programm automatisch konsistent zueinander gehalten. Die Grundlage bildet dabei eine formale Modellierungssprache (z. B. UML), aus der dann durch automatische Programmcodegenerierung zumindest ein Rumpfcod des Programms erstellt wird. Die Erkenntnisse und Änderungen, die durch Verfeinerung und Test dieses Programmcodes gewonnen werden, fließen dann in einem Reverse Engineering Prozess wieder in die Modellierung des Programms zurück. Bei mehrmaligem Durchlaufen der Schleife Codegenerierung-Reverse Engineering erhält man in relativ kurzer Zeit praxisbezogene Software mit einer zum Sourcecode konsistenten Architektur. Angewandt wird das Round-trip Engineering fast ausschließlich bei objektorientierten Programmiersprachen (z. B. C++, Java) in Verbindung mit UML.
RSA (Ronald L. Rivest, Adi Shamir, Leonard Adleman)	Bekanntester und meistverbreiteter asymmetrische kryptografische Algorithmus (→ <i>Kryptoalgorithmus</i>). Er wurde von Ronald L. Rivest, Adi Shamir und Leonard Adleman im Jahr 1978 publiziert und ist nach den Anfangsbuchstaben der drei Familiennamen benannt. Das sehr einfache Funktionsprinzip basiert auf der Arithmetik großer Ganzzahlen. ⁴¹
R-UIIM (<i>removable user identity module</i>)	Übliche Bezeichnung für die GSM-spezifische Chipkarte. Sie ist ein optionales Sicherheitsmodul, das sich austauschbar in mobilen Endgeräten des Mobilfunksystems CDMA 2000 befindet. Die R-UIIM hat eine ähnliche Funktionalität wie die → <i>SIM</i> , jedoch mit dem Kryptoalgorithmus CAVE (<i>cellular authentication, voice privacy and encryption</i>), der für eine Vielzahl von kryptografisch gesicherten Funktionen in der R-UIIM eingesetzt wird. Angelehnt an das SIM Application Toolkit, ist für die R-UIIM auch ein UIM Application Toolkit (UATK) spezifiziert.
salt	Zufallssequenz, die zur Verlängerung von Passwörtern verwendet wird, um Wörterbuchangriffe auf die gespeicherten Passwörter zu erschweren.
SAM (<i>secure application module</i>)	→ <i>Sicherheitsmodul</i>
Sammelbeauftragter	→ <i>Acquirer</i>
Sandbox (<i>sandbox</i>)	→ <i>Virtual Machine</i>

⁴⁰ Siehe auch Abschnitt 3.4.2 „Speicherarten“.

⁴¹ Siehe auch Abschnitt 4.7.2 „Asymmetrische Kryptoalgorithmen“.

Schlechtfall (<i>bad case</i>)	Der Fall bei einer logischen Entscheidung, der zum ungünstigeren oder ungewollten Ergebnis führt.
Schlüssel (<i>key</i>)	Bezeichnet bei einem → <i>Kryptoalgorithmus</i> denjenigen Parameter, der die Ver- oder Entschlüsselung individualisiert. Schlüssel müssen im Falle von symmetrischen Kryptoalgorithmen zur Gewährleistung der Sicherheit ein Geheimnis sein oder dürfen bei asymmetrischen Kryptoalgorithmen im Falle eines öffentlichen Schlüssels auch bekannt sein.
Schlüsselmanagement (<i>key management</i>)	Alle Verwaltungsfunktionen für die Erzeugung, Verteilung, Speicherung, Aktualisierung, Vernichtung und Adressierung von kryptografischen Schlüsseln.
SCOPE (<i>smart card open platform environment</i>)	Die Spezifikation einer Art → <i>HAL (hardware abstraction layer)</i> von → <i>Global Platform</i> .
Scrambling	Vermischte Anordnung der Busse (Adress-, Daten- und Steuerbus) auf dem Chip eines Mikrocontrollers, so dass eine Zuordnung nach Funktionen ohne Hintergrundinformationen nicht mehr möglich ist. Statisches Scrambling bedeutet, dass die Busse einer Serie von Mikrocontrollern identisch gescrambelt sind. Beim dynamischen Scrambling sind die Busse chipindividuell oder sogar sitzungsindividuell gescrambelt. ⁴²
Scratch-Karte (<i>scratch card</i>)	Karte im üblichen → <i>ID-1-Format</i> , jedoch mit geringerer Kartendicke, die mit einer Geheimzahl oder einem Passwort unter einer abkratzbaren und undurchsichtigen Deckschicht bedruckt ist. Diese Schicht wirkt als Siegel, so dass die Unversehrtheit vor Benutzung visuell geprüft werden kann. Eine Scratch-Karte verfügt somit über eine ähnliche Funktionalität wie ein PIN-Brief. Scratch-Karten werden oft als Voucher zur Verteilung von Einmalpasswörtern zum Wiederaufladen von → <i>prepaid SIMs</i> benutzt.
SDMA (<i>space division multiple access</i> – Raumvielfachzugriff)	Vielfachzugriffsverfahren zur parallelen Datenübertragung von mehreren Sendern zu einem Empfänger auf einer Frequenz. Dazu verwenden die Sender richtungsselektive Antennen, die auf den jeweiligen Empfänger ausgerichtet werden. Dies lässt sich aufgrund des hohen Aufwandes im Mobilfunkbereich nur bei den Basisstationen, beispielsweise mit Antennenarrays (adaptive Antennen), realisieren. ⁴³
SECCOS (<i>security card operating system</i>)	Name des Multiapplication-Chipkarten-Betriebssystems (→ <i>Chipkarten-Betriebssystem</i>) für die deutsche ec-Karte mit Chip bzw. → <i>Geldkarte</i> .
Secret-Key-Algorithmus	→ <i>Kryptoalgorithmus</i>
Secure Messaging	Alle Mechanismen, Protokolle und Kryptoalgorithmen, um die Datenübertragung von und zu einer → <i>Chipkarte</i> gegen Manipulationen oder Abhören zu schützen. ⁴⁴
Security Environment (SE, Sicherheitsumgebung)	Bezeichnet bei einer Chipkarte einen logischer Container, der eine Menge von vollständig definierten Sicherheitsmechanismen enthält, die von sicherheitsrelevanten → <i>Kommandos</i> und → <i>Secure Messaging</i> benutzt werden. Security Environments eignen sich beispielsweise sehr gut für die sicherheitstechnische Absicherung der unterschiedlichen Lebenszyklen (siehe auch Lebenszyklus) einer Chipkarte. Im einfachsten Fall würde man in diesem Beispiel für Personalisierung und späteren Betrieb unterschiedliche Security Environments definieren, sodass die → <i>Zugriffsbedingungen</i> auf die Dateien je nach Lebensabschnitt der Chipkarte unterschiedlich festgelegt sind. Im Falle der Personalisierung dürfte man dann auf alle Dateien schreibend zugreifen, und im Falle des Betriebs wären die Zugriffsbedingungen entsprechend der eigentlichen → <i>Anwendung</i> gesetzt.

⁴² Siehe auch Abschnitt 8.2.4.1 „Angriffe auf der physikalischen Ebene“.

⁴³ Siehe auch Abschnitt 13.1.1 „Vielfachzugriffsverfahren“.

⁴⁴ Siehe auch Abschnitt 6.6 „Sicherung der Datenübertragung“.

Security Target (Sicherheitsvorgabe)	Die Sicherheitsvorgaben beschreiben im Rahmen einer → <i>Evaluierung</i> die zu prüfenden Mechanismen beim → <i>Target of Evaluation</i> . Sie sind damit eine Art Anforderungskatalog für die Evaluierung. Die Sicherheitsvorgaben für bestimmte Arten und Anwendungsgebiete von Targets of Evaluation können in → <i>Protection Profiles</i> beschrieben sein.
seed number (Initialzahl)	Auch <i>seed</i> ; der aus einer Zufallszahl bestehende Startwert für Pseudozufallszahlengeneratoren.
Seitenorientierung	Zusammenfassung von mehreren Bytes in einem Speicher, die nur als Ganzes geschrieben oder gelöscht werden können. Eine Seitenorientierung gibt es bei → <i>Chipkarten-Mikrocontrollern</i> nur für Speicher des Typs EEPROM oder Flash-EEPROM. Typische Größen von Speicherseiten sind 4 Byte, 32 Byte, 64 Byte und 128 Byte. Allerdings gibt es mittlerweile auch Mikrocontroller, die keine fixe Seitenorientierung mehr haben, sondern eine Variable in einem bestimmten Bereich, z. B. 1 Byte bis 128 Byte.
serielle Datenübertragung	Hier werden die einzelnen Datenbits nacheinander auf einer Datenleitung übertragen (→ <i>parallele Datenübertragung</i>).
SET (<i>secure electronic transaction standard</i>)	Zahlungsverkehrsprotokoll zur Abwicklung von sicheren Kreditkartenzahlungen im Internet. Es wurde von Visa und Mastercard 1996 veröffentlicht. SET verlangt beim Bezahler nicht zwangsläufig eine Chipkarte, sondern kann dort vollständig in Software auf dem PC realisiert sein. Eine Erweiterung von SET namens C-SET (Chip-SET) ist bislang nur in Frankreich von Relevanz und nicht international standardisiert.
shall, should, may	In internationalen Normen kommen diese drei englischsprachigen Hilfsverben normalerweise in vielen Bereichen vor. Ihre Bedeutung ist genau geregelt und weicht daher teilweise vom verbreiteten laxen umgangssprachlichen Gebrauch dieser Wörter ab. „Shall“ bedeutet, dass der bezeichnete Sachverhalt entsprechend der Beschreibung verpflichtend realisiert werden muss. „Should“ hingegen entspricht nur im Ansatz einer Empfehlung, da das Beschriebene wenn irgendwie möglich erfüllt werden muss. Eine echte Entscheidungsmöglichkeit bei der Realisierung ist nur bei „may“ gegeben.
Short-FID (<i>SFI</i>)	5 Bit langes Kennzeichen für EFs und hat den Wertebereich 1 bis 31. Es wird zur impliziten Selektion eines EFs innerhalb eines Schreib- oder Lesekommandos (z. B. READ BINARY) an die Chipkarte benutzt. ⁴⁵
Shrink	Bezeichnet die Flächenreduzierung eines Halbleiterchips durch Verwendung einer Halbleitertechnologie mit kleinerer Strukturbreite. Auf Grund des reduzierten Flächenbedarfs können pro Wafer mehr Halbleiterchips untergebracht werden. Dies reduziert wiederum die Kosten der Halbleiterchips, da der Chip-Preis annähernd proportional zum Platzbedarf auf dem Wafer ist.
Shutter	Mechanische Vorrichtung in Terminals, die gegebenenfalls alle von der Chipkarte aus dem Terminal führenden Drähte abschneidet. Damit soll eine Manipulation der Kommunikation verhindert werden. Falls ein Abscheiden nicht möglich ist, wird die gesteckte Chipkarte elektrisch nicht aktiviert.
Sicherheitsmodul	Ein sowohl mechanisch als auch informationstechnisch abgesichertes Bauteil, das zur Aufbewahrung von geheimen Daten und zur Ausführung von Kryptoalgorithmen dient. Andere Namen für Sicherheitsmodul sind SAM (<i>secure application module</i>) und HSM (<i>hardware security module</i> oder <i>host security module</i>).
Signalburst	Oft auch kurz Burst genannt; ein zusammenhängendes Datenpaket, das über die Luftschnittstelle zwischen Feststation und Mobilstation übertragen wird.

⁴⁵ Siehe auch Abschnitt 5.6.2 „Dateinamen“.

Signaturgesetz (SigG)	Im Allgemeinen ein Gesetz, das den Gebrauch und Einsatz von → <i>digitalen Signaturen</i> regelt. In Deutschland versteht man darunter das „Gesetz über Rahmenbedingungen für elektronische Signaturen“ vom 22. Mai 2001. Darin ist der Rahmen für den Einsatz von digitalen Signaturen in Deutschland vorgegeben, ⁴⁶ welche in der → <i>Signaturverordnung</i> konkretisiert werden.
Signaturkarte (signature card)	→ <i>Chipkarte</i> mit → <i>Mikrocontroller</i> , deren Hauptaufgabe die sichere Aufbewahrung und Nutzung von geheimen Schlüsseln für → <i>digitale Signaturen</i> ist.
Signaturverordnung (SigV)	In der deutschen Signaturverordnung vom 8. Oktober 1997 werden die Rahmenbedingungen, die vom → <i>Signaturgesetz</i> vorgegeben sind, so weit konkretisiert, dass darauf aufbauend konkrete Maßnahmenkataloge als Vorschläge zur praktischen Anwendung von digitalen Signaturen erstellt werden können. Die Signaturverordnung beschreibt beispielsweise die notwendigen Verfahren für die Erzeugung von Signaturschlüsseln und Identifikationsdaten sowie notwendige Sicherheitskonzepte und die notwendigen Prüfstufen nach ITSEC für die Signaturkomponenten. ⁴⁷
SIM (subscriber identity module)	Übliche Bezeichnung für die GSM-spezifische Chipkarte. ⁴⁸ Sie ist ein obligatorisches Sicherheitsmodul, das sich austauschbar in mobilen Endgeräten des GSM-Mobilfunksystems befindet. Sie kann die herkömmliche Kreditkartengröße ID-1 haben oder auch als kleine Plug-In-Karte in ID-000 ausgeführt sein. Das SIM ist der Träger der Identität des Teilnehmers und hat als Hauptaufgabe die Sicherstellung der Echtheit der Mobilstation gegenüber dem Netzwerk. Zusätzliche Aufgaben sind die manipulationssichere Ausführung von Programmen (Authentisierung), die Benutzeridentifizierung (mittels PIN) und die Speicherung von Daten, wie beispielsweise Telefonnummern. Das Äquivalent des SIM bei UMTS ist das → <i>USIM</i> . ⁴⁹
SIM Alliance [SIM Alliance]	Ein im Jahr 1999 von Gemplus, G + D, ORGA und Schlumberger gegründetes Konsortium, um Dienstleistungen, die für WAP entwickelt wurden, auch auf nicht WAP-fähigen Mobiltelefonen zu ermöglichen. Dazu muss die SIM über einen SIM Alliance-fähigen Browser verfügen und das Mobiltelefon die Phase 2+ von GSM unterstützen. Damit ist die → <i>SIM</i> fähig, das Mobiltelefon über → <i>SIM Application Toolkit</i> so weit zu steuern, dass darauf ein Großteil von WAP-Inhalten und deren Funktionalität abgebildet werden kann. ⁵⁰
SIM Application Toolkit (SAT, unüblich und veraltet STK)	Baukastenähnliche Erweiterung der GSM 11.11-Spezifikation (genormt in GSM 11.14), die es der SIM-Karte erlaubt, eine aktive Rolle bei der Steuerung des Mobiltelefons zu übernehmen. Mit dem SIM Application Toolkit ist es beispielsweise einer SIM möglich, Anzeigen auf dem Display auszugeben, Eingaben über die Tastatur anzufordern und Nachrichten über die Luftschnittstelle zu senden und zu empfangen. Das SIM Application Toolkit ist die Grundlage für die meisten Zusatzanwendungen bei Mobiltelefonen. Das Äquivalent des SIM Application Toolkit bei UMTS ist das → <i>USIM Application Toolkit</i> (USAT), ⁵¹ und bei der → <i>R-UM</i> ist es das UATK (UIM Application Toolkit). Die von der Expertengruppe → <i>EP SCP</i> definierte zukünftige generische Grundlage für alle Application Toolkits von Chipkarten im Mobilfunk wird das CAT (<i>card application toolkit</i>) sein.

⁴⁶ Siehe auch Abschnitt 14.4 „Digitale Signatur“.

⁴⁷ Siehe auch Abschnitt 14.4 „Digitale Signatur“.

⁴⁸ Siehe auch Abschnitt 13.2 „Das GSM-System“.

⁴⁹ Siehe auch Abschnitt 13.3 „Das UMTS-System“.

⁵⁰ Siehe auch Abschnitt 13.5 „Die WIM“.

⁵¹ Siehe auch Abschnitt 13.3 „Das UMTS-System“.

SIM Lock (SIM Sperre)	Verfahren, das Mobiltelefone fest mit einer bestimmten → <i>Chipkarte</i> , d. h. einem → <i>SIM</i> , verbindet. Dazu liest entweder das Mobiltelefon aus der SIM bestimmte Daten aus und vergleicht diese mit im Mobiltelefon gespeicherten Daten oder das SIM liest eindeutige Daten aus dem Mobiltelefon und vergleicht diese mit gespeicherten Daten. Sind die Daten gleich, dann kann das Mobiltelefon benutzt werden. Es ist in der Regel möglich, entweder über die Luftschnittstelle oder durch Eingabe eines geheimen Schlüssels am Mobiltelefon, die SIM-Lock-Funktion abzuschalten, so dass anschließend auch andere Chipkarten verwendet werden können. Die SIM-Lock-Funktion wird benutzt, um vom Netzbetreiber subventionierte Mobiltelefone für eine gewisse Zeit an eine bestimmte Chipkarte und deren Bezahlmodalität (→ <i>prepaid</i>) zu binden. ⁵²
SIM Toolkit	Kurzform von → <i>SIM Application Toolkit</i> .
SIMEG (<i>subscriber identity module expert group</i>)	Die SIMEG war eine 1988 gegründete Expertengruppe, die im Rahmen von → <i>ETSI</i> die Spezifikation für die Schnittstelle zwischen Chipkarte und Mobiltelefon festgelegt hat (GSM 11.11). Der Name SIMEG wurde 1994 in → <i>SMG9</i> geändert.
Simulator (<i>simulator</i>)	Software, die die Funktionsweise eines Geräts (des Zielsystems) nachahmt. Die Nachahmung mittels Hardware bezeichnet man hingegen als → <i>Emulator</i> . Simulatoren werden oft zur Entwicklung von Software für noch nicht existierende Zielsysteme eingesetzt. Ein Chipkarten-Simulator ist beispielsweise eine Software, die eine echte Chipkarte auf logischer Ebene vollständig nachbildet. Simulatoren sind in der Regel langsamer als Emulatoren, d. h. sie simulieren oft nicht in Echtzeit.
Single-Sign-On (SSO, Einfachanmeldung)	Hierbei werden mehrere unterschiedliche Benutzeridentifizierungen für verschiedene Anwendungen durch eine einzelne zentrale Benutzeridentifizierung ersetzt. Realisiert wird dies durch eine Software, die aufgrund der korrekten zentralen Benutzeridentifizierung die entsprechenden Identifizierungsdaten (→ <i>Identifizierung</i>) und Passwörter an die zugehörigen Identifizierungsinstanzen sendet. Damit entfällt für den Benutzer die Notwendigkeit, sich viele verschiedene Passwörter zu merken.
Sitzung (<i>session</i>)	Zeitspanne zwischen An- und Abschaltsequenz einer Chipkarte, in der sowohl der gesamte Datenaustausch als auch die dazu notwendigen informationstechnischen Mechanismen ablaufen.
skimming	Typischer Angriff bei Magnetstreifenkarten. Dazu werden die Daten des Magnetstreifens von einer fremden Magnetstreifenkarte unerlaubterweise gelesen und auf den Magnetstreifen einer Blankokarte kopiert. Diese Karte kann nun bezüglich des Magnetstreifens wie die Originalkarte verwendet werden.
Skript (<i>script</i>)	Interpretiertes Programm hauptsächlich zur Realisierung von einfachen kurzen Anwendungen und zur Automatisierung von immer wiederkehrenden Abläufen.
Smart Card	Anderer Ausdruck für Mikroprozessorkarte. Er steht für eine Chipkarte, die „smart“, also schlau ist, weshalb Speicherkarten nicht mehr unter diesen Überbegriff fallen.
Smart Label	Datenträger in dünner Bauform, mit denen mittels kontaktloser Datenübertragung kommuniziert werden kann. Dabei können in der einfachsten Variante (oft auch ohne Chip) Daten lediglich vom Smart Label gelesen werden. Höher entwickelte Varianten von Smart Labels ermöglichen es, auch Daten zu schreiben bzw. Daten analog einer → <i>Chipkarte</i> im Smart Label zu verarbeiten.
Smart Object	→ <i>Chipkarten-Mikrocontroller</i> , verpackt in einer zur üblichen Kartenform alternativen Bauform. Beispiele für Smart Objects sind USB-Stecker oder Ringe, die mit einem Chipkarten-Mikrocontroller ausgerüstet sind.

⁵² Siehe auch Abschnitt 13.2 „Das GSM-System“.

Smartcard [Groupmark]	Dieser Begriff ist ein eingetragenes Warenzeichen der kanadischen Firma Groupmark.
SMG9 (<i>special mobile group 9</i>)	Die SMG9 war eine Expertengruppe, die im Rahmen von → ETSI die Spezifikation für die Schnittstelle zwischen Chipkarte und Mobiltelefon festgelegt hat (z. B.: GSM 11.11, GSM 11.14, ...). Sie setzte sich aus Vertretern von Karten-, Mobiltelefonherstellern und Netzbetreibern zusammen. Der frühere Name der SMG9 war → SIMEG. Im Jahr 2000 wurde die SMG9 aufgelöst und die Aufgaben auf zwei neue Expertengruppen verteilt. Für die applikationsspezifische Schnittstelle Mobiltelefon zu → SIM bzw. → USIM ist die 3GPP Expertengruppe T3 verantwortlich und alle generischen Themen im Bereich von Chipkarten im Bereich der Telekommunikation werden von der ETSI-Expertengruppe EP SCP (<i>ETSI project smart card platform</i>) bearbeitet.
SMS (<i>short message service</i> , Kurzmitteilungsdienst)	Kurzmitteilungsdienst von GSM mit einer maximalen Nachrichtenlänge von 160 alphanumerischen Zeichen. SMS-Nachrichten werden über den Signalisierungskanal und nicht über den Datenkanal übertragen, d. h. sie können auch während eines aktiven Telefonats gesendet und empfangen werden. Der SMS-Dienst wird nicht nur zur Übermittlung von Kurznachrichten für die Teilnehmer verwendet, sondern auch als → Bearer-Dienst zur Übertragung von Daten an das Mobiltelefon beispielsweise bei → WAP oder an die SIM (→ OTA).
Softmaske (<i>soft mask</i>)	Dieser Begriff bedeutet, dass sich, aufbauend auf einem → Chipkarten-Betriebssystem, im ROM ein Teil des Programmcodes im EEPROM befindet (→ ROM-Maske). Programme im EEPROM lassen sich durch Überschreiben leicht ändern, sind also „soft“. Der Ausdruck „Maske“ ist in diesem Zusammenhang eigentlich falsch, da man für ein Programm im EEPROM keine halbleitertechnische Belichtungsmaske erstellen muss. Softmasken werden üblicherweise nicht für große Stückzahlen (z. B. für Feldversuche) bei → rapid prototyping oder für Erweiterungen verwendet. Das Gegenteil einer Softmaske ist die → Hardmaske, bei der die wesentlichen Funktionen Teile des ROMs sind.
SPA (<i>simple power analysis</i>)	Die einfache Leistungsanalyse ist eine Angriffsmethode auf Chipkarten, bei der mit hoher zeitlicher Auflösung der Stromverbrauch eines Mikrocontrollers gemessen wird. Aufgrund des Stromverbrauchs kann auf die internen Abläufe und verarbeiteten Daten des Mikrocontrollers geschlossen werden. Bekannt wurde die SPA durch eine Veröffentlichung von Paul Kocher, Joshua Jaffe und Benjamin Jun im Juni 1998 [Kocher 98] (→ DPA). ⁵³
SPA/DPA-resistent	Eigenschaft eines Kryptoalgorithmus, dass der benutzte geheime Schlüssel nicht durch eine → SPA- bzw. → DPA-Analyse herausgefunden werden kann.
Speicherkarte (<i>memory card</i>)	Karte mit Chip, der eine einfache Logikschaltung mit zusätzlichem schreib- und lesbaren Speicher besitzt. Speicherkarten können zusätzlich noch Sicherheitsbaugruppen aufweisen, welche beispielsweise eine Authentisierung ermöglichen.
Sperrliste (<i>blacklist</i>)	Liste in einer Datenbank, auf der alle Chipkarten oder Geräte vermerkt sind, die in einer bestimmten → Anwendung nicht mehr verwendet werden dürfen (→ Hotlist, Greylist, Whitelist).
Spezifikation (<i>specification</i>)	Eindeutige, vollständige und redundanzfreie Beschreibung einer Software. Ihr Inhalt darf keine auslegungsfähigen Teile enthalten und muss von allen Lesergruppen mit unterschiedlicher Funktion (Entwickler, Tester, Qualitätssicherer, ...) in vertretbarer Zeit verstanden werden können.
SRAM (<i>static random access memory</i>)	RAM-Speicher in statischer Bauweise, der zum Erhalt des Speicherinhalts lediglich eine konstante Stromversorgung und keine zyklische Wiederauffrischung des Inhalts benötigt. Die Zugriffszeit auf SRAM-Speicher ist geringer als auf

⁵³ Siehe auch Abschnitt 8.2.4.1 „Angriffe auf der physikalischen Ebene“.

	DRAM-Speicher, allerdings benötigen SRAMs mehr Platz auf dem Chip und sind deshalb auch teurer. ⁵⁴
Stack	Datenstruktur, in der die zuletzt abgelegten Objekte als Erste wieder entfernt werden können (<i>last in first out</i> – LIFO). Der wohl bekannteste Stack ist der Programmstack, auf dem beim Aufruf von Unterprogrammen die Rücksprungadressen abgelegt werden.
Standard	Damit werden in diesem Buch und in der Regel alle normungsähnlichen Dokumente bezeichnet, die beispielsweise von Firmen oder im industriellen Umfeld publiziert werden und nicht von einem nationalen oder internationalen Normungsgremium erstellt bzw. veröffentlicht worden sind (→ <i>Norm</i>). Verwirrenderweise werden in Deutschland die beiden Begriffe Norm und Standard oft gleichwertig verwendet, was streng genommen nicht korrekt ist.
STARCOS	Markenname eines Multiapplication-Chipkarten-Betriebssystems (→ <i>Chipkarten-Betriebssystem</i>) von Giesecke und Devrient [GD], das seit 1991 in verschiedenen Version erhältlich ist.
Steganografie	Zweck der Steganografie ist es, Nachrichten in anderen Nachrichten so zu verbergen, dass sie von einem unbedarften Beobachter (Mensch oder Maschine) nicht mehr erkannt werden können. Beispielsweise könnte ein Text codiert und in einer Bilddatei versteckt werden, so dass das betreffende Bild sich nur geringfügig ändert, und deshalb die Bildmodifikation praktisch nicht mehr wahrnehmbar ist (→ <i>digitales Wasserzeichen</i>). Mit einem entsprechenden Analyseprogramm könnte man jedoch den in der Bilddatei versteckten Text zu einem späteren Zeitpunkt wiederherstellen und so beispielsweise einen Copyright-Vermerk wieder zum Vorschein bringen.
Super Smart Card	Chipkarte mit integrierten aufwändigen Kartenelementen wie Display und Tastatur. Seit einigen Jahren hat sich jedoch anstelle von Super Smart Card der Begriff → <i>System-on-Card</i> eingebürgert.
symmetrischer Kryptoalgorithmus	→ <i>Kryptoalgorithmus</i>
synchrone Datenübertragung	Hier werden die Daten abhängig von einem fest vorgegebenen Zeitraster übertragen. Das Zeitraster kann beispielsweise von der an dem Chip angelegten Taktfrequenz abgeleitet sein (→ <i>asynchrone Datenübertragung</i>).
System-on-Card	Bezeichnet in der Chipkartenwelt eine um zusätzliche Kartenelemente neben dem Chipmodul erweiterte Chipkarte. Die üblicherweise aufgeführten Kartenelemente sind Display, Stromversorgung (Batterie, Solarzellen), Tastatur, Antenne, Sensoren für biometrische Benutzeridentifizierung (z. B. Fingerabdruck) und Lautsprecher. Die Ansteuerung dieser Kartenelemente kann vom Chip im Modul aus geschehen, muss aber nicht. Ein anderer, mittlerweile jedoch seltener benutzter Begriff für System-on-Card ist Super-Smart-Card.
T = 0	Dieses Übertragungsprotokoll regelt die Datenübertragung zwischen Terminal und Chipkarte und war das erste international genormte Übertragungsprotokoll für Chipkarten. Es ist ein byteorientiertes Halbduplex-Protokoll, das asynchron betrieben wird, und ist für minimalen Speicherbedarf und maximale Einfachheit ausgelegt. Dieses Protokoll wird weltweit in der GSM-Karte verwendet und hat deshalb den größten Verbreitungsgrad aller Chipkarten-Protokolle. Genormt ist T = 0 in ISO/ IEC 7816-3, dazu kompatible Spezifikationen sind in der GSM 11.11, TS 102.221 und dem EMV-Standard enthalten.
T = 1	Dieses Übertragungsprotokoll regelt die Datenübertragung zwischen Terminal und Chipkarte. Es ist ein blockorientiertes Halbduplex-Protokoll, das asynchron

⁵⁴ Siehe auch Abschnitt 3.4.2 „Speicherarten“.

	betrieben wird, und bietet eine Trennung zwischen Datenübertragung und → <i>Anwendung</i> . Genormt ist T = 1 in ISO/ IEC 7816-3, dazu compatible Spezifikationen sind in TS 102.221 und dem EMV-Standard enthalten.
T3	→ <i>SMG9</i>
Tag (engl.)	Kennzeichnung für Datenobjekte, die unter anderem vor allem bei ASN.1-Codierungen zum Einsatz kommt. ⁵⁵
tape out	Zeitpunkt, an dem das Chipdesign fertiggestellt ist und die dabei erzeugten Daten an die Maskenherstellung (→ <i>ROM-Maske</i>) überführt werden. Dies ist ein wichtiger Meilenstein bei der Chipherstellung. Der Begriff rührt daher, dass früher die Maskendaten auf Magnetbändern (tape) übergeben (out) wurden.
Target of Evaluation (TOE)	Das zu evaluierende informationstechnische System (→ <i>Evaluierung</i>), d. h. das zu prüfende Objekt. Ein TOE kann beispielsweise eine Mikrocontroller-Chipkarte (→ <i>Chipkarten-Mikrocontroller</i>) mit integrierter Software sein, die bestimmte Sicherheitsvorgaben, die → <i>Security Targets</i> , erfüllen muss.
TCSEC (<i>trusted computer system evaluation criteria</i>)	Kriterienkatalog zur Entwicklung und → <i>Evaluierung</i> der Sicherheit von informationstechnischen Systemen im US-amerikanischen Bereich und wurden 1983 vom National Computer Security Center → <i>NCSC</i> veröffentlicht. Der Nachfolger der nationalen TCSEC sind die international gültigen Common Criteria (→ <i>CC</i>).
TD/CDMA (<i>time division/code division multiple access</i>)	→ <i>CDMA</i>
TDES	anderer Ausdruck für → <i>Triple-DES</i>
TDMA (<i>time division multiple access</i> – Zeitvielfachzugriff)	Vielfachzugriffsverfahren zur quasi parallelen Datenübertragung von mehreren Sendern zu einem Empfänger auf einer Frequenz. Dazu erhält jeder Sender einen bestimmten Zeitschlitz, in dem er exklusiv senden darf, was eine sehr genaue Synchronisation erfordert. TDMA wird in Verbindung mit FDMA bei GSM auf der Luftschnittstelle zwischen Mobiltelefon und Basisstation benutzt. ⁵⁶
Teiler	In der Chipkartenwelt die gebräuchliche Kurzform von <i>clock rate conversion factor</i> (CRCF). Der CRCF gibt die Dauer eines Bits bei der Datenübertragung in der Anzahl der Takte auf der Clock-Leitung an.
Terminal (<i>terminal</i>)	Das Gegenstück zur Chipkarte ist das (Chipkarten-)Terminal. Es ist ein Gerät, z. T. mit Tastatur und Display, das die elektrische Versorgung und den Datenaustausch mit der Chipkarte ermöglicht. Nach ISO ist die normgerechte Bezeichnung für ein Chipkarten-Terminal IFD (<i>interface device</i>) und im Zahlungsverkehr ist die übliche Bezeichnung für ein Terminal CAD (<i>card accepting device</i>).
Test (<i>test</i>)	Hier wird ein bereits gedebugtes Programm in systematischer Vorgehensweise auf seine Funktionsfähigkeit und die Erfüllung der bei der → <i>Analyse</i> ermittelten Anforderungen geprüft. Das vorrangige Ziel ist nicht die Suche nach Fehlern im Programm, sondern die Prüfung der geforderten Funktionen. Das Testing ist deshalb nicht identisch mit dem → <i>Debugging</i> .
TETRA (<i>terrestrial trunked radio</i> , früher: <i>trans european trunked radio</i>)	Spezifikation für ein digitales Bündelfunksystem im 380 MHz bis 420 MHz Bereich mit → <i>TDMA</i> -Vielfachzugriff und wird von → <i>ETSI</i> herausgegeben. Zur Teilnehmeridentifikation ist bei TETRA analog → <i>GSM</i> eine → <i>SIM</i> vorgesehen, die üblicherweise TETRA-SIM genannt wird. Die TETRA-SIM ist jedoch optional, kann also auch als Software in der Mobilstation realisiert werden.
TETRA-SIM	→ <i>TETRA</i>
Thread	→ <i>Multithreading</i>

⁵⁵ Siehe auch Abschnitt 4.1 „Strukturierung von Daten“.

⁵⁶ Siehe auch Abschnitt 13.1.1 „Vielfachzugriffsverfahren“.

TLV-Format	Umgangssprachlicher Ausdruck für → <i>BER</i> -codierte Datenobjekte nach → <i>ASN.1</i> , bei dem ein Datum (<i>value</i>) durch ein vorangestelltes Kennzeichen (<i>tag</i>) und die Länge (<i>length</i>) eindeutig beschrieben wird.
TPDU	→ <i>APDU</i>
Transaktion (<i>transaction</i>)	Aufeinanderfolgender Ablauf von mehreren zusammengehörigen → <i>Kommandos</i> zu einer Chipkarte, um eine bestimmte Aufgabe zu erledigen. Ein typisches Beispiel für eine Transaktion sind die Kommandos für das Laden einer elektronischen Geldbörse.
Transaktionsnummer (<i>transaction number</i>)	Eine TAN ist im Gegensatz zu einer PIN nur für eine einzige Transaktion gültig und kann deshalb nur einmal verwendet werden. Üblicherweise erhält man mehrere TANs in Form einer Tabelle mit fünfstelligen Zahlen, ausgedruckt auf Papier, die dann exakt in der vorgegebenen Reihenfolge für die einzelnen Transaktionen bzw. Sitzungen benutzt werden müssen.
Transferkarte (<i>transfer card</i>)	→ <i>Chipkarte</i> , die als Transportmedium zwischen zwei Instanzen genutzt wird. Dazu besitzt sie einen großen Datenspeicher und in der Regel Schlüssel für eine Authentisierung, die prüft, ob die zu transferierenden Daten von der jeweiligen Stelle gelesen bzw. geschrieben werden dürfen.
Transistenz (<i>transient</i>)	Fähigkeit eines Objekts, nur während der Ausführungszeit eines Prozesses zu existieren; das Gegenteil ist die → <i>Persistenz</i> .
Transportprotokoll	Unüblicher Ausdruck für → <i>Übertragungsprotokoll</i> .
trapdoor (Falltür)	Eine in eine Software bewusst eingebaute Möglichkeit, unter Umgehung von Sicherungsmaßnahmen auf Daten und Programme eines Systems zuzugreifen.
Triple-Band-Mobiltelefon	Mobiltelefon, das in drei Frequenzbändern (z. B. GSM 900, GSM 1800, und GSM 1900) arbeiten kann.
Triple-DES	auch TDES und 3 DES genannt; eine modifizierte DES-Verschlüsselung durch aufeinander folgenden dreifachen Aufruf des DES-Algorithmus mit abwechselnder Ver- und Entschlüsselung. Wird für die drei DES-Aufrufe der gleiche Schlüssel verwendet, entspricht die Triple-DES-Verschlüsselung einer normalen DES-Verschlüsselung. Werden hingegen zwei bzw. drei unterschiedliche Schlüssel verwendet, dann stärkt dies die DES-Verschlüsselung erheblich gegenüber einer einfachen DES-Verschlüsselung. ⁵⁷
Trivial-PIN	Leicht zu erratende → <i>PIN</i> , wie beispielsweise "1234" (→ <i>0-PIN</i>).
trojanisches Pferd	Historisch gesehen das Holzpferd, mit dem es Odysseus gelang, sich Zugang zur stark befestigten Stadt Troja zu verschaffen. In der modernen Fassung ein Programm, das vordergründig eine definierte Aufgabe erfüllt, doch zusätzliche und unbekannte Funktionen ausführen kann. Es wird bewusst in ein Computersystem oder Wirtsprogramm eingebracht und kann sich im Gegensatz zu Viren nicht vermehren.
Trustcenter (<i>trust center</i> – TC)	Neben der → <i>Signaturkarte</i> das wesentliche Element bei einer → <i>PKI</i> . Es ist die Instanz, die Zertifikate erzeugt, verteilt und verwaltet. Sie übernimmt damit je nach Ausprägung die Aufgaben einer → <i>Zertifizierungsinstanz</i> , einer → <i>Registrierungsinstanz</i> , eines → <i>Verzeichnisdienstes</i> und eines Zeitstempeldienstes (→ <i>Zeitstempel</i>).
Trusted-Third-Party (TTP – vertrauenswürdige dritte Partei)	Instanz, die von zwei oder mehreren anderen Instanzen als vertrauenswürdig anerkannt wird und beispielsweise → <i>Zertifikate</i> herausgibt.

⁵⁷ Siehe auch Abschnitt 4.7.1 „Symmetrische Kryptoalgorithmen“.

tunneling	Kryptografisch gesicherte → <i>Ende-zu-Ende-Verbindung</i> zweier Instanzen unter Zuhilfenahme von Kommunikationspfaden einer oder mehrerer weiterer Instanzen, die jedoch den Informationsgehalt des eigentlichen Datenaustausches nicht verändern.
UART (<i>universal asynchronous receiver transmitter</i>)	Universell einsetzbarer und unabhängig vom → <i>Mikroprozessor</i> asynchron arbeitender Baustein zum Senden und Empfangen von Daten. Somit ist es nicht mehr notwendig, dass die Kommunikation auf Bit- und Byteebene vom Mikroprozessor erledigt wird. Dies führt zu einer Vereinfachung der Kommunikationsprotokolle und kann auch zur Realisierung von höheren Datenübertragungsgeschwindigkeiten als bei einer reinen Softwarelösung durch den Mikroprozessor benutzt werden. ⁵⁸
Übertragungsprotokoll (<i>transport protocol</i>)	Bezeichnet in der Chipkartenwelt die Mechanismen zum Senden und Empfangen von Daten zwischen Terminal und Chipkarte. Das Übertragungsprotokoll beschreibt im Detail die benutzten OSI-Protokollschichten, den Datenaustausch im Gutfall, Fehlererkennungsmechanismen und Reaktionsmechanismen bei Fehlern. ⁵⁹
UCS (<i>universal character set</i>)	Weiterentwicklung von ASCII und Unicode zur Codierung von Schriftzeichen und ist in der internationalen Norm ISO/IEC 10646 festgelegt. UCS verwendet 32 Bit für die Codierung eines Zeichens, wobei jedoch nur die Hälfte des möglichen Adressraums genutzt wird ($2^{32}/2 = 2\,147\,483\,648$). Mit diesem Adressraum können alle Schriftzeichen in allen Sprachen dieser Welt dargestellt werden. UCS ist so definiert, dass → <i>Unicode</i> eine Untermenge davon darstellt und die Codierung der ersten 128 Zeichen der ASCII-Codierung entspricht. ⁶⁰
UICC (<i>universal integrated chip card</i>)	→ <i>Chipkarte</i> mit einem für Chipkarten-Anwendungen (→ <i>Anwendung</i>) in der Telekommunikation optimierten → <i>Chipkarten-Betriebssystem</i> nach ISO/IEC 7816. Die normative Grundlage für die UICC ist die TS 102.221, die von → <i>ETSI</i> herausgegeben wird. Die UICC ist die Grundlage für die → <i>USIM</i> . Sie kann die herkömmliche Kreditkartengröße ID-1 haben oder auch als kleine Plug-In-Karte in ID-000 ausgeführt sein.
IM (<i>user identity module</i>)	Veralteter Begriff für → <i>USIM</i> .
UML (<i>unified modeling language</i>) [OMG]	Grafisch orientierte methodenunabhängige Modellierungssprache zur abstrakten Beschreibung statischer und dynamischer Aspekte von objektorientierten Programmen. Die aktuelle Version der UML ist 1.3. Die Grundlagen zur Notation und Semantik von UML wurden in den 90er Jahren von Grady Booch, James Rumbaugh und Ivar Jacobson geschaffen. UML ist unabhängig von einem bestimmten → <i>Vorgehensmodell</i> zur Softwareentwicklung. ⁶¹ Die Object Management Group (OMG) ist verantwortlich für die Weiterentwicklung der UML.
UMTS (<i>universal mobile telecommunication system</i>)	Der europäische Nachfolger von GSM; gehört zur → <i>IMT-2000-Familie</i> . UMTS ist ein digitales, zellulares, betreiberübergreifendes, länderübergreifendes und bodengebundenes Mobilfunksystem der dritten Generation (→ <i>3G</i>). Der diesem Mobilfunksystem zugewiesene Frequenzbereich liegt bei 2 000 MHz. Das UMTS-System ist durch eine Reihe von Spezifikationen definiert, die unter der Schirmherrschaft von → <i>3GPP</i> erstellt wurden und deren Herausgeber → <i>ETSI</i> ist. UMTS stellt den nächsten großen Evolutionsschritt nach → <i>GSM</i> dar. Die wesentlichen Änderungen gegenüber GSM sind eine neue Luftschnittstelle in → <i>CDMA-Technik</i> und deutlich höhere Datenübertragungsraten von bis zu 2 MBit/s. ⁶²

⁵⁸ Siehe auch Abschnitt 3.4.3 „Zusatzhardware“.

⁵⁹ Siehe auch Kapitel 6 „Datenübertragung zur Chipkarte“.

⁶⁰ Siehe auch Abschnitt 4.2 „Codierung alphanumerischer Daten“.

⁶¹ Siehe auch Abschnitt 15.7 „Vorgehensmodelle“.

⁶² Siehe auch Abschnitt 13.3 „Das UMTS-System“.

Unicode [Unicode]	Weiterentwicklung der bekannten ASCII-Codierung von Schriftzeichen. Im Gegensatz zum 7-Bit-ASCII-Code verwendet Unicode 16 Bit für die Codierung. Dies ermöglicht es, die Schriftzeichen der am meisten verbreiteten Sprachen dieser Welt zu unterstützen. Die ersten 256 Zeichen von Unicode sind mit ASCII nach ISO 8859-1 identisch. ⁶³
Uplink	Verbindung von einem untergeordneten System (z. B. Mobiltelefon) zu einem übergeordneten System (z. B. Basisstation). Das Gegenteil ist der → <i>Downlink</i> .
Upload	Übertragen von Daten von einem untergeordneten System (z. B. Terminal) an ein übergeordnetes System (Hintergrundsystem, Host). Das Gegenteil ist der → <i>Download</i> .
URL (<i>uniform resource locator</i>)	Eindeutige alphanumerische Adresse im → <i>WWW</i> .
USIM (<i>universal subscriber identity module</i>)	Übliche Bezeichnung für die Chipkarten-Anwendung (→ <i>Anwendung</i>) für UMTS, ⁶⁴ die auf einer → <i>UICC</i> residiert. In der Praxis wird der Begriff USIM jedoch nicht nur für die Anwendung, sondern auch für die UMTS-Chipkarte verwendet, obwohl dies eigentlich nicht ganz korrekt ist. Die USIM ist der Träger der Identität des Teilnehmers und hat als Hauptaufgabe die Sicherstellung der Echtheit der Mobilstation gegenüber dem Netzwerk und umgekehrt. Zusätzliche Aufgaben sind die manipulationssichere Ausführung von Programmen (Authentisierung), die Benutzeridentifizierung (mittels PIN) und die Speicherung von Daten, wie beispielsweise Telefonnummern. Die normative Grundlage für die USIM ist die TS 31.102, die von → <i>ETSI</i> herausgegeben wird. Das Äquivalent der USIM bei → <i>GSM</i> ist die → <i>SIM</i> . ⁶⁵
USIM Application Toolkit (USAT)	Genormt in TS 31.111; erlaubt einer USIM-Karte, eine aktive Rolle bei der Steuerung des Mobiltelefons zu übernehmen. Mit dem USIM Application Toolkit ist es beispielsweise einer USIM möglich, Anzeigen auf dem Display auszugeben, Eingaben über die Tastatur anzufordern und Nachrichten über die Luftschnittstelle zu senden und zu empfangen. Das USIM Application Toolkit ist die Grundlage für die meisten Zusatzanwendungen bei Mobiltelefonen. Das Äquivalent des USIM Application Toolkit bei GSM ist der → <i>SIM Application Toolkit (SAT)</i> . ⁶⁶
verbindungsorientiert (<i>circuit-switched</i>)	Bei verbindungsorientierten Datenübertragungen wird zum Datenaustausch eine direkte Verbindung (d. h. eine Leitung) zwischen zwei Instanzen hergestellt. In der Regel wird dabei die Zeitdauer der Verbindung abgerechnet und nicht die übertragene Datenmenge (→ <i>paketorientiert</i>). Ein ebenfalls benutzter und äquivalenter Ausdruck ist leitungsorientiert. Typische Beispiele für verbindungsorientierten Datenübertragungen sind analoge Telefonanschlüsse ans Festnetz und ISDN.
vertikaler Prototyp	→ <i>Prototyp</i>
Verwaltungsdaten	Daten, die nur der Verwaltung von → <i>Nutzdaten</i> dienen und für eine → <i>Anwendung</i> ansonsten keinerlei Bedeutung haben.
Verzeichnisdienst (<i>directory service</i>)	Dienst, der in einer Datenbank Listen mit bestimmten Informationen für Anfrager zur Verfügung stellt. Ein typisches Beispiel dafür sind → <i>Zertifikatswiderrufs-listen</i> , in denen bei einer → <i>PKI</i> alle nicht mehr gültigen und akzeptierten Zertifikate aufgeführt sind.

⁶³ Siehe auch Abschnitt 4.2 „Codierung alphanumerischer Daten“.

⁶⁴ Siehe auch Abschnitt 13.2 „Das GSM-System“.

⁶⁵ Siehe auch Abschnitt 13.2.4 „Die SIM“.

⁶⁶ Siehe auch Abschnitt 13.2.4 „Die SIM“.

Vielfachzugriffsverfahren (<i>multiple access</i>)	Funk- und informationstechnische Methoden, um eine begrenzte Frequenzbandbreite einer Funkübertragung gleichzeitig oder scheinbar gleichzeitig möglichst vielen Teilnehmern zur Verfügung zu stellen. Die üblichen vier Verfahren sind Frequenzvielfachzugriff ($\rightarrow$ <i>FDMA</i>), Zeitvielfachzugriff ($\rightarrow$ <i>TDMA</i>), Codevielfachzugriff ($\rightarrow$ <i>CDMA</i>) und Raumvielfachzugriff ($\rightarrow$ <i>SDMA</i>). ⁶⁷
Virginalkarte	Karte, die noch nicht mit einem Chip versehen und noch nicht optisch oder elektrisch personalisiert ist. Eine Virginalkarte ist im Wesentlichen ein bedruckter, uniformer $\rightarrow$ <i>Kartenkörper</i> , wie er in der Massenproduktion von Karten hergestellt wird.
Virtual Machine (VM)	Ein in Software simulierter $\rightarrow$ <i>Mikroprozessor</i> mit in der Regel eigenem Opcode für die Maschinenbefehle und einem ebenfalls simulierten Adressraum. Auf diese Weise wird eine von den Hardware-Gegebenheiten unabhängige Gestaltung von Software möglich. So kann beispielsweise der virtuelle Adressraum einer VM um ein Vielfaches größer sein als derjenige, welcher durch die Hardware zur Verfügung gestellt wird. Im Umfeld von Java wird für die geschlossene Umgebung der VM oft auch der Begriff „Sandbox“ verwendet. ⁶⁸
virtuelle Chipkarte (<i>virtual smart card</i>)	Die softwaretechnische Simulation einer Chipkarte in einem anderen System, z. B. in einem Sicherheitsmodul oder Mobiltelefon. Ein Spezialfall der virtuellen Chipkarte ist die virtuelle Händlerkarte, bei der die Simulation der Chipkarte im Terminal des Händlers abläuft.
virtuelle Händlerkarte (<i>virtual merchant card</i>)	$\rightarrow$ <i>virtuelle Chipkarte</i>
Visa Cash	Markenname von Visa für technologisch unterschiedliche elektronische Geldbörsensysteme mit Chipkarten.
Visa Easy Entry (VEE)	Verfahren für die einfache Migration von magnetstreifenbasierten Kreditkarten hin zu Kreditkarten mit Mikrocontrollerchip. Dazu werden der Name des Kartenbesitzers und alle Daten der Magnetstreifenspur in einem EF unter einem für Visa reservierten DF abgelegt. Bei einer Kreditkartenzahlung liest das Terminal die für diese Transaktion notwendigen Daten statt vom Magnetstreifen aus dem Chip. Der Vorteil des Verfahrens besteht darin, dass nur am POS das Terminal mit einer Chipkartenkontaktiereinheit umgerüstet werden muss und das gesamte Hintergrundsystem ohne Änderungen weiter betrieben werden kann.
voll duplex (<i>full duplex</i>)	Datenübertragung, bei der miteinander kommunizierende Geräte gleichzeitig senden und empfangen können ($\rightarrow$ <i>halbduplex</i>).
VOP	$\rightarrow$ <i>OP</i>
Vorgehensmodell (<i>life cycle model</i>)	Auch Prozessmodell genannt; legt in abstrakter Form den organisatorischen Rahmen, die Arbeitsabläufe und die Aktivitäten mit dazugehörigen Voraussetzungen und Ergebnissen für Entwicklungen fest. Zweck ist eine allgemein einsetzbare und einheitliche Arbeitsweise bei Entwicklungen. Beispiele für Vorgehensmodelle sind Wasserfallmodell, V-Modell und zyklische Entwicklungsmodelle. ⁶⁹
Vorpersonalisierung	andere Bezeichnung für $\rightarrow$ <i>Initialisierung</i>
Wafer	Dünne Scheibe Silizium, auf der mit halbleitertechnischen Mitteln Chips aufgebaut werden. Typische Durchmesser von Wafer sind 150 mm (6 Zoll), 200 mm (8 Zoll) und 300 mm (12 Zoll).
WAP (<i>wireless application</i>)	Dieser Begriff wird für eine Reihe von Spezifikationen zur Realisierung der Anbindung von mobilen Endgeräten (Mobiltelefone, PDAs, ...) über ein draht-

⁶⁷ Siehe auch Abschnitt 13.1.1 „Vielfachzugriffsverfahren“.

⁶⁸ Siehe auch Abschnitt 5.14.1 „Java Card“.

⁶⁹ Siehe auch Abschnitt 15.7 „Vorgehensmodelle“.

<i>protocol</i>) [WAP]	loses Netzwerk an einen Server zwecks unmittelbaren Informationsaustausches benutzt. Die übliche Anwendung von WAP ist die Realisierung von Internetdiensten auf einem Mobiltelefon, weitgehend unabhängig vom jeweiligen Mobilfunkstandard. Die Bezeichnung Wireless Application Protocol steht im Übrigen neben der Technologie auch noch für das Protokoll zwischen Endgerät und Hintergrundsystem. Das seit Juni 1997 existierende und von den Firmen Phone.com, Ericson, Motorola und Nokia gegründete WAP-Forum ist das international agierende Normungsgremium für WAP und setzt sich aus Vertretern von über 350 Firmen zusammen. ⁷⁰
WAP Forum	→ <i>WAP</i>
Warmreset	→ <i>Reset</i>
WCDMA (<i>wideband code division multiple access</i>)	→ <i>CDMA</i>
Whitelist	Liste in einer Datenbank, auf der alle Chipkarten oder Geräte vermerkt sind, die in einer bestimmten → <i>Anwendung</i> verwendet werden dürfen (→ <i>Sperrliste</i> , <i>White List</i>).
white plastic (weiße Karte)	Nicht personalisierte Blankokarten, die in betrügerischer Absicht verwendet werden. Der Begriff kommt ursprünglich von den typischen Kartenrohlingen aus weißem Kunststoff zur Herstellung von Testkarten. Mittlerweile versteht man jedoch unter <i>white plastics</i> auch bedruckte Karten mit den unterschiedlichsten → <i>Kartenelementen</i> , wie beispielsweise noch nicht hochgeprägte Kreditkartenrohlinge mit Magnetstreifen und Hologramm.
Whitebox Test	Beim diesem Test, oft auch Glassbox Test genannt, geht man davon aus, dass die testende Instanz vollständige Kenntnis über alle internen Abläufe und Daten der zu prüfenden Software hat.
WIM (<i>wireless application protocol (WAP) identity module</i>)	Sicherheitsmodul für → <i>WAP</i> -Endgeräte. Die Spezifikation beschreibt eine zu PKCS#15 kompatible Chipkarten-Anwendung (→ <i>Anwendung</i>). Die Hauptaufgaben eines WIM sind die Erstellung und Prüfung von → <i>digitalen Signaturen</i> und die Verschlüsselung von Daten. Eine WIM kann damit entweder eine eigene physikalische Chipkarte sein oder eine von mehreren Anwendungen auf einer Multiapplikationskarte. Typischerweise ist die WIM eine Anwendung auf einer → <i>SIM</i> oder → <i>USIM</i> .
Windows for Smart Cards [Microsoft]	Auch WfSC, WSC; offenes → <i>Chipkarten-Betriebssystem</i> , (→ <i>offenes Chipkarten-Betriebssystem</i>) der Firma Microsoft, das mehrere → <i>Anwendungen</i> (→ <i>Multiapplication-Chipkarte</i>) und nachladbare Programme unterstützt. Eine Besonderheit von Windows for Smart Cards ist, dass ein → <i>FAT</i> -basierendes Dateisystem benutzt wird. ⁷¹
Wissensteilung (<i>shared secrets</i>)	Dieses Prinzip besagt, dass es niemanden gibt, der von einem bestimmten System alles weiß. Die bewusste Aufteilung des Wissens verhindert, dass einzelne Personen durch Dritte angreifbar werden oder durch ihr Wissen zu viel Macht über ein System erlangen können. Bei der Entwicklung von Sicherheitskomponenten ist die Wissensteilung auf mehrere Personen ein üblicher Ansatz.
WML (<i>wireless markup language</i>)	Auf XML aufbauende logische Auszeichnungssprache, um Anwendungen für WAP zu erstellen. WML ist sehr ähnlich zu HTML. Die auf einem WAP-Server in einer WML-Seite abgelegte WML-Applikation wird von einem Converter on-the-fly in kompakten WML-Bytecode übersetzt, über das drahtlose Netzwerk zum mobilen Endgerät übertragen und dann dort von einem Microbrowser (→ <i>Browser</i>) interpretiert.

⁷⁰ Siehe auch Abschnitt 13.5 „Die WIM“.

⁷¹ Siehe auch Abschnitt 5.7 „Dateiverwaltung“.

work around	In der Softwareentwicklung die Umgehung eines bekannten Fehlers durch „Um-den-Fehler-Herumprogrammieren“. Der Fehler als solcher wird durch einen work around nicht beseitigt, sondern lediglich seine negativen Auswirkungen auf den Rest des Programms. Beispielsweise werden festgestellte Fehler von ROM-basierten → <i>Chipkarten-Betriebssystemen</i> nachträglich typischerweise mit work arounds im EEPROM durchgeführt, sodass diese Fehler keine negativen Auswirkungen auf das Chipkarten-Betriebssystem haben. Dabei kann jedoch durch den work around durchaus die Funktionalität des Chipkarten-Betriebssystems gegenüber dem ursprünglichen Umfang eingeschränkt werden.
WWW, W3 (world wide web)	Teil des weltweiten Internets, den vor allem die Möglichkeit der beliebigen Verknüpfung von Dokumenten durch Hyperlinks und die Integration von multimedialen Objekten in Dokumente charakterisiert.
X.509	Die von der → ITU herausgegebene X.509-Norm definiert Aufbau und Codierung von → <i>Zertifikaten</i> . Sie ist die weltweit am häufigsten eingesetzte Norm für Zertifikatsstrukturen (→ <i>PKI</i>).
XML (extended markup language)	Logische Auszeichnungssprache und sowohl Nachfolger als auch Ergänzung von HTML. Mit XML lassen sich eigene Sprachelemente festlegen, sodass sich andere Auszeichnungssprachen wie HTML oder WML mittels XML definieren lassen. XML selber ist wiederum eine Untermenge der sehr mächtigen ISO-Norm SGML (standard generalized markup language).
Zeitstempel (time stamp)	Einmal mit einer → <i>digitalen Signatur</i> versehene Bescheinigung einer Instanz, dass dieser Instanz bestimmte digitale Daten zu einem bestimmten Zeitpunkt vorgelegen haben (→ <i>PKI</i>).
Zelle (cell)	Im Mobilfunkbereich ist die kleinste geografische Einheit, in der das Netzwerk aufgeteilt ist.
Zellulartechnik	Analoges oder digitales Mobilfunksystem, das in Zellen organisiert ist. Üblicherweise befindet sich im ungefähren Mittelpunkt der Zelle die Send- und Empfangsstation des Netzwerks, üblicherweise Basisstation genannt. ⁷²
Zertifikat (certificate)	Ein von einer vertrauenswürdigen Instanz digital signierter öffentlicher Schlüssel mit dazugehörigen Verwaltungsdaten, damit dieser von Dritten als authentisch anerkannt werden kann (→ <i>PKI</i>). Die verbreitetste und bekannteste Festlegung des Aufbaus und Codierung von Zertifikaten ist die X.509-Norm.
Zertifikatswiderrufsliste (certificate revocation list – CRL)	Liste in einem → <i>Verzeichnisdienst</i> , die alle gesperrten und nicht mehr akzeptierten Zertifikate innerhalb einer → <i>PKI</i> enthält.
Zertifizierung	Der Begriff Zertifizierung wird manchmal und in nicht ganz korrekter Weise als Äquivalent zum Begriff Evaluierung (siehe auch Evaluierung) benutzt.
Zertifizierungsinstanz (CA – certificate authority)	Zertifizierungsstelle in einer Public-Key Infrastruktur (→ <i>PKI</i>), die öffentliche Schlüssel für → <i>digitale Signaturen</i> beglaubigt, d. h. sich für ihre Echtheit verbürgt. Dazu unterschreibt die Zertifizierungsinstanz mit ihrem geheimen Schlüssel die öffentlichen Schlüssel der Anwender und stellt bei Bedarf die signierten öffentlichen Schlüssel in einem Verzeichnis (→ <i>Verzeichnisdienst</i>) in Form eines → <i>Zertifikates</i> zur Verfügung. Die Zertifizierungsinstanz kann die dazu notwendigen Schlüsselpaare (geheimer und öffentlicher Schlüssel) selber erzeugen. Zertifizierungsinstanzen sind oft aus organisatorischen Gründen hierarchisch aufgebaut, wobei die oberste Zertifizierungsinstanz auch als Top-Level-CA oder Root-CA bezeichnet wird.

⁷² Siehe auch Abschnitt 13.1.2 „Zellulartechnik“.

ziffern	Aufprägen oder Aufdrucken einer Nummer bei Chipkarten. Dies wird typischerweise bei der Produktion von anonymen Telefonwertkarten durchgeführt, um diesen eine sichtbare und einzigartige Nummer zur eindeutigen Identifizierung zu geben.
ZKA	In Deutschland die Koordinierungsinstanz für die elektronischen Zahlungsverfahren der deutschen Banken. Der ZKA setzt sich aus dem Deutsche Sparkassen- und Giroverband (DSGV), dem Bundesverband der Deutschen Volks- und Raiffeisenbanken (BVR), dem Bundesverband deutscher Banken (BdB) und dem Verbund öffentlicher Banken (VÖB) zusammen. Den Vorsitz des ZKA übernimmt jedes Jahr ein anderer der vier Bankenverbände.
Zugriffsbedingungen (<i>access conditions</i> – AC)	Im Zusammenhang mit dem Dateisystem einer Chipkarte versteht man unter den Zugriffsbedingungen einer Datei eine endliche Zahl von Bedingungen für vom Betriebssystem unterstützte Zugriffe (z. B. Lesen, Schreiben, Löschen, ...) auf diese Datei, wobei die Bedingungen für jede Zugriffsart in der Regel unabhängig voneinander sind und die Bedingungen vor dem Zugriff erfüllt sein müssen.
Zusatzanwendungen (<i>VAS</i> – <i>value added</i> <i>service</i>)	Weitere Chipkarten-Anwendungen (→ <i>Anwendung</i>), die sich neben einer Hauptanwendung auf einer Chipkarte befinden. Voraussetzung dafür sind in der Regel → <i>Multiplication-Chipkarten</i> .

16.2 Übersetzung von Fachwörtern

Die beiden Übersetzungstabellen (Deutsch – Englisch, Englisch – Deutsch) auf den folgenden Seiten umfassen Fachbegriffe aus dem Umfeld von Chipkarten und Kryptografie. Die Tabellen sind auch die Referenz für die in diesem Buch benutzten Übersetzungen. Ein ausführliches Abkürzungsverzeichnis befindet sich am Anfang des Buches, die entsprechenden Definitionen und Erläuterungen am im Glossar in Kapitel 16.1.

Das Zeichen „★“ symbolisiert, dass es keine allgemein übliche Übersetzung gibt und deshalb im Deutschen der englische Begriff gebraucht wird. Mögliche Übersetzungen, die in der Praxis nicht oder nicht mehr verwendet werden und im umgangssprachlichen Gebrauch deshalb etwas seltsam klingen, werden durch „(unüblich)“ gekennzeichnet. So wird beispielsweise in manchen Normen der Begriff „Terminal“ mit „Datenendgerät“ und ein „Message Authentication Code (MAC)“ mit „Nachrichtenglaubigkeitscode“ übersetzt. Dies entspricht jedoch nicht den in der Chipkartenwelt üblichen Begriffen und ist in den Tabellen deshalb eigens markiert.

Bei möglichen unterschiedlichen Übersetzungen eines Begriffs wurde immer die allgemein übliche Form verwendet; bei zwei gebräuchlichen Übersetzungen wurden beide Begriffe aufgeführt. Gängige Abkürzungen werden bei den jeweiligen Begriffen in Klammern genannt.

16.2.1 Übersetzungsliste Deutsch – Englisch

Abbuchung	billing	Auflade-SAM	load SAM (LSAM)
Ablaufdatum	expiry date	Aufzeichnungstechnik	recording technique
Ablaufkontrolle	sequence control	Ausfall	failure
Abnutzung des	wear of magnetic stripe	Ausfuhrbeschränkung	export regulation
Magnetstreifens		Ausfuhrgenehmigung	export license
Abrechnung	clearing	Ausgang	output
Abtastzone	test zone	Ausgangszustand	initial state
Adhäsion	adhesion	äußere Störung	external disturbance
aktiver Zustand	active state	Austausch	interchange
Aktivierung	activation	Ausweiskarte	ID card, identification card
Aktivierungszustand	activation state	Ausweissysteme	identification systems
Algorithmus mit geheimem Schlüssel	secret key algorithm	Auswurf	ejection
(unüblich) ★		authentisiert	authenticated
Änderungshäufigkeit	update rate	Authentisierung	authentication
Anfangswartezeit ★	initial waiting time	Authentizität	authenticity
Anfangszeichen ★	initial character		
Anforderung	request	Bankanwendung	banking application
Angriff	attack	Banknachricht	banking message
Anpassungsfaktor der	bit rate adjustment factor	Bediener	operator
Bitrate (unüblich) ★		Beendigungscode	completion code
ansteigende Flanke	rising edge	Beendigungszustand	completion state
Anstiegszeit	rise time	Befehl	command
Antenne	antenna	Befehlsabdeckung	statement coverage
Antwort	response	Begrenzung	delimitation
Anwenderkarte	user card	beliebige Daten	discretionary data (DD)
Anwendung	application	Benutzermodus	user mode
Anwendung für Vorauszahlungsfunktion	pre-payment application	Benutzerschlüssel	user key
Anwendungsanbieter	application provider	Berechnung	computation
Anwendungsbezeichner (unüblich) ★	application identifier (AID)	berechtigte Instanz	entitled authority
Anwendungsebene	application level	Berechtigung	authorisation
Anwendungsinstanz	application entity	Bestätigung	confirmation
Anwendungsprotokoll	application protocol	Betrag	amount
Anwendungsschicht	application layer	Betriebssystem	operating system
Anwendungssicherheitsmodul	secure application module (SAM)	Betriebssystemhersteller	operating system manufacturer
anwendungsspezifischer Befehlssatz	application-specific command set	Betriebssystemkern	kernel of operating system
Anzeige	display	Bezeichner	identifier
Architekturentwurf	architectural design	Biegesteifigkeit	bending stiffness
asymmetrischer Algorithmus	asymmetric algorithm	Bitmap-Codierung	bitmap coding
asynchron	asynchronous	Bitsynchronisationsfolge	bit synchronisation sequence
Attribut	attribute	Blockprotokoll	block protocol
Aufdrückkraft	pushing force	Blocksicherungszeit (unüblich) ★	block guard time (BGT)
Aufforderung	challenge	Blockverschlüsselungsalgorithmus	block cipher algorithm
Auffüllung	padding	Bodensatz	float
Aufladebevollmächtigtger	load agent	boolesch	boolean
Aufladeeinrichtung	load device	Börsenanbieter	purse provider
Aufladeeinrichtungsanwendung	load device application (LDA)	Börsenanbieter-SAM	purse provider SAM (PPSAM)
Aufladeprotokoll	load log	Börseninhaber	purse holder
		branchenübergreifende elektronische Geldbörse	intersector electronic purse (IEP)

branchenübergreifende Nachricht	interindustry message	differentielle Leistungsanalyse	differential power analysis (DPA)
branchenübergreifende Umgebung	intersector environment	digitale Signatur	digital signature
branchenübergreifendes Kommando	interindustry command	digitaler Fingerabdruck	digital fingerprint
Byteprotokoll	byte protocol	Diversifikationsdaten	diversification data
Chargennummer	batch number	diversifizierter Schlüssel	diversified key
Chefprogrammiererteam	chief programmer team	Draht-Bonding	wire-bonding
Chemikalienbeständigkeit	resistance to chemicals	Dualmodus	dual mode
chinesischer Restklassensatz	chinese remainder theorem (CRT)	dynamischer Biegetest	dynamic bending stress
Chipkarte	smart card (SC), integrated chip card (ICC)	dynamischer Torsionstest	dynamic torsional stress
Chipkarten-Anwendung	smart card application	E/A (<i>unüblich</i>) ★	I/O
Chipkarten-Betriebssystem	smart card operating system	E-/A-Kontakte (<i>unüblich</i>) ★	I/O contacts
Chipkarten-Entwicklung	smart card development	Ebene	level
Chipkarten-Hersteller	smart card manufacturer	Echtheit	authenticity
Chipkarten-Mikrocontroller	smart card micro-controller	Echtschlüssel	lifekey
Chipkarten-Technologie	smart card technology	Echtzeit	real-time
Codeabdeckung	code coverage	ec-Karte	ec card
Codierung	coding	eigene	proprietary
Datei	file	Eingangskapazität	input capacitance
Dateibezeichner (<i>unüblich</i>) ★	file identifier (FID)	Eingangsspannung	input voltage
Dateimanagement	file management	Eingangsstrom	input current
Datenaustausch	data exchange	eingebettet	embedded
Datenelement	data element	Einlagenkarte	mono layer card
Datenendgerät (<i>unüblich</i>)	terminal	Einschubleser (<i>unüblich</i>)	internal reader
Datensatz	record	Einsteckkräfte	insertion forces
Datensatzbezeichner (<i>unüblich</i>) ★	record identifier (RID)	Einweg-Authentisierung	one-way authentication
Datenschutzgesetz	data protection law	Einwegfunktion	one-way function
Datensicherungscode (<i>unüblich</i>) ★	message authentication code (MAC)	einzigartig	unique
Datenspeicher	data storage	Electronic Cash	electronic cash
Datenteil	data component	elektrische Verbindung	electrical link
Datenübertragung	data transmission	elektrischer Widerstand der Kontakte	electrical resistance and impedance of contacts
Datenübertragungsgeschwindigkeit	data transmission rate	elektromagnetische Felder	electromagnetic fields
Datenverbindung	data link	elektronische Börse	electronic purse
Deaktivierung	deactivation	elektronische Geldbörse	electronic purse
Deckfolie	overlay film	elektronischer Wert	electronic value
deterministisch	deterministic	elektronisches Signal	electronic signal
differentielle Fehleranalyse	differential fault analysis (DFA)	elektrostatische Entladung	electrostatic discharge (ESD)
		elementare Zeiteinheit	elementary time unit (etu)
		(<i>unüblich</i>) ★	
		Elementary File ★	elementary file (EF)
		Empfänger	receiver
		Ende-zu-Ende-Protokoll	end-to-end protocol
		Endgerät (<i>unüblich</i>) ★	terminal

endlicher Automat	finite state machine	geschützt	protected
Endsumme	total	geschützter Modus	protected mode
Entflammbarkeit	flammability	geschütztes Modul	secure module
Entscheidungs- abdeckung	decision coverage	gewickelte Spule	wound coil
Entsprechung	equivalence	gewidmete Datei	dedicated file (DF)
Erfassung	collection	(unüblich) ★	
Erkennung	recognition	Grad der Genauigkeit	level of rigour
Erweiterung	extension	Grenzwert	limit value
Erzeugung	generation	Grundzustand	idle state
extern eingeleitet	externally initiated	Guillochen	guilloche pattern
externe Anwendung	external application		
externe Authentisierung	external authenticate	halbduplex	half-duplex
		Handy	mobile telephone
		Hardmaske	hard mask
		Hardwaresicherheits- merkmal	hardware security feature
fallende Flanke	falling edge	Hash-Algorithmus	hash algorithm
Falltür	trap door	Hash-Funktion	hash function
falscher Block	false block	Hash-Wert	hash value
fälschungssicher	forgery-proof	Hauptschlüssel	master key
Fehlbedenungs- zähler (FBZ)	key fault presentation counter (KFPC), retry counter (RC)	Heißsiegel-Verfahren	heat-sealing method
Fehlerbehandlung	error handling	Herausgeber	issuer
Fehlerbehebung	error recovery	Herausziehungskräfte	withdrawing forces
Fehlererkennung	error detection	High-Zustand	high state
Fehlererkennungscode	error detection code (EDC)	hochohmiger Zustand	high impedance state
fehlerhafter Block	erroneous block	Hochprägung	embossing
Fehlfunktion	malfuction	höchstwertiges Bit	most significant bit
Feinentwurf	detailed design	hochzählen	increment
Feldwechsel	flux transition	Höhe der Prägung	embossing relief height of character
feststehender Schlüssel	fixed key	Höhe und Ober- flächenprofil des Magnetstreifens	height and surface profile of the magnetic stripe
flüchtiger Speicher	volatile memory	höherwertiges Bit	Most Significant Bit (msb)
Folgezustand	consecutive state		
formales	formal model of security	I/O-Kontakte	I/O contacts
Sicherheitsmodell	policy	Identifikations- nummer	identification number
Formatzeichen	format character	Identifikations- nummer des Her- ausgebers	issuer identification number
(unüblich)		IEP-Ausgabezentrum	IEP issuing center
Funktionalität	functionality	IEP-Saldo	IEP balance
für spätere Verwen- dung freigehalten	reserved for future use (RFU)	Implementierung	implementation
(unüblich) ★		Impulsgenerator	pulse generator
		Impulslänge	pulse duration
ganzzahlig	integer	Initialisierung	initialisation
geätzte Spule	etched coil	Initialzahl	seed number
Gebührenimpuls	charge pulse	Instruction Byte	instruction byte (INS)
gedruckte Spule	printed coil	(INS) ★	
gegenseitige Authentisierung	mutual authentication	Integrität	integrity
geheimer Schlüssel	secret key	Interface Device	interface device (IFD)
Geldausgabeautomat	automated teller machine (ATM)	(IFD) ★	
Geldeinheit	monetary unit	Irisdruck	rainbow print
Generierung	generation		
gerade Parität	even parity	Karte einführen	insert card
geschlossenes System	closed system	Karte rücksetzen	card reset

Kartenakzeptant	card acceptor	Kundenkarte	loyalty card
Kartenausgeber	card issuer	Kurznachricht	short message
Kartenbesitzer	card holder		
Kartenendgerät (unüblich) ★	card accepting device (CAD)	Lage der Kontakte	location of contacts
Kartenentnahme	card removal	Lasergravur	laser engraving
Kartenfolgenummer	card sequence number	Last	load
Kartenformat	dimensions of cards	Lastenheft	user's requirement specification
Kartenherausgeber	card issuer	Lebenszyklus	life cycle
Karteninhaber	card holder	Leistungsanbieter	service provider
Kartenkörper	card body	leitende Farbe	conductive ink
Kartenleser	card reader	Leseschutz	read protection
Kartenschnittstelle	card interface	Lichtdurchlässigkeit	light transmittance
Kartenstatus	card status	linear rückgekoppel- tes Schieberegister	linear feed shift register (LFSR)
Kartentelefon	card phone	logische Kanäle	logical channels
Kartenterminal	card terminal	logische Ressourcen	logical resources
Kartenwölbung	card warpage	logisches Komple- ment	logical complement
Kavität	cavity	logisches ODER	logical OR
Kennzeichen	tag	logisches UND	logical AND
Kernfolien	core foils	low Pegel	low level
Kettung	chaining	low Zustand	low state
Klartext	plaintext	Luftschnittstelle	air interface
Knotenadresse (unüblich) ★	node address (NAD)		
Koerzitivität	coercivity	MAC-Berechnung	MAC computation
Kommando	command	MAC-Überprüfung	MAC verification
Kommando-Antwort- Paar	command response pair	Magnetstreifen	magnetic strip, magstripe
Kommandocode	command code	Magnetstreifenkarte	magnetic strip card, mag- stripe card
Kommando-Header	command header		
Kommandointerpreter	command interpreter	Mangel	defect
Kommandoklasse (unüblich) ★	command class	maschinenlesbare Karte	machine-readable card
Kommandokopf (unüblich) ★	command header	Maskenprogrammierung	mask programming
Kommunikations- einrichtung	communication device	Masse	ground
Kommunikations- kanal	communication channel	Master File (MF) ★	master file (MF)
Kommunikations- netzschnittstelle	communication network interface	Mehrlagenkarte	multi layer card
Kontaktbelegung	contact assignment	Mehrwertdienst	value added service (VAS)
Kontaktiereinheit	contacting unit	Menge	set
Konto	account	Meßfühler	probe
Kontonummer	account number	Mikroschrift	microprint
Konzentrator	concentrator	Mobiltelefon	mobile telephone
Krankenversiche- rungskarte (KVK)	health insurance card	Modus	mode
Kriterium	criteria	Multiapplication- Chipkarte	multiapplication smart card
kryptoanalytisch	cryptoanalytic	multifunktional	multifunctional
kryptografische Prüffunktion	cryptographic check function	Münzfernsprecher	coin-operated phone, pay- phone
Kryptogramm	cryptogram		
kryptographischer Algorithmus	cryptographic algorithm	Nachricht	message
kryptographischer Schlüssel	cryptographic key	Nachrichtenaustausch	message exchange
		Nachrichtenbegläu- bigungscode (unüblich) ★	Message Authentication Code (MAC)
		Nachrichtenrückge- winnung	message recovery

Nachsatz	trailer	Primzahl	prime number
nachträglich bezahlt	post-paid	Produkthaftung	product liability
Nachweis	proof	Produktspezifikation	product specification
Negativliste	blacklist	Programmierspannung	programming voltage
Nettodaten	net data	Programmierstrom	programming current
nicht umkehrbar	irreversible	proprietäre	proprietary
nicht zertifiziertes	untrusted monitor	Protokollauswahl	protocol type selection
Kontrollgerät		(unüblich) ★	(PTS)
Nichtabstreitbarkeit	non-repudiation	Protokollauswahlbyte	protocol control byte
nichtflüchtiger Speicher	non-volatile memory	(unüblich) ★	(PCB)
niedrigstwertiges Bit	least significant bit (lsb)	Protokollierung	logging
nur Lesespeicher	read only memory	Protokollschicht	protocol layer
Nutzen	copies from one sheet	Prüfkosten	appraisal costs
		Prüfplanung	inspection planning
		Prüfspezifikation	inspection specification
		Prüfsumme	checksum
		Prüfzeichen	check character
Oberflächenprofil der Kontakte	surface profile of contacts	Qualitätsaufzeichnung	quality record
Oberflächenrauheit des Magnetstreifens	surface roughness of the magnetic stripe	Qualitätssicherung (QS)	quality assurance (QA)
obligatorisch	mandatory	Quelladresse	source address (SAD)
obligatorische Datei	mandatory file		
offenes System	open system	Rauschsignal	noise signal
öffentlicher Exponent	public exponent	Realisierung	implementation
öffentlicher Modulo	public modulus	Rechnerknotenpunkt	computer node
öffentlicher Schlüssel	public key	rechtmäßiger Kartenbesitzer	genuine card holder
Offline	offline	Record	record
Offline-Transaktion	offline transaction	Restklassensatz	remainder theorem
Offset	offset	Resynchronisation	resynchronisation
Online	online	Risikomanagement	risk management
Online-Transaktion	online transaction	rohe Gewalt	brute-force
optisch variable Druckfarbe	optically variable ink	Routine	routine
Parität	parity	Rückgewinnung der Nachricht	message recovery
Paritätsbit	parity bit	Rücksetzung	reset
Paritätsprüfung	parity check	(unüblich) ★	
Passwort	password	Rückverfolgbarkeit	traceability
Passwortprüfung	password verification		
Personal Unblocking Key (PUK) ★	personal unblocking key (PUK)	Saldo	balance
Personalisierung	personalisation	Sammelbeauftragter	acquirer
Personalisierungsschema	personalisation scheme	Schattenkonto	shadow account
Personalisierungszentrum	personalisation center	Scheibenummer	wafer number
persönliche Identifikationsnummer (PIN)	personal identification number (PIN)	(unüblich) ★	
physikalischer Angriff	physical attack	Schlüssel	key
PIN-Brief	PIN mailer	Schlüsselaufteilung	key separation
PIN-Eingabe	PIN entry	Schlüsselbezeichner	key identifier
PIN-Pad	PIN pad	Schlüsseldiversifikation	key diversification
PIN-Tastatur	PIN pad	Schlüsselmanagement	key management
Point of Sale (POS) ★	point of sale (POS)	Schlüsselraum	key space
Präfix	prefix	Schlüsseltext	cipher text
Prägen	embossing	Schlüsselzertifizierung	key certification
		Schnittstelle	interface
		Schreibweise	notation
		Schreibzyklus	write cycle

schutzwürdige Daten	sensitive data	Synchronisierungs- verlust	loss of synchronisation
Schutzzeit	guardtime	Takt	clock
schwarze Liste	blacklist	Taktfrequenz	clock frequency
Schwingungen	vibrations	Taktraten Umrech- nungsfaktor	clock rate conversion factor
Sendebetrieb	transmission mode	(<i>unüblich</i>) ★	
Sendefolgezähler	send sequence counter (SSC)	Tastatur	keyboard
Sender	transmitter	Tastaturbedienung	key operation
serielle Schnittstelle	serial port	Tastenanordnung	key layout
Sicherheitsarchitektur	security architecture	Tastenblock	keypad
Sicherheitsattribut	security attribute	Teilnehmer	participant
Sicherheitsdruck	security print	Testmodus	test mode
Sicherheitsfaden	security thread	Testschlüssel	testkey
Sicherheitsmecha- nismus	security mechanism	Tischleser	desktop reader
Sicherheitsmerkmal	security feature	Transaktionsdatensatz	transaction record
Sicherheitsmodell	security model	Transaktionsver- arbeitung	transaction processing
Sicherheitsniveau	security level	Transaktionsvorgang	transaction process
Sicherheitspolitik	security policy	Transportinstanz	transport entity
Sicherheitsprotokoll	security protocol	Transportschicht	transport layer
Sicherheitsprozedur	security procedure	Trennung	disconnection
Sicherheitsstruktur	security scheme	Übergangsbetrieb	transient mode
Sicherheitsvorgaben	security targets	Übergangszone	transition zone
Sicherungsschicht	data link layer	Überprüfung	verification
Signalamplitude	signal amplitude	Übertragung im Blockmodus	block mode transmission
Signatur	signature	Übertragung im Zei- chenmodus	character mode trans- mission
signierte Quittung	signed acknowledgement	Übertragungsbetrieb	transmission mode
Sitzung	session	(<i>unüblich</i>) ★	
Sitzungsschlüssel	session key	Übertragungsfehler	transmission error
Smart Card Query Language (SCQL) ★	smart card query language (SCQL)	Übertragungsproto- koll	transmission protocol
Softmaske	soft mask	Übertragungsver- fahren	transmission method
Spannungsüber- wachung	voltage watchdog	umkehrbar	reversible
Speicherbereich	memory range	Umrechnungsfaktor	conversion factor
Speicherkarte	memory card	unberechtigter Zugriff	unauthorised access
Speicherplatz	memory space	ungerade Parität	odd parity
Sperrliste	revocation list	unsichtbare UV-Farbe	invisible UV ink
Sperrliste	blacklist	Unterschriftstreifen	signature panel
Sperrliste	stoplist	Unterzeichner	signer
Sperrzustand	locked state	Urheber	originator
Spritzgießen	injection moulding	Ursprungswert	default value
Spur	track	Verbindung	link
Stammverzeichnis	root	vereinbarte Berechnung	agreed calculation
Stapelspeicher	stack	Vereinzelner	singular
Startvorderflanke	start rising edge	Verfahren	procedure
Statische Elektrizität	static electricity	Verfallsdatum	expiry date
Status Word (SW) ★	status word (SW)	Vergleichsprüfung	comparison check
Statusantwort	status response	verkettete Liste	linked list
Störspannung	noise voltage		
Störung	malfuction		
Störungsangriff	disruption attack		
Stromaufnahme	power consumption		
Stromversorgung	power supply		
Summierung	accumulation		
symmetrischer Algo- rithmus	symmetric algorithm		

Verkettung	chaining	Wiederholungsangriffe	replay attacks
Verkettungsunterbrechung (<i>unüblich</i>) ★	chaining abort	Wölbung	warping
verlegte Spule	embedded coil	Working Elementary File (WEF) ★	working elementary file (WEF)
vermeidbarer Abfall	avoidable waste	Write Once Read Multiple (WORM) ★	write once read multiple (WORM)
Verminderung	decrementing	Wurzelverzeichnis	root directory
Verminderungs-funktion	decrease function		
Verschlüsselungsmodul	encipherment module	zahle jetzt	pay now
Version der Schlüssel-datei	key file version	zahle später	pay later
Versorgungsspannung	power supply voltage	zahle vorher	pay before
Versorgungsstrom	supply current	Zähler	counter
Verträglichkeit (<i>unüblich</i>)	compatibility	Zählerwert	counter value
Vertragsüberprüfung	contract review	Zahlungsabbruch	purchase cancellation
vertrauenswürdige	trusted device	Zahlungseinrichtung	purchase device
Gerät		Zahlungseinrichtungsanwendung	purchase device application (PDA)
Vertrauenswürdigkeit	trusted	Zahlungsprotokoll	purchase log
vertrauliche Parameter	confidential parameters	Zahlungs-SAM	purchase SAM
Vertraulichkeit	confidentiality	Zahlungsverkehrsterminal	payment terminal
Vertraulichkeitsvereinbarung	non-disclosure agreement (NDA)	Zeichendarstellung	character representation
Vertreter	agent	Zeichendauer	character duration
Verwaltung	administration	Zeichenprotokoll	character protocol
Verweis	reference	Zeichensatz	character set
Verzeichnis	directory	Zeichenwartezeit	character waiting time
Verzeichnisdienst	directory service	(<i>unüblich</i>) ★	(CWT)
Verzögerung	delay	Zeichenwiederholung	character repetition
vor Inbetriebnahme	pre-operational	Zeitstempeldienst	time stamp service
vorausbezahlt	prepaid	Zertifikat	certificate
Vorgabewert	default value	Zertifikat mit Hilfe eines öffentlichen Schlüssels (<i>unüblich</i>) ★	public key certificate
Vorpersonalisierung	pre-personalisation		
		Zertifikatsdienst	certification service
Währungscode	currency code	Zertifizierungsinstanz	certifying authority (CA)
Währungsumrechnung	currency conversion	Zieladresse	destination address (DAD)
Warenausgabeautomat	vending machine	Zufallszahl	random number (RND)
Wärmeableitung	heat dissipation	zugehöriger Schlüssel	related key
Wärmebeständigkeit	heat resistance	Zugriff	access
Wartezeitverlängerung	waiting time extension (WTX)	Zugriffsbedingung	access condition (AC)
Wasserfallmodell	waterfall model	Zugriffsberechtigung	access right
Wechselkurs	exchange rate	Zugriffskontrolle	access control
Wert	value	zusätzliche Daten	additional data
wiederaufladbare Karte	reloadable card	zusätzliche Schutzzeit	extra guardtime
Wiederherstellungsfunktion	recovery function	Zustand	state
Wiederherstellungs-prozedur	recovery procedure	Zustandsübergang	state transition
Wiederholanforderung	repetition request	Zustandsautomat	state machine
		zuteilbarer Speicher	allocable memory
		Zuverlässigkeit	reliability
		Zuweisung	assignment
		zyklisch	cyclic

16.2.2 Übersetzungsliste Englisch – Deutsch

access	Zugriff	factor	Bitrate (<i>unüblich</i>) ★
access condition (AC)	Zugriffsbedingung	bit synchronisation sequence	Bitsynchronisationsfolge
access control	Zugriffskontrolle	bitmap coding	Bitmap-Codierung
access right	Zugriffsberechtigung	blacklist	Negativliste, schwarze Liste, Sperrliste
account	Konto	block cipher algorithm	Blockverschlüsselungsalgorithmus
account number	Kontonummer	block guard time (BGT)	Blocksicherungszeit (<i>unüblich</i>) ★
accumulation	Summierung	block mode transmission	Übertragung im Blockmodus
acquirer	Sammelbeauftragter	block protocol	Blockprotokoll
activation	Aktivierung	boolean	boolesch
activation state	Aktivierungszustand	brute-force	rohe Gewalt
active low reset	active low reset ★	byte protocol	Byteprotokoll
active state	aktiver Zustand		
additional data	zusätzliche Daten	card accepting device (CAD)	Kartenendgerät (<i>unüblich</i>) ★
adhesion	Adhäsion	card acceptor	Kartenakzeptant
administration	Verwaltung	card body	Kartenkörper
agent	Vertreter	card holder	Karteninhaber, Kartenbesitzer
agreed calculation	vereinbarte Berechnung	card interface	Kartenschnittstelle
air interface	Luftschnittstelle	card issuer	Kartenausgeber, Kartenherausgeber
allocable memory	zuteilbarer Speicher	card phone	Kartentelefon
amount	Betrag	card reader	Kartenleser
antenna	Antenne	card removal	Kartenentnahme
application	Anwendung	card reset	Karte rücksetzen
application entity	Anwendungsinstanz	card sequence number	Kartenfolgenummer
application identifier (AID)	Anwendungsbezeichner	card status	Kartenstatus
application layer	Anwendungsschicht	card terminal	Kartenterminal
application level	Anwendungsebene	card warpage	Kartenwölbung
application protocol	Anwendungsprotokoll	cavity	Kavität
application provider	Anwendungsanbieter	certificate	Zertifikat
application-specific command set	anwendungsspezifischer Befehlssatz	certification service	Zertifikatsdienst
appraisal costs	Prüfkosten	certifying authority (CA)	Zertifizierungsinstanz
architectural design	Architekturentwurf	chaining	Kettung, Verkettung
assignment	Zuweisung	chaining abort	Verkettungsunterbrechung (<i>unüblich</i>) ★
asymmetric algorithm	asymmetrischer Algorithmus	challenge	Aufforderung
asynchronous	asynchron	character duration	Zeichendauer
attack	Angriff	character mode transmission	Übertragung im Zeichenmodus
attribute	Attribut	character protocol	Zeichenprotokoll
authenticated	authentisiert	character repetition	Zeichenwiederholung
authentication	Authentisierung	character representation	Zeichendarstellung
authenticity	Echtheit	character set	Zeichensatz
authenticity	Authentizität	character waiting time (CWT)	Zeichenwartezeit (<i>unüblich</i>) ★
authorisation	Berechtigung	charge pulse	Gebührenimpuls
automated teller machine (ATM)	Geldausgabeautomat (GAA)	check character	Prüfzeichen
avoidable waste	vermeidbarer Abfall		
balance	Saldo		
banking application	Bankanwendung		
banking message	Banknachricht		
batch number	Chargennummer		
bending stiffness	Biegesteifigkeit		
billing	Abbuchung		
bit rate adjustment	Anpassungsfaktor der		

checksum	Prüfsumme	cryptographic algorithm	kryptographischer Algorithmus
chief programmer team	Chefprogrammiererteam	cryptographic check function	kryptografische Prüffunktion
chinese remainder theorem (CRT)	chinesischer Restklassensatz	cryptographic key	kryptographischer Schlüssel
cipher text	Schlüsseltext	currency code	Währungscode
clearing	Abrechnung	currency conversion	Währungsumrechnung
clock	Takt	cyclic	zyklisch
clock frequency	Taktfrequenz		
clock rate conversion factor	Taktraten Umrechnungsfaktor (<i>unüblich</i>) ★	data component	Datenteil
closed system	geschlossenes System	data element	Datenelement
code coverage	Codeabdeckung	data exchange	Datenaustausch
coding	Codierung	data link	Datenverbindung
coercivity	Koerzitivität	data link layer	Sicherungsschicht
coin-operated phone collection	Münzfernsprecher Erfassung	data protection law	Datenschutzgesetz
command	Befehl, Kommando	data storage	Datenspeicher
command class	Kommandoklasse (<i>unüblich</i>)	data transmission	Datenübertragung
command code	Kommandocode	data transmission rate	Datenübertragungsgeschwindigkeit
command header	Kommando-Header, Kommandokopf (<i>unüblich</i>)	deactivation	Deaktivierung
command interpreter	Kommandointerpreter	decision coverage	Entscheidungsabdeckung
command response pair	Kommando-Antwort-Paar	decrease function	Verminderungsfunktion
communication channel	Kommunikationskanal	decrementing	Verminderung
communication device	Kommunikationseinrichtung	dedicated file (DF)	gewidmete Datei (<i>unüblich</i>) ★
communication network interface	Kommunikationsnetz-schnittstelle	default value	Ursprungswert, Vorgabewert
comparison check	Vergleichsprüfung	defect	Mangel
compatibility	Verträglichkeit (<i>unüblich</i>)	delay	Verzögerung
completion code	Beendigungscode	delimitation	Begrenzung
completion state	Beendigungszustand	desktop reader	Tischleser
computation	Berechnung	destination address (DAD)	Zieladresse
computer node	Rechnernotenpunkt	detailed design	Feinentwurf
concentrator	Konzentrator	deterministic	deterministisch
conductive ink	leitende Farbe	differential fault analysis (DFA)	differentielle Fehleranalyse
confidential parameters	vertrauliche Parameter	differential power analysis (DPA)	differentielle Leistungsanalyse
confidentiality	Vertraulichkeit	digital fingerprint	digitaler Fingerabdruck
confirmation	Bestätigung	digital signature	digitale Signatur
consecutive state	Folgezustand	dimensions of cards	Kartenformat
contact assignment	Kontaktbelegung	directory	Verzeichnis
contacting unit	Kontaktiereinheit	directory service	Verzeichnisdienst
contract review	Vertragsüberprüfung	disconnection	Trennung
conversion factor	Umrechnungsfaktor	discretionary data (DD)	beliebige Daten
copies from one sheet	Nutzen	display	Anzeige
core foils	Kernfolien	disruption attack	Störungsangriff
counter	Zähler	diversification data	Diversifikationsdaten
counter value	Zählerwert	diversified key	diversifizierter Schlüssel
criteria	Kriterium	dual mode	Dualmodus
cryptoanalytic	kryptoanalytisch	dynamic bending stress	dynamischer Biegetest
cryptogram	Kryptogramm	dynamic torsional stress	dynamischer Torsionstest

ec card	ec-Karte	flux transition	Feldwechsel
ejection	Auswurf	forgery-proof	fälschungssicher
electrical link	elektrische Verbindung	formal model of security policy	formales Sicherheitsmodell
electrical resistance and impedance of contacts	elektrischer Widerstand der Kontakte	format character	Formatzeichen (<i>unüblich</i>)
electromagnetic fields	elektromagnetische Felder	functionality	Funktionalität
electronic cash	Electronic Cash	generation	Erzeugung, Generierung
electronic purse	elektronische Börse	genuine card holder	rechtmäßiger Kartenbesitzer
electronic purse	elektronische Geldbörse	ground	Masse
electronic signal	elektronisches Signal	guardtime	Schutzzeit
electronic value	elektronischer Wert	guilloche pattern	Guillochen
electrostatic discharge (ESD)	elektrostatische Entladung	half-duplex	halbduplex
elementary file (EF)	Elementary File ★	hard mask	Hardmaske
elementary time unit (etu)	elementare Zeiteinheit (<i>unüblich</i>) ★	hardware security feature	Hardware Sicherheitsmerkmal
embedded	eingebettet	hash algorithm	Hash-Algorithmus
embedded coil	verlegte Spule	hash function	Hash-Funktion
embossing	Hochprägung, Prägen	hash value	Hash-Wert
embossing relief	Höhe der Prägung	health insurance card	Krankenversicherungskarte (KVK)
height of character		heat dissipation	Wärmeableitung
encipherment module	Verschlüsselungsmodul	heat resistance	Wärmebeständigkeit
end-to-end protocol	Ende-zu-Ende-Protokoll	heat-sealing method	Heißsiegel-Verfahren
entitled authority	berechtigte Instanz	height and surface	Höhe und Oberflächenprofil des Magnetstreifens
equivalence	Entsprechung	profile of the magnetic stripe	
erroneous block	fehlerhafter Block	high impedance state	hochohmiger Zustand
error detection	Fehlererkennung	high state	High-Zustand
error detection code (EDC)	Fehlererkennungscode	I/O	E/A (<i>unüblich</i>) ★
error handling	Fehlerbehandlung	I/O contacts	E-/A-Kontakte (<i>unüblich</i>), I/O-Kontakte
error recovery	Fehlerbehebung	ID card	Ausweiskarte
etched coil	geätzte Spule	identification card	Ausweiskarte
even parity	gerade Parität	identification number	Identifikationsnummer
exchange rate	Wechselkurs	identification systems	Ausweissysteme
expiry date	Ablaufdatum, Verfallsdatum	identifier	Bezeichner
export license	Ausfuhrgenehmigung	idle state	Grundzustand
export regulation	Ausfuhrbeschränkung	IEP balance	IEP-Saldo
extension	Erweiterung	IEP issuing center	IEP-Ausgabezentrum
external application	externe Anwendung	implementation	Implementierung, Realisierung
external authenticate	externe Authentisierung	increment	hochzählen
external disturbance	äußere Störung	initial character	Anfangszeichen ★
externally initiated	extern eingeleitet	initial state	Ausgangszustand
extra guardtime	zusätzliche Schutzzeit	initial waiting time	Anfangswartezeit ★
failure	Ausfall	initialisation	Initialisierung
falling edge	fallende Flanke	injection moulding	Spritzgießen
false block	falscher Block	input capacitance	Eingangskapazität
file	Datei	input current	Eingangsstrom
file identifier (FID)	Dateibezeichner (<i>unüblich</i>) ★	input voltage	Eingangsspannung
file management	Dateimanagement	insert card	Karte einführen
finite state machine	endlicher Automat	insertion forces	Einsteckkräfte
fixed key	feststehender Schlüssel		
flammability	Entflammbarkeit		
float	Bodensatz		

inspection planning	Prüfplanung	link	Verbindung
inspection specification	Prüfspezifikation	linked list	verknüpfte Liste
instruction byte (INS)	Instruction Byte (INS) ★	load	Last
integer	ganzzahlig	load agent	Aufladebevollmächtigter
integrated chip card (ICC)	Chipkarte	load device	Aufladeeinrichtung
integrity	Integrität	load device applicati- on (LDA)	Aufladeeinrichtungs- anwendung
interchange	Austausch	load log	Aufladeprotokoll
interface	Schnittstelle	load SAM (LSAM)	Auflade-SAM
interface device	Interface Device	location of contacts	Lage der Kontakte
(IFD)	(IFD) ★	locked state	Sperrzustand
interindustry com- mands	branchenübergreifende Kommandos	logging	Protokollierung
interindustry message	branchenübergreifende Nachricht	logical AND	logisches UND
internal reader	Einschubleser (<i>unüblich</i>)	logical channels	logische Kanäle
Intersector electronic purse (IEP)	branchenübergreifende elektronische Geldbörse	logical complement	logisches Komplement
intersector environ- ment	branchenübergreifende Umgebung	logical OR	logisches ODER
invisible UV ink	unsichtbare UV-Farbe	logical resources	logische Ressourcen
irreversible	nicht umkehrbar	loss of synchronisa- tion	Synchronisierungsverlust
issuer	Herausgeber	low level	low Pegel
issuer identification number	Identifikationsnummer des Herausgebers	low state	low Zustand
		loyalty card	Kundenkarte
kernel of operating system	Betriebssystemkern	MAC computation	MAC-Berechnung
key	Schlüssel	MAC verification	MAC-Überprüfung
key certification	Schlüsselzertifizierung	machine-readable card	maschinenlesbare Karte
key diversification	Schlüsseldiversifikation	magnetic strip	Magnetstreifen
key fault presentation counter (KFPC)	Fehlbedienungs-zähler (FBZ)	magnetic strip card	Magnetstreifenkarte
key file version	Version der Schlüsseldatei	magstripe	Magnetstreifen
key identifier	Schlüsselbezeichner	magstripe card	Magnetstreifenkarte
key layout	Tastenanordnung	malfunction	Fehlfunktion, Störung
key management	Schlüsselmanagement	mandatory	obligatorisch
key operation	Tastaturbedienung	mandatory file	obligatorische Datei
key separation	Schlüsselaufteilung	mask programming	Maskenprogrammierung
key space	Schlüsselraum	master file (MF)	Master File (MF) ★
keyboard	Tastatur	master key	Hauptschlüssel
keypad	Tastenblock	memory card	Speicherkarte
laser engraving	Lasergravur	memory range	Speicherbereich
least significant bit (lsb)	niedrigstwertiges Bit	memory space	Speicherplatz
level	Ebene	message	Nachricht
level of rigour	Grad der Genauigkeit	message authenticati- on code (MAC)	Datensicherungscode (<i>unüblich</i>) ★
life cycle	Lebenszyklus	Message Authentica- tion Code (MAC)	Nachrichtenbegläubi- gungscode (<i>unüblich</i>) ★
lifekey	Echtschlüssel	message exchange	Nachrichtenaustausch
light transmittance	Lichtdurchlässigkeit	message recovery	Nachrichtenrückgewin- nung, Rückgewinnung der Nachricht
limit value	Grenzwert	microprint	Mikroschrift
linear feed shift regis- ter (LFSR)	linear rückgekoppeltes Schieberegister	mobile telephone	Handy, Mobiltelefon
		mode	Modus
		monetary unit	Geldeinheit
		mono layer card	Einlagenkarte
		most significant bit	höchstwertiges Bit
		Most Significant Bit (msb)	höherwertiges Bit

multi layer card	Mehrlagenkarte
multiapplication	Multiapplication-
smart card	Chipkarte
multifunctional	multifunktional
mutual authentication	gegenseitige Authentisierung
net data	Nettodaten
node address (NAD)	Knotenadresse (<i>unüblich</i>) ★
noise signal	Rauschsignal
noise voltage	Störspannung
non-disclosure agreement (NDA)	Vertraulichkeitsvereinbarung
non-repudiation	Nichtabstreitbarkeit
non-volatile memory	nichtflüchtiger Speicher
notation	Schreibweise
odd parity	ungerade Parität
offline	Offline
offline transaction	Offline-Transaktion
offset	Offset
one-way authentication	Einweg-Authentisierung
one-way function	Einwegfunktion
online	Online
online transaction	Online-Transaktion
open system	offenes System
operating system	Betriebssystem
operating system manufacturer	Betriebssystemhersteller
operator	Bediener
optically variable ink	optisch variable Druckfarbe
originator	Urheber
output	Ausgang
overlay film	Deckfolie
padding	auffüllen
parity	Parität
parity bit	Paritätsbit
parity check	Paritätsprüfung
participant	Teilnehmer
password	Passwort
password verification	Passwortprüfung
pay before	zahle vorher
pay later	zahle später
pay now	zahle jetzt
payment terminal	Zahlungsverkehrsterminal
pay-phone	Münzfernsprecher
personal identification number (PIN)	persönliche Identifikationsnummer (PIN)
personal unblocking key (PUK)	Personal Unblocking Key (PUK) ★
personalisation	Personalisierung
personalisation center	Personalisierungszentrum

personalisation scheme	Personalisierungsschema
physical attack	physikalischer Angriff
PIN entry	PIN-Eingabe
PIN mailer	PIN-Brief
PIN pad	PIN-Pad, PIN-Tastatur
plaintext	Klartext
point of sale (POS)	Point of Sale (POS) ★
post-paid	nachträglich bezahlt
power consumption	Stromaufnahme
power supply	Stromversorgung
power supply voltage	Versorgungsspannung
prefix	Präfix
pre-operational	vor Inbetriebnahme
prepaid	vorausbezahlt
pre-payment application	Anwendung für Vorauszahlungsfunktion
pre-personalisation	Vorpersonalisierung
prime number	Primzahl
printed coil	gedruckte Spule
probe	Meßfühler
procedure	Verfahren
product liability	Produkthaftung
product specification	Produktspezifikation
programming current	Programmierstrom
programming voltage	Programmierspannung
proof	Nachweis
proprietary	eigene, proprietäre
protected	geschützt
protected mode	geschützter Modus
protocol control byte (PCB)	Protokollauswahlbyte (<i>unüblich</i>) ★
protocol layer	Protokollschicht
protocol type selection (PTS)	Protokollauswahl (<i>unüblich</i>) ★
public exponent	öffentlicher Exponent
public key	öffentlicher Schlüssel
public key certificate	Zertifikat mit Hilfe eines öffentlichen Schlüssels (<i>unüblich</i>) ★
public modulus	öffentlicher Modulo
pulse duration	Impulslänge
pulse generator	Impulsgenerator
purchase cancellation	Zahlungsabbruch
purchase device	Zahlungseinrichtung
purchase device application (PDA)	Zahlungseinrichtungsanwendung
purchase log	Zahlungsprotokoll
purchase SAM	Zahlungs-SAM
purse holder	Börseninhaber
purse provider	Börsenanbieter
purse provider SAM (PPSAM)	Börsenanbieter-SAM
pushing force	Aufdrückkraft
quality assurance (QA)	Qualitätssicherung (QS)

quality record	Qualitätsaufzeichnung	security level	Sicherheitsniveau
rainbow print	Irindruck	security mechanism	Sicherheitsmechanismus
random number	Zufallszahl	security model	Sicherheitsmodell
(RND)		security policy	Sicherheitspolitik
read only memory	nur Lesespeicher	security print	Sicherheitsdruck
read protection	Leseschutz	security procedure	Sicherheitsprozedur
real-time	Echtzeit	security protocol	Sicherheitsprotokoll
receiver	Empfänger	security scheme	Sicherheitsstruktur
recognition	Erkennung	security targets	Sicherheitsvorgaben
record	Datensatz (<i>unüblich</i>), Record	security thread	Sicherheitsfaden
record identifier	Datensatzbezeichner	seed number	Initialzahl
(RID)	(<i>unüblich</i>) ★	send sequence	Sendefolgezähler
recording technique	Aufzeichnungstechnik	counter (SSC)	
recovery function	Wiederherstellungs- funktion	sensitive data	schutzwürdige Daten
recovery procedure	Wiederherstellungs- prozedur	sequence control	Ablaufkontrolle
reference	Verweis	serial port	serielle Schnittstelle
related key	zugehöriger Schlüssel	service provider	Leistungsanbieter
reliability	Zuverlässigkeit	session	Sitzung
reloadable card	wiederaufladbare Karte	session key	Sitzungsschlüssel
remainder theorem	Restklassensatz	set	Menge
repetition request	Wiederholanforderung	shadow account	Schattenkonto
replay attacks	Wiederholungsangriffe	short message	Kurznachricht
request	Anforderung	signal amplitude	Signalamplitude
reserved for future use (RFU)	für spätere Verwendung freigehalten (<i>unüblich</i>) ★	signature	Signatur
reset	Rücksetzung (<i>unüblich</i>) ★	signature panel	Unterschriftsstreifen
resistance to chemi- cals	Chemikalienbeständigkeit	signed acknowledge- ment	signierte Quittung
response	Antwort	signer	Unterzeichner
resynchronisation	Resynchronisation	singler	Vereinzelner
retry counter (RC)	Fehlbedienungs-zähler (FBZ)	smart card	Chipkarte
reversible	umkehrbar	smart card application	Chipkarten-Anwendung
revocation list	Sperrliste	smart card develop- ment	Chipkarten-Entwicklung
rise time	Anstiegszeit	smart card manufac- turer	Chipkarten-Hersteller
rising edge	ansteigende Flanke	smart card microcon- troller	Chipkarten- Mikrocontroller
risk management	Risikomanagement	smart card operating system	Chipkarten- Betriebssystem
root	Stammverzeichnis	smart card query lan- guage (SCQL)	Smart Card Query Language (SCQL) ★
root directory	Wurzelverzeichnis	smart card technology	Chipkarten-Technologie
routine	Routine	soft mask	Softmaske
secret key	geheimer Schlüssel	source address (SAD)	Quelladresse
secret key algorithm	Algorithmus mit gehei- mem Schlüssel (<i>unüblich</i>) ★	stack	Stapelspeicher
secure application module (SAM)	Anwendungssicherheits- modul	start rising edge	Startvorderflanke
secure module	geschütztes Modul	state	Zustand
security architecture	Sicherheitsarchitek- tur	state machine	Zustandsautomat
security attribute	Sicherheitsattribut	state transition	Zustandsübergang
security feature	Sicherheitsmerkmal	statement coverage	Befehlsabdeckung
		static electricity	Statische Elektrizität
		status response	Statusantwort
		status word (SW)	Status Word (SW) ★
		stoplist	Sperrliste
		supply current	Versorgungsstrom
		surface profile of contacts	Oberflächenprofil der Kontakte

surface roughness of the magnetic stripe	Oberflächenrauheit des Magnetstreifens	untrusted monitor	nicht zertifiziertes Kontrollgerät
symmetric algorithm	symmetrischer Algorithmus	update rate	Änderungshäufigkeit
tag	Kennzeichen	user card	Anwenderkarte
terminal	Datenendgerät (<i>unüblich</i>), Endgerät (<i>unüblich</i>)	user key	Benutzerschlüssel
test mode	Testmodus	user mode	Benutzermodus
test zone	Abtastzone	user's requirement specification	Lastenheft
testkey	Testschlüssel	value	Wert
time stamp service	Zeitstempeldienst	value added service (VAS)	Mehrwertdienst
total	Endsumme	vending machine	Warenausgabeautomat
traceability	Rückverfolgbarkeit	verification	Überprüfung
track	Spur	vibrations	Schwingungen
trailer	Nachsatz	volatile memory	flüchtiger Speicher
transaction process	Transaktionsvorgang	voltage watchdog	Spannungsüberwachung
transaction processing	Transaktionsverarbeitung	wafer number	Scheibenummer (<i>unüblich</i>) ★
transaction record	Transaktionsdatensatz	waiting time extension (WTX)	Wartezeitverlängerung
transient mode	Übergangsbetrieb	warping	Wölbung
transition zone	Übergangszone	waterfall model	Wasserfallmodell
transmission error	Übertragungsfehler	wear of magnetic stripe	Abnutzung des Magnetstreifens
transmission method	Übertragungsverfahren	wire-bonding	Draht-Bonding
transmission mode	Sendebetrieb, Übertragungsbetrieb (<i>unüblich</i>)	withdrawing forces	Herausziehungskräfte
transmission protocol	Übertragungsprotokoll	working elementary file (WEF)	Working Elementary File (WEF) ★
transmitter	Sender	wound coil	gewickelte Spule
transport entity	Transportinstanz	write cycle	Schreibzyklus
transport layer	Transportschicht	write once read multiple (WORM)	Write Once Read Multiple (WORM) ★
trap door	Falltür		
trusted	Vertrauenswürdigkeit		
trusted device	vertrauenswürdiges Gerät		
unauthorised access	unberechtigter Zugriff		
unique	einzigartig		

16.3 Weiterführende Literatur

Der inhaltliche Fokus des Handbuchs der Chipkarten liegt im Gebiet von Chipkarten und deren Anwendungen. Es gibt jedoch noch eine Reihe von Disziplinen, die starken Einfluss auf Chipkarten und deren Weiterentwicklung haben, jedoch ihrerseits wiederum Spezialgebiete mit eigener Fachliteratur sind. Die Autoren des Handbuchs der Chipkarten möchten den Fokus auf ihr Spezialgebiet beibehalten und nicht in größerem Umfang Abhandlungen über assoziierte Fachgebiete verfassen, da dies den Rahmen dieses Buches sprengen würde. Für alle Leser, die ihr Wissen auf diesen verknüpften Themengebieten vertiefen möchten, haben wir die folgende kurze Literaturliste erstellt.

Themengebiet	Literatur
Betriebssysteme	[Tanenbaum 02]
Fertigung von Chipkarten	[Haghiri 02]
Java als Programmiersprache	[Arnold 00]
Kryptografie	[Menezes 97], [Schneier 96]
RFID	[Finkenzeller 02]
Sicherheit von Komponenten und Systemen	[Anderson 01]
Softwareentwicklung	[Balzert 98]
Softwareentwicklung für Java Card	[Chen 00]

16.4 Literatur

Die nachfolgenden Veröffentlichungen sind mit dem ersten Kriterium nach dem Familiennamen des Autors und mit dem zweiten in aufsteigender Reihenfolge nach dem Datum sortiert. Veröffentlichungen in Newsgroups oder Diskussionsforen des Internets sind mit der Quellenangabe „Internet“ gekennzeichnet.

[Anderson 01] Ross J. Anderson: *Security Engineering*, John Wiley & Sons, Chichester 2001

[Anderson 92] Ross J. Anderson: *Automatic Teller Machines*, Internet, Dezember 1992

[Anderson 96 a] Ross J. Anderson, Markus G. Kuhn: *Improved Differential Fault Analysis*, Internet, November 1996

[Anderson 96 b] Ross J. Anderson, Markus G. Kuhn: *Tamper Resistance – a Cautionary Note*, USENIX Workshop, November 1996

[Arnold 00] Ken Arnold, James Gosling, David Holmes: *The Java Programming Language*, 3. Auflage, Addison Wesley, Boston 2000

[Balzert 98] Helmut Balzert: *Lehrbuch der Software-Technik, Band 2*, 2. Auflage, Spektrum Akademischer Verlag, Heidelberg 1998

[Bellare 95 a] Mihir Bellare, Juan Garay, Ralf Hulse, Amir Herzberg, Hugo Krawczyk, Michael Steiner, Gene Tsudik, Michael Waidner: *iKP – A Family of Secure Electronic Payment Protocols*, Internet, 1995

- [Bellare 95 b] Mihir Bellare, Philip Rogaway: *Optimal Asymmetric Encryption – How to encrypt with RSA*, Internet, 1995
- [Bellare 96] Mihir Bellare, Philip Rogaway: *The Exact Security of Digital Signatures – How to Sign with RSA and Rabin*, Internet, 1996
- [Beutelsbacher 93] Albrecht Beutelsbacher: *Kryptologie*, 3. Auflage, Vieweg Verlag, Braunschweig 1993
- [Beutelsbacher 96] Albrecht Beutelsbacher, Jörg Schwenk, Klaus-Dieter Wolfenstetter: *Moderne Verfahren der Kryptografie*, Vieweg Verlag, Braunschweig 1996
- [Biham 91] Eli Biham, Adi Shamir: *Differential Cryptanalysis of DES-like Cryptosystems*, Journal of Cryptology, Vol. 4, No. 1, 1991
- [Biham 93] Eli Biham, Adi Shamir: *Differential Cryptanalysis of the Data Encryption Standard*, Springer Verlag, New York 1993
- [Biham 96] Eli Biham, Adi Shamir: *A new cryptanalytic attack on DES*, Internet, 1996
- [BIS 96] Bank for International Settlements: *Security of Electronic Money – Report by the Committee on Payment and Settlement Systems and the Group of Computer Experts of the central banks of the Group on Ten countries*, Basel, August 1996
- [Blumtritt 97] Oskar Blumtritt: *Nachrichtentechnik*, 2. Auflage, München, Deutsches Museum, 1997
- [Boehm 81] Barry W. Boehm: *Software Engineering Economics*, Prentice Hall, Upper Saddle River, New Jersey 1981
- [Boneh 96] Dan Boneh, Richard A. DeMillo, Richard J. Lipton, *On the Importance of Checking Computations*, Math and Cryptography Research Group, Bellcore, 1996
- [Boneh 96] Dan Boneh, Richard A. DeMillo, Richard J. Lipton: *On the Importance of Checking Computations*, Internet, 1996
- [Bronstein 96] I. N. Bronstein, K. A. Semendjajew: *Taschenbuch der Mathematik*, 7. Auflage, B. G. Teubner Verlagsgesellschaft Leipzig 1997
- [Buchmann 96] Johannes Buchmann: *Faktorisierung großer Zahlen*, Spektrum der Wissenschaft, September 1996
- [Chen 00] Zhiqun Chen: *Java Card Technology for Smart Cards*, Addison Wesley, Boston 2000
- [Dhem 96] J. F. Dhem, D. Veithen, J.-J. Quisquater: *SCALPS: Smart Card Applied to Limited Payment Systems*, UCL Crypto Group Technical Report Series, Université catholique de Louvain, 1996
- [Dictionary of Computing 91] *Dictionary of Computing*, Oxford University Press, Oxford 1991
- [Diffie 76] Whitfield Diffie, Martin E. Hellman: *New Directions in Cryptography*, Internet, 1976
- [Dröschel 99] Wolfgang Dröschel, Manuela Wiemers: *Das V- Modell 97*, Oldenbourg Verlag, München 1999
- [Eberspächer 00] Jörg Eberspächer, Hans-Jörg Vogel, Christian Bettstetter: *GSM – Global System for Mobile Communication*, B.G. Teubner Verlag, Stuttgart 2000
- [EFF 98] Electronic Frontier Foundation: *Frequently Asked Questions (FAQ) about the Electronic Frontier Foundation's "DES Cracker" Machine*, Internet, 1998

- [EG 91] Europäische Gemeinschaften – Kommission: *Kriterien für die Bewertung der Sicherheit von Systemen der Informationstechnik (ITSEC)*, Version 1.2, Juni 1991
- [EG 98] Europäische Gemeinschaften: *Verordnung Nr. 2135/98 des Rates der Europäischen Gemeinschaften vom 24. September 1998 zur Änderung der Verordnung Nr. 3821/85 über das Kontrollgerät im Straßenverkehr und der Richtlinie 88/599/EWG über die Anwendung der Verordnungen Nr. 3820/85 und Nr. 3821/85*
- [Fenton 96] Norman E. Fenton, Shari Lawrence Pfleeger: *Software Metrics*, Thomson Computer Press, London 1996
- [Finkenzeller 02] Klaus Finkenzeller: *RFID-Handbuch*, 3. Auflage, Carl Hanser Verlag, München, Wien 2002
- [Franz 98] Michael Franz: *Java – Anmerkungen eines Wirth-Schülers*, Informatik Spektrum, Springer Verlag, Berlin 1998
- [Freeman 97] Adam Freeman, Darrel Ince: *Active Java – Object Oriented Programming for the World Wide Web*, Addison-Wesley, Reading 1997
- [Fumy 94] Walter Fumy, Hans Peter Ries: *Kryptographie*, 2. Auflage, R. Oldenbourg Verlag, München, Wien, 1994
- [Gentz 97] Wolfgang Gentz: *Die elektronische Geldbörse in Deutschland*, Diplomarbeit an der Fachhochschule München, München 1997
- [Glade 95] Albert Glade, Helmut Reimer, Bruno Struif: *Digitale Signatur*, Vieweg Verlag, Braunschweig 1995
- [Gora 98] Walter Gora: *ASN.1 – Abstract Syntax Notation One*, 3. Auflage 1998, Fossil Verlag, Köln
- [Gosling 95] James Gosling, Henry McGilton: *The Java Language Environment – A White Paper*, Sun Microsystems, USA, 1995
- [Grün 96] Herbert Grün: *Card Manufacturing Materials and Environmental Responsibility*, Presentation at CardTech/SecurTech, Atlanta, May 1996
- [GSM 95] Proceedings of the Seminar for Latin America Decision Makers by GSM MoU Association and ECTEL: *Personal Communication Services based on the GSM Standard*, Buenos Aires 1995
- [Guthery 02] Scott B. Guthery, Mary J. Cronin: *Mobile Application Development with SMS and the SIM Toolkit*, McGraw-Hill, New York, 2002
- [Gutmann 96] Peter Gutmann: *Secure Deletion of Data from Magnetic and Solid-State Memory*, USENIX Konferenz, San Jose 1996
- [Gutmann 98 a] Peter Gutmann: *Software Generation of Practically Strong Random Numbers*, Internet, 1998
- [Gutmann 98 b] Peter Gutmann: *X.509 Style Guide*, Internet, 1998
- [Haghir 02] Yahya Haghir, Thomas Tarantino: *Smart Card Manufacturing: A Practical Guide*, John Wiley & Sons, Chichester 2002
- [Hassler 02] Vesna Hassler, Martin Manninger, Mikhail Gordeev, Christoph Muller: *Java Card for E-Payment Applications*, Artech House, London 2002
- [Hellmann 79] Martin E. Hellmann: *The Mathematics of Public-Key cryptography*, Scientific American, August 1979

- [Hillebrand 2002] Friedhelm Hillebrand (Editor): *GSM and UMTS*, John Wiley & Sons, Chichester 2002
- [IC Protection 97] *Common Criteria for IT Security Evaluation Protection Profile – Smartcard Integrated Circuit Protection Profile*, Internet, 1997
- [Isselhorst 97] Hartmut Isselhorst: *Betreiberorientierte Sicherheitsanforderungen für Chipkarten-Anwendungen*, Card-Forum, Lüneburg 1997
- [Jones 91]. C. Jones: *Applied software measurement*, McGraw-Hill, New York, 1991
- [Jun 99] Benjamin Jun, Paul Kocher: *The Intel Random Number Generator*, Internet 1999
- [Kaliski 93] Burton S. Kaliski Jr.: *A Layman's Guide to a Subset of ASN.1, BER and DER*, RSA Laboratories Technical Note, Internet, 1993
- [Kaliski 96] Burton S. Kaliski Jr.: *Timing Attacks on Cryptosystems*, RSA Laboratories, Redwood City 1996
- [Karten 97] Zeitschrift Karten: *Zur Sicherheit der ec-Karte PIN: Das Urteil des OLG Hamm*, Fritz Knapp Verlag, Frankfurt, August 1997
- [Knuth 97] Donald Ervin Knuth: *The Art of Computer Programming, Volume 2, Seminumerical Algorithms*, 3. Auflage, Addison Wesley Longman, Reading, Massachusetts 1997
- [Kocher 95] Paul C. Kocher: *Timing Attacks on Implementations of Diffie-Hellmann, RSA, DSS, and Other Systems*, Internet, 1995
- [Kocher 98 a] Paul C. Kocher, Joshua Jaffe, Benjamin Jun: *Introduction to Differential Power Analysis and Related Attacks*, Internet, 1998
- [Kocher 98b] Paul C. Kocher, Joshua Jaffe, Benjamin Jun: *Differential Power Analysis: Leaking Secrets*, Internet, 1998
- [Kömmmerling 99] Oliver Kömmmerling, Markus G. Kuhn, *Design Principles for Tamper-Resistant Smartcard Processors*, USENIX Workshop on Smartcard Technology, Chicago USA, 10–11 Mai 1999
- [Kuhn 97] Markus G. Kuhn: *Probability Theory for Pickpockets – ec-PIN Guessing*, COAST Laboratory, Purdue University, West Lafayette, Indiana 1997
- [Kuhn] Markus G. Kuhn: *Attacks on Pay-TV Access Control Systems*, University of Cambridge, Internet, Jahr unbekannt
- [Lamla 00] Michael Lamla: *Hardware Attacks on Smart Cards – Overview*, Eurosmart Security Conference, Marseille, 13–15 Juni 2000
- [Leiberich 99] Otto Leiberich: *Vom diplomatischen Code zur Falltürfunktion*, Spektrum der Wissenschaft, Juni 1999
- [Lender 96] Friedwart Lender: *Production, Personalisation and Mailing of Smart Cards – A Survey*, Smart Card Technologies and Applications Workshop, Berlin, November 1996
- [Levy 99] Steven Levy: *The open secret*, Wired, April 1999
- [Lindholm 97] Tim Lindholm, Frank Yellin: *Java – Die Spezifikation der virtuellen Maschine*, Addison-Wesley, Reading 1997
- [Massey 88] James L. Massey: *Proceedings of the IEEE, An Introduction to Contemporary Cryptology*, S. 533–549, IEEE, Vol. 76, No. 5, 1988
- [Massey 97] James L. Massey: *Cryptography, Fundamentals and Applications*, 1997

- [Meister 95] Giesela Meister, Eric Johnson: *Schlüsselmanagement und Sicherheitsprotokolle gemäß ISO/SC 27 – Standards in Smart Card – Umgebungen* in: Albert Glade, Helmut Reimer, Bruno Struif: *Digitale Signatur*, Vieweg Verlag, Braunschweig 1995
- [Menezes 93] Alfred J. Menezes: *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishing, Boston 1993
- [Menezes 97] Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone: *Handbook of Applied Cryptography*, CRC Press, Boca Raton 1997
- [Merkle 81] Ralph C. Merkle, Martin E. Hellman: *On the Security of Multiple Encryption*, Internet, 1981
- [Messerges 99] Thomas S. Messerges, Ezzy A. Dabbish, Robert H. Sloan: *Investigations of Power Analysis Attacks on Smartcards*, USENIX Workshop on Smartcard Technology, Chicago USA, 10–11 Mai 1999
- [Meyer 82] Carl H. Meyer, Stephen M. Matyas: *Cryptography*, John Wiley & Sons, New York 1982
- [Meyer 96] Carsten Meyer: *Nur Peanuts – Der Risikofaktor Magnetkarte*, c't, Juli 1996
- [Montenegro 99] Sergio Montenegro: *Sichere und fehlertolerante Steuerungen*, Carl Hanser Verlag, München, Wien 1999
- [Moore 02] Simon Moore, Ross Anderson, Paul Cunningham, Robert Mullins, George Tayler: *Improving Smart Card Security using Self-timed Circuits*, Internet, Mai 2002
- [Müller-Maguhn 97 a] Andy Müller-Maguhn: „Sicherheit“ von EC-Karten, Die Datenschleuder, Ausgabe 53, 1997
- [Müller-Maguhn 97 b] Andy Müller-Maguhn: *EC-Karten Unsicherheit*, Die Datenschleuder, Ausgabe 59, 1997
- [Myers 95] Glenford J. Myers: *The Art of Software Testing*, 5. Auflage, John Wiley & Sons, New York, 1995
- [Nebelung 96] Brigitte Nebelung: *Das Geldbörsen-Konzept der ec-Karte mit Chip*, debis Systemhaus, Bonn 1996
- [Nechvatal 00] James Nechvatal, Elaine Barker, Lawrence Bassham, William Burr, Morris Dworkin, James Foti, Edward Roback, NIST, 2000, *Report on the Development of the Advanced Encryption Standard (AES)*, Internet
- [Odlyzko 95] Andrew. M. Odlyzko: *The future of integer factorization*, AT & T Bell Laboratories, 1995
- [Otto 82] Siegfried Otto: *Echt oder falsch? Die maschinelle Echtheitserkennung*, Betriebswirtschaftliche Blätter, Heft 2, Februar 1982
- [Peyret 97] Patrice Peyret: *Which Smart Card technologies will you need to ride the Information Highway safely?*, Gemplus, 1997
- [Pfaffenberger 97] Bryan Pfaffenberger: *Dictionary of Computer Terms*, Simon & Schuster Macmillan Company, New York 1997
- [Piller 96] Ernst Piller: *Die „ideale“ Geldbörse für Europa*, Card-Forum, Lüneburg 1996
- [Pomerance 84] C. Pomerance: *The Quadratic Sieve Factoring Algorithm*, Advances in Cryptology – Eurocrypt 84
- [Press 92] William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery: *Numerical Recipes in C – The Art of Scientific Computing*, 2. Auflage, Cambridge University Press, Cambridge 1992

- [Rivest 78] Ronald L. Rivest, Adi Shamir, Leonard Adleman: *Method for obtaining Digital Signatures and Public-Key Cryptosystems*, Internet, 1976
- [Robertson 96] James Robertson, Suzanne Robertson: *Vollständige Systemanalyse*, Carl Hanser Verlag, München, Wien 1996
- [Robshaw 95] Security Estimates for 512-bit RSA, M. J. B. Robshaw, RSA Laboratories, Juni 1995
- [Rother 98 a] Stefan Rother: *Prüfung von Chipkarten-Sicherheit*, Card-Forum, Lüneburg 1998
- [Rother 98 b] Stefan Rother: *Prüfung von Chipkarten-Sicherheit* im Tagungsband Chipkarten, Vieweg Verlag, Braunschweig 1998
- [RSA 97] RSA Data Security Inc.: *DES Crack Fact Sheet*, Internet, 1997
- [Scherzer 00] Helmut Scherzer: *Chipkarten-Betriebssysteme - Gefahrenpotentiale und Sicherheitsmechanismen*, Forum IT-Sicherheit Smartcards, 14. März 2000
- [Schief 87] Rudolf Schief: *Einführung in die Mikroprozessoren und Mikrocomputer*, 10. Auflage, Attempto Verlag, Tübingen 1987
- [Schindler 97] Werner Schindler: *Wie sicher ist die PIN?*, Vortrag bei der Tagung „Kreditkartenkriminalität“, Heppenheim, Oktober 1997
- [Schlumberger 97] Schlumberger: *Cyberflex – Programmers Guide*, Version 6d, April 1997
- [Schneier 96] Bruce Schneier: *Applied Cryptography*, 2. Auflage, John Wiley & Sons, New York 1996
- [Schneier 99] Bruce Schneier: *Attack Trees – Modeling security threats*, Dr. Dobbs's Journal, December 1999
- [Sedgewick 97] Robert Sedgewick: *Algorithmen*, 3. Auflage, Addison-Wesley, Bonn München Reading 1997
- [SigG 01] Gesetz über Rahmenbedingungen für elektronische Signaturen, 22. Mai 2001
- [Silverman 97] Robert D. Silverman: *Fast Generation of Random, Strong RSA Primes*, RSA Laboratories Crypto Byte, Internet, 1997
- [Simmons 92] Gustavus J. Simmons (Editor): *Contemporary Cryptology*, New York: IEEE Press, 1992
- [Simmons 93] Gustavus J. Simmons: *The Subliminal Channels in the U.S. Digital Signature Algorithm*, Proceedings of Symposium on State and progress of Research in Cryptography, Rom 1993
- [Skorobogatov 02] Sergei Skorobogatov, Ross Anderson: *Optical Fault Induction Attacks*, Internet, Mai 2002
- [Sommerville 90] Ian Sommerville: *Software Engineering*, Addison-Wesley, Wokingham 1990
- [Steele 01] Raymond Steele, Chin-Chun Lee, Peter Gould: *GSM, cdmaOne and 3G Systems*, John Wiley & Sons, Chichester 2001
- [Stix 96] Gary Stix: *Herausforderung „Komma eins“*, Spektrum der Wissenschaft, Februar 1996
- [Stocker 98] Thomas Stocker: *Java für Chipkarten* im Tagungsband Chipkarten, Vieweg Verlag, Braunschweig 1998
- [Tanenbaum 02] Andrew S. Tanenbaum: *Moderne Betriebssysteme*, 3. Auflage, Addison-Wesley Longman, Reading, Massachusetts, 2002

- [Thaller 93] Georg Erwin Thaller: *Qualitätsoptimierung der Software- Entwicklung. Das Capability Maturity Model (CMM)*, Vieweg Verlag, Braunschweig 1993
- [Tietze 93] Ulrich Tietze, Christoph Schenk: *Halbleiter-Schaltungstechnik*, 10. Auflage, Springer Verlag, Berlin 1993
- [Vedder 97] Klaus Vedder, Franz Weikmann: *Smart Cards – Requirements, Properties and Applications*, ESAT-COSIC Course, Katholische Universität Leuven, 1997
- [Walke 00] Bernhard Walke: *Mobilfunknetze und ihre Protokolle, Band 1 Grundlagen, GSM, UMTS und andere zellulare Mobilfunknetze*, B. G. Teubner Verlag, Stuttgart 2000
- [Walke 00] Bernhard Walke: *Mobilfunknetze und ihre Protokolle, Band 2 Bündelfunk, schnurlose Telefonsysteme, W-ATM, HIPERLAN, Satellitenfunk, UPT*, B. G. Teubner Verlag, Stuttgart 2000
- [Weikmann 92] Franz Weikmann: *Smart Card-Chips – Technik und weitere Perspektiven*, Der GMD-Spiegel 1'92, Gesellschaft für Mathematik und Datenverarbeitung, Sankt Augustin 1992
- [Weikmann 98] Franz Weikmann, Klaus Vedder: *Smart Cards Requirements, Properties and Applications* in: Tagungsband Chipkarten, Vieweg Verlag, Braunschweig 1998
- [Wiener 93] Michael J. Wiener: *Efficient DES Key Search*, Crypto 93, Santa Barbara 1993
- [Yellin 96] Frank Yellin: *Low Level Security in Java*, Internet, 1996
- [Zieschang 98] Thilo Zieschang: *Differentielle Fehleranalyse und Sicherheit von Chipkarten*, Internet, 1998

16.5 Kommentiertes Normen- und Spezifikationsverzeichnis

Der folgende Abschnitt enthält ein ausführlich kommentiertes Verzeichnis der für Karten mit und ohne Chip relevanten Normen und Industriestandards. Das Verzeichnis nimmt vor allem Rücksicht auf internationale und nicht so sehr auf lokale, länderspezifische Normen. Es sind sowohl Normen von offiziellen Normungsorganisationen (z. B.: ANSI, CEN, ETSI, ISO) als auch kartenrelevante Quasi-Normen wie beispielsweise der EMV-Standard oder Internet-RFCs aufgeführt.

Zusätzlich zum kommentierten Verzeichnis zeigt die Tabelle 16.1 eine Übersicht von unter Umständen hilfreichen Aufstellungen, Überblicke und Quellen für themenspezifische Normen und Standards. Vor allem die Industriestandards sind oft kostenlos via WWW erhältlich, was bei Normen, die von Normungsinstituten herausgegeben werden, in der Regel leider nicht der Fall ist.

Die für Chipkarten wichtigsten Normen und Standards sind mit dem Symbol „☒“ gekennzeichnet. Alle Normen und Standards sind nach der herausgebenden Institution und ihrer Nummer, ohne Berücksichtigung von Präfixen (z. B.: „pr“) oder Zustandsbezeichnungen (z. B.: „DIS“) geordnet. Das aufgeführte Datum ist das Erscheinungsdatum der aktuell gültigen Ausgabe.

Tabelle 16.1 Aufstellung der wichtigsten Web-Sever für den Download von Normen und Informationen über Chipkarten-relevante Normen

Normungs-/Standardisierungsinstanz	Web-Server	Bemerkung
ANSI	[ANSI]	---
CEN	[CEN]	---
DIN	[DIN]	---
EMV	[EMVCO]	Die Standards sind kostenlos vom Web-Server downloadbar.
ETSI	[ETSI]	Alle ETSI-Normen (u.a. für GSM und UMTS) sind kostenlos vom Web-Server downloadbar.
FIPS	[NIST]	Alle FIPS-Normen sind kostenlos vom Web-Server downloadbar.
Global Platform	[Global Platform]	Die Standards sind kostenlos vom Web-Server downloadbar.
IEEE	[IEEE]	---
ISO/IEC	[ISO]	---
ITU	[ITU]	---
Java Card Forum		Die Standards sind kostenlos vom Web-Server downloadbar.
RFC	[RFC]	Die Standards sind kostenlos vom Web-Server downloadbar.
RSA Inc.	[RSA]	Die Standards sind kostenlos vom Web-Server downloadbar.
SEIS	[SEIS]	Die Standards sind kostenlos vom Web-Server downloadbar.

Im Folgenden noch einige kurze Anmerkungen zur Bezeichnung der jeweiligen Normen: Ergänzungen zu ISO- und ISO/IEC-Normen befinden sich in der Regel in einem Anhang (*Amendment, Amd.*). Bei der nächsten Revision dieser Norm, die üblicherweise alle fünf Jahre stattfindet, wird dann der Anhang bei Bedarf in die jeweilige Norm aufgenommen. Die revidierte Fassung der Norm unterscheidet sich von der vorherigen Norm dann nur durch eine neue Jahreszahl und eine laufende Nummer zur Auflage. Neue Versionen von CEN-Normen werden in analoger Art wie ISO-Normen bezeichnet. Bei FIPS-Normen ist die Nummer der revidierten Auflage Norm ein Bestandteil der Normungsbezeichnung (z. B. FIPS 140-2). Telekommunikationsnormen von ETSI benutzen eine dreistellige Versionsnummer als das ausschlaggebende Unterscheidungskriterium. Bei Industrie-Standards kommen je nach Herausgeber sowohl Jahreszahlen wie auch Versionsnummern als eindeutiges Merkmal für bestimmte Dokumentenstände zum Einsatz.

ANSI X9.8	Banking – Personal Identification Number Management and Security
– 1 : 1995	Part 1: PIN Protection Principles and Techniques
– 2 : 1995	Part 2: Approved Algorithms for PIN Encipherment
ANSI X9.9 : 1986	Financial Institution Message Authentication
ANSI X9.17 : 1985	Financial Institution Key Management
ANSI X9.19 : 1996	Financial Institution Retail Message Authentication
ANSI X9.30	Public Key cryptography using irreversible algorithms for the financial services industry
– 1 : 1997	Part 1: The Digital Signature Algorithm (DSA)
– 2 : 1997	Part 2: The Secure Hash Algorithm (SHA-1)
ANSI X9.31 : 1998	Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry
ANSI X9.55 : 1997	Public Key Cryptography for the Financial Services Industry: Extensions to Public Key Certificates and Certificate Revocation Lists
ANSI X9.84 : 2001	Biometric Information Management and Security <i>Diese sehr umfangreiche Norm legt für die unterschiedlichsten biometrischen Identifizierungsverfahren die grundlegenden Architekturprinzipien, die Verwendung, Verwaltung und Sicherheitsanforderungen für biometrische Daten fest.</i>
ANSI X3.92 : 1981	Data Encryption Algorithm <i>Diese Norm beschreibt den DES-Algorithmus.</i>
ANSI X3.106 : 1983	American National Standard for Information Systems – Data Encryption Algorithm – Modes of Operation
ANSI / IEEE 829 : 1991	Standard for Software Test Documentation <i>Beschreibung der Vorgehensweise und notwendigen Dokumentation beim Test von Software.</i>
ANSI / IEEE 1008 : 1987	Standard for Software Unit Testing <i>Beschreibung der grundlegenden Vorgehensweisen beim Test von Software.</i>
ANSI / IEEE 1012 : 1992	Software Verification and Validation Plans <i>Festlegung der notwendigen Testaktivitäten und Testpläne für Softwareentwicklungen. Die Grundlage für diese Norm ist das Wasserfallmodell der Softwareproduktion.</i>

CCITT Z.100 : 1993	CCITT Specification and Description Language (SDL)
CEPS, Version 2.1.3 : 2001	Joint Specification for Common Electronic Purse Cards <i>CEPS ist eine wichtige Norm für elektronische Geldbörsen, welche in Zukunft die Grundlage der meisten Börsensysteme in Europa sein wird bzw. schon ist und sich an EN 1546 orientiert.</i>
Common Criteria, Version 2.1 : 1999	identisch mit ISO/IEC 15 408, siehe dort
DIN 9781-10 : 1985	Büro- und Datentechnik; Identifikationskarten aus Kunststoff oder kunststofflaminiertem Werkstoff; Anforderungen an Echtheitsmerkmale <i>Diese sehr kurze Norm definiert die Begriffe im Umfeld von Echtheitsmerkmalen und zeigt generelle Anforderungen an Echtheitsmerkmale auf.</i>
DIN 44 300 – 1 ...9 : 1988	Informationsverarbeitung – Begriffe <i>Definition vieler Begriffe aus der Informationstechnik.</i>
EMV 2000	Integrated Circuit Card Specification for Payment Systems ☒ <i>Dies ist die wichtigste Normenreihe für Chipkarten im Zahlungsverkehr und wird gemeinsam von EMVCo [EMV] veröffentlicht. Die Reihe besteht aus vier Teilen, Bücher genannt, die die Chipkarte und die dazugehörige Debit- oder Kredit-Zahlungsverkehrsanwendung sowie das dazugehörige Terminal festlegen.⁷³</i>
Book 1 Version 4.0 : 2000	Application Independent ICC to Terminal Interface Requirements <i>Dieser Teil enthält die Spezifikation der mechanischen und elektrischen Eigenschaften von Chipkarten und Terminals. Definition der An- und Abschaltsequenz, der Datenübertragung auf elektrischer Ebene, des ATRs sowie der dazugehörigen Datenelemente. Außerdem sind die beiden Übertragungsprotokolle T = 0 und T = 1 sowie APDU-Aufbau, logische Kanäle, einige wenige grundlegende Kartenkommandos und Mechanismen zur Anwendungsselektion spezifiziert.</i>
Book 2 Version 4.0 : 2000	Security and Key Management <i>In diesem Teil sind statische und dynamische Datenauthentisierung, PIN-Verschlüsselung und Secure Messaging beschrieben. Er enthält auch generelle Rahmenbedingungen für das Schlüsselmanagement der Public Keys eines Zahlungsverkehrssystems und Anforderungen für die Terminalsicherheit inklusive dazugehörigem Schlüsselmanagement.</i>
Book 3 Version 4.0 : 2000	Application Specification <i>Dieser Teil der EMV-Spezifikation definiert eine Reihe von notwendigen Kommandos der Chipkarte und Chipkarten-Anwendung für Debit- und Kreditkarten. Des Weiteren Vorgaben für die Transaktionsabläufe. Im Anhang befindet sich unter anderem eine Beschreibung inklusive Codierung aller Datenelemente, Vorgaben zur TLV-Codierung von Daten und generelle Ansätze zur Integration von EMV-Chipkarten in SET-basierte Zahlungsverkehrssysteme.</i>
Book 4 Version 4.0 : 2000	Cardholder, Attendant and Acquirer Interface Requirements <i>In Book 4 sind die obligatorischen und optionalen Anforderungen für ein Terminal, das Chipkarten nach EMV unterstützt, aufgeführt. Dazu gehören denkbare Konfigurationen, funktionale und sicherheitstechnische Anforderungen an das Terminal, mögliche und erlaubte Benutzermeldungen inklusive verwendetem Zeichensatz sowie die Schnittstelle zum Acquirer. Dieser Standard definiert in den Grundzügen auch die Architektur der Terminalsoftware sowie das Modell eines Interpreters im Terminal für ausführbaren Programmcode. Im Anhang befindet sich eine Aufstellung</i>

⁷³ Siehe auch Abschnitt 12.4 „EMV-Anwendung“.

	<i>von Terminal-relevanten Datenelementen und Hinweise zur technischen Gestaltung der Terminals sowie Beispiele für Terminals am Point-of-Sale, an Geldausgabe- und Warenausgabeautomaten.</i>
EN 726	Identification card systems – Telecommunications integrated circuit(s) card and terminals <i>Diese Normenreihe war bis Mitte der 90er Jahre führend hinsichtlich der Beschreibung der Funktionalität von Chipkarten-Betriebssystemen. Sie wurde aber mittlerweile vollständig von der ISO/IEC 7816-Normenreihe, den UICC-Normen und EMV-Standards substituiert, so dass sie keine Bedeutung mehr hat.</i>
– 1 : 1994	Part 1: System overview
– 2 : 1995	Part 2: Security framework
– 3 : 1994	Part 3: Application independent card requirements ☒ <i>Definition von Dateistrukturen, Kommandos, Returncodes, Dateien für allgemein verwendbare Funktionen und der grundlegenden Mechanismen von Chipkarten für Anwendungen im Bereich der Telekommunikation. Diese Norm ist das ETSI-Gegenstück zur ISO/IEC 7816-4 und der korrespondierende Rahmen zur GSM 11.11.</i>
– 4 : 1994	Part 4: Application independent card related terminal requirements
– 5 : 1999	Part 5: Payment methods <i>Definition von verschiedenen Zahlungsmethoden mit dazugehörigen Dateistrukturen, Datenelementen und Abläufen für Chipkarten. Die Zahlungsmethoden sind für Anwendungen im Bereich der Telekommunikation konzipiert.</i>
– 6 : 1995	Part 6: Telecommunication features
– 7 : 1999	Part 7: Security Module
EN 753	Identification card systems – Intersector thin flexible cards
– 1 : 1997	Part 1: General technical specifications
– 2 : 1997	Part 2: Magnetic recording technique
– 3 : 1999	Part 3: Test methods
EN 1038 : 1995	Identification card systems – Telecommunication applications – Integrated circuit(s) card payphone <i>Definition der Grundlagen für die Anwendung von Chipkarten an öffentlichen Kartentelefonen. Die Norm enthält vor allem Verweise auf bestehende Normen und legt die sinnvollen Varianten fest, an welcher Stelle im System sich das Sicherheitsmodul zur Authentisierung der Telefonkarte befinden soll.</i>
prEN 1105 : 1995	Identification card systems – General concepts applying to systems using IC cards in inter-sector environments – Rules for inter-application consistency <i>Definition von grundlegenden Forderungen an eine Chipkarte, um einen anwendungsübergreifenden Einsatz sicherzustellen. Enthält vor allem Verweise auf bestehende Normen sowie diverse Festlegungen für Chipkarten und Terminals.</i>
prEN 1292 : 1995	Additional Test Methods for IC Cards and Interface Devices <i>Definition von Tests für die elektrischen Rahmenwerte und die grundlegende Datenübertragung von Chipkarte und Terminal. Diese Norm ist eine Ergänzung zur ISO/IEC 10 373.</i>

EN 1332	Identification card systems – Man-Machine Interface
– 1 : 1999	Part 1: Design principles and symbols for the user interface
– 2 : 1998	Part 2: Definition of a Tactile Identifier for ID-1 cards <i>Festlegung einer fühlbaren Aussparung für ID-1 Karten zur Erkennung der Ausrichtung der Karte.</i>
– 3 : 1999	Part 3: Key pads
– 4 : 1999	Part 4: Coding of user requirements for people with special needs
EN 1362 : 1997	Identification card systems – Device interface characteristics – Classes of device interfaces
EN 1387 : 1996	Machine readable cards – Health care applications – Cards: General characteristics
EN 1545-1 : 1998	Identification card systems – Surface transport applications – Part 1: General data elements
EN 1545-2 : 1998	Identification card systems – Surface transport applications – Part 2: Transport payment
prEN 1545-3 : 1995	Identification card systems – Surface transport applications – Part 3: Tachograph
prEN 1545-4 : 1995	Identification card systems – Surface transport applications – Part 4: Vehicle and driver licencing
EN 1546	Identification card systems – Inter-sector electronic purse ☒ <i>Die weltweit wichtigste Norm für elektronische Geldbörsen, welche die Grundlage der meisten Börsensysteme darstellt. Die Normenreihe ist relativ allgemein gehalten, enthält somit viele Optionen, ist aber eine sehr gute und auch vollständige Beschreibung einer elektronischen Geldbörse.</i>
– 1 : 1999	Part 1: Definition, concepts and structures <i>Definition der Begriffe für die gesamte Normenreihe und Beschreibung der grundlegenden Konzepte und Strukturen für die branchenübergreifende elektronische Geldbörse.</i>
– 2 : 1999	Part 2: Security architecture <i>Beschreibung der verwendeten Notationen für Sicherheitsmechanismen, der Sicherheitsarchitektur und der dazugehörigen Abläufe und Mechanismen bei der branchenübergreifenden elektronischen Geldbörse.</i>
– 3 : 1999	Part 3: Data elements and interchanges <i>Beschreibung der Datenelemente, Dateien, Kommandos und Returncodes zwischen allen Komponenten der branchenübergreifenden elektronischen Geldbörse.</i>
– 4 : 1999	Part 4: Data objects <i>Beschreibung des TLV-Mechanismus zum Lesen beliebiger Datenelemente aus Dateien sowie detaillierte Darstellung der Komponenten und der Zustände für die Zustandsautomaten der branchenübergreifenden elektronischen Geldbörse. Inklusive eine Liste der Kennzeichen (tags) für alle verwendeten Datenelemente.</i>
EN 1867 : 1997	Machine-readable cards – Health care applications – Numbering system and registration procedure for issuer identifiers
EN 13343	Identification card systems – Telecommunications IC cards and terminals – Test methods and conformance testing for EN 726-3
– 1 prEN : 1998	Part 1: Implementation Conformance Statement (ICS) proforma specification
– 2 prEN : 1998	Part 2: Test Suite Structure and Test Purposes (TSS & TP)
– 3 prEN : 1998	Part 3: Abstract Test Suite (ATS) and Implementation extra Information for Testing (IXIT) proforma specification

EN 13 344	Identification card systems – Telecommunications IC cards and terminals – Test methods and conformance testing for EN 726-4
– 1 prEN : 1998	Part 1: Implementation Conformance Statement (ICS) proforma specification
– 2 prEN : 1998	Part 2: Test Suite Structure (TSS) and Test Purposes (TP)
– 3 prEN : 1998	Part 3: Abstract Test Suite (ATS) and Implementation extra Information for Testing (IXIT) proforma specification
EN 13 345	Identification card systems – Telecommunications IC cards and terminals – Test methods and conformance testing for EN 726-7
– 1 prEN : 1998	Part 1: Implementation Conformance Statement (ICS) proforma specification
– 2 prEN : 1998	Part 2: Test Suite Structure and Test Purposes (TSS & TP)
– 3 prEN : 1998	Part 3: Abstract Test Suite (ATS) and Implementation extra Information for Testing (IXIT) proforma specification
EN 1750 : 1999	Identification card systems – Inter-sector messages between devices and hosts – Acceptor to acquirer messages
PC/SC V 1.0 Dezember 1997	<i>Interoperability Specification for ICCs and Personal Computer Systems</i> <i>Diese umfangreiche und detaillierte Spezifikation ist die Grundlage der Anbindung von Chipkarten und Terminals an das Ressourcenmanagement der 16-Bit- und 32-Bit-Betriebssysteme von Microsoft.</i>
– 1	Part 1: Introduction and Architecture Overview
– 2	Part 2: Interface Requirements for Compatible IC Cards and Readers
– 3	Part 3: Requirements for PC-Connected Interface Devices
– 4	Part 4: IFD Design Considerations and Reference Design Information
– 5	Part 5: ICC Resource Manager Definition
– 6	Part 6: ICC Service Provider Interface Definition
– 7	Part 7: Application Domain and Developer Design Considerations
– 8	Part 8: Recommendations for ICC Security and Privacy Devices
prEN 13 344 : 1998	Identification card systems – Telecommunications IC cards and terminals – Test methods and conformance testing for EN 726-4
– 1	Part 1: Implementation Conformance Statement (ICS) proforma specification
– 2	Part 2: Test Suite Structure (TSS) and Test Purposes (TP)
– 3	Part 3: Abstract Test Suite (ATS) and Implementation extra Information for Testing (IXIT) proforma specification
prEN 13 345-1 : 1998	Identification card systems – Telecommunications IC cards and terminals – Test methods and conformance testing for EN 726-7
– 1	Part 1: Implementation Conformance Statement (ICS) proforma specification
– 2	Part 2: Test Suite Structure and Test Purposes (TSS & TP)
– 3	Part 3: Abstract Test Suite (ATS) and Implementation extra Information for Testing (IXIT) proforma specification
ENV 13 729 : 2000	Health informatics – Secure user identification – Strong authentication using microprocessor cards

ENV 1257	Identification card systems – Rules for Personal Identification Number handling in intersector environments <i>Darstellung und Erläuterung der Sicherheitsaspekte im Umgang mit der PIN, von der Übergabe der PIN zum Kartenbesitzer (PIN-Brief) bis hin zur Eingabe der PIN an einer Tastatur (PIN-Pad).</i>
– 1 prENV : 1997	Part 1: PIN presentation
– 2 prENV: 1997	Part 2: PIN protection
– 3 prENV : 1997	Part 3: PIN verification
ETS 300 331 : 1995	Digital Enhanced Cordless Telecommunications (DECT); DECT Authentication Module (DAM) <i>Beschreibung der Chipkarte (DAM) für das DECT-System. Sie umfasst alle dazugehörigen Kommandos, Dateien, Zugriffsbedingungen und Authentisierungsabläufe. Die Abmessungen der Kartenformate Mini ID-Karte und Plug-In-Karte sind ebenfalls definiert. Diese Norm lehnt sich stark an die GSM 11.11 Norm an.</i>
EN 300812, Version 2.1.1 : 2001	Terrestrial Trunked Radio (TETRA); Security aspects; Subscriber Identity Module to Mobile Equipment (SIMME) interface
FIPS 46-3 : 1999	Data Encryption Standard (DES) ☒ <i>Diese Norm beschreibt den DES- und Triple-DES-Algorithmus.</i>
FIPS 74 : 1981	Guidelines for Implementing and Using the NBS Encryption Standard
FIPS 81 : 1980	DES Modes of Operation
FIPS 140-2 : 2001	Security Requirements for Cryptographic Modules ☒ <i>Diese Norm ist eine weltweit eingesetzte Grundlagennorm bezüglich der Sicherheitsanforderungen von Sicherheitsmodulen, zu denen auch Chipkarten gezählt werden können. Sie definiert vier unterschiedliche hohe Sicherheitsniveaus für Sicherheitsmodule und beschreibt detailliert sieben sicherheitsrelevante Anforderungsbereiche. Der Inhalt dieser Norm ist sehr praxisbezogen und geht auch auf technische Realisierungsdetails wie beispielsweise Kriterien für die Qualität von Zufallszahlengeneratoren ein.</i>
FIPS 180-1 : 1995	Secure Hash Standard (SHA) ☒ <i>Diese Norm beschreibt die SHA-1 Hash-Funktion.</i>
FIPS 186-2 : 2000	Digital Signature Standard (DSS) ☒ <i>Diese Norm beschreibt den DSS-Algorithmus.</i>
FIPS 197 : 2001	Advanced Encryption Standard (AES) ☒ <i>Diese Norm beschreibt den AES-Algorithmus.</i>
GSM 01.02, Version 6.0.1 : 2001	Digital cellular telecommunications system (Phase 2+) (GSM); General description of a GSM Public Land Mobile Network (PLMN) <i>Dies ist die Grundlage der Architektur aller GSM-Mobilfunknetzwerke.</i>
GSM 01.04, Version 8.0.0 : 1999	Digital cellular telecommunications systems (Phase 2) (GSM); Abbreviations and acronyms
GSM 01.60, Version 6.0.0 : 1998	Digital cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS) Requirements specification of GPRS
GSM 02.09, Version 7.0.1 : 1998	Digital cellular telecommunications systems (Phase 2) (GSM); Security Aspects

GSM 02.17, Version 8.0.0 : 1999

Digital cellular telecommunications systems (Phase 2) (GSM); SIM Functional Characteristics

Kurze Norm mit den Festlegungen der grundsätzlich notwendigen Funktionalitäten eines Sicherheitsmoduls, d. h. einer SIM, für ein GSM-Mobilfunknetzwerk. Diese Norm ist das GSM-Äquivalent zur UMTS-Norm TS 21.111.

GSM 02.19, Version 7.1.0 : 1998

Digital cellular telecommunications system (Phase 2+) (GSM); Subscriber Identity Module Application Programming Interface (SIM API); Service description; Stage 1

Kurze Norm mit der Aufstellung aller grundlegenden Dienste eines sprachunabhängigen API für ausführbaren Programmcode (z.B. Java) in der SIM. Aufbauend auf dieser Norm spezifiziert die GSM 03.19 eine detaillierte Umsetzung für die Realisierung eines Java Card APIs für SIMs.

GSM 02.22, Version 7.0.0 : 1999

Digital cellular telecommunications system (Phase 2+) (GSM); ; Personalisation of GSM Mobile Equipment (ME); Mobile functionality specification

Beschreibung der Mechanismen für Personalisierung und Depersonalisierung des Mobile Equipments über spezifische Daten der SIM. Dies wird im Allgemeinen als SIM-Lock bezeichnet.

GSM 02.34, Version 6.0.0 : 1997

Digital cellular telecommunications system (Phase 2+); High Speed Circuit Switched Data (HSCSD); Stage 1

GSM 02.48, Version 8.0.0 : 2000

Digital cellular telecommunications system (Phase 2+) (GSM); Security mechanisms for the SIM application toolkit; Stage 1

Kurze Norm mit der Beschreibung der grundlegenden anwendungsunabhängigen Sicherheitsmechanismen in Verbindung mit dem SIM Application Toolkit nach GSM 11.14. Aufbauend auf dieser Norm spezifiziert die GSM 03.48 eine detaillierte Umsetzung für die Realisierung.

GSM 02.60, Version 6.3.0 : 1997

Digital cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS); Service description; Stage 1

GSM 03.19, Version 8.2.0 : 2001

Digital cellular telecommunications system (Phase 2+); Subscriber Identity Module Application Programming Interface (SIM API); SIM API for Java Card; Stage 2

☒ *Spezifikation einer Java-Card-Variante für den Einsatz als SIM mit SIM Application Toolkit auf der Basis der Java Card 2.1-Spezifikationen. Diese Norm ist das zentrale Dokument für den Einsatz von Java Card bei GSM. Die Grundlage dazu bildet die GSM 02.19.*

GSM 03.20, Version 8.1.0 : 1999

Global System for Mobile communication (GSM) (Phase 2+); Security related network functions

GSM 03.38, Version 7.2.0 : 1999

Digital cellular telecommunications system (Phase 2+) (GSM); Alphabets and language-specific information

In dieser Norm ist der an ASCII angelehnte GSM-Zeichensatz festgelegt.

GSM 03.40, Version 7.4.0 : 2000

Digital cellular telecommunications system (Phase 2+) (GSM); Technical realization of the Short Message Service (SMS)

GSM 03.48, Version 8.7.0 : 2001

Digital cellular telecommunications system (Phase 2+); Security Mechanisms for the SIM application toolkit; Stage 2

☒ *Diese Norm enthält die Spezifikation aller Sicherheitsmechanismen für eine abhör- und manipulationssichere Verbindung zwischen Hintergrundsystem und SIM. Außerdem ist der grundsätzliche Mechanismus eines Remote File Management, auch durch SMS, beschrieben. Die Grundlage dieses Dokuments ist die GSM 02.48.*

GSM 09.91 : 1995

European digital cellular telecommunications system (Phase 2); Interworking aspects of the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface between Phase 1 and Phase 2

GSM 11.10 Version 8.2.0 : 2000

Digital cellular telecommunications system (Phase 2+) (GSM) – Mobile Station (MS) conformance specification

Sehr umfangreiche Testspezifikation für die GSM Mobilstationen.

GSM 11.11 Version 8.5.0 : 2001

Digital cellular telecommunications system (Phase 2+) – Specification of the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface

☒ *Spezifikation der physikalischen und logischen Eigenschaften einer SIM, wobei diese durch eine Beschreibung der Schnittstelle zwischen SIM und dem GSM-Mobiltelefon realisiert ist. Die Spezifikation beinhaltet die Definition der Kartengrößen ID-1 und Plug-In, der mechanischen Rahmenparameter für die Karte und die Kontakte. Weiterhin sind in GSM 11.11 die Festlegung aller elektrischen Rahmenwerte sowie Struktur und Dateninhalt von ATR und PPS enthalten, sowie die Festlegung der möglichen Dateistrukturen, Sicherheitsmechanismen, Kommandos und Returncodes. Zusätzlich sind alle für eine SIM notwendigen Datenelemente und Dateien aufgeführt sowie typische Kommandosequenzen. Diese Norm ist das GSM-Äquivalent zu den UMTS-Normen TS 31.101 und TS 31.102.*

GSM 11.12 Version 4.3.1 : 1998

Digital cellular telecommunications system (Phase 2); Specification of the 3 Volt Subscriber Identity Module – Mobile Equipment (SIM – ME) interface

Spezifikation für 3 Volt SIMs inklusive einer Kompatibilitätsliste für SIMs, die nach den Vorgängerspezifikationen programmiert wurden. Diese Norm beinhaltet lediglich Unterschiede und Ergänzungen zu GSM 11.11 in Bezug auf 3 Volt SIMs.

GSM 11.13 Version 7.2.0 : 2000

Digital cellular telecommunications system (Phase 2+); Test specification for SIM API for Java card

Festlegung von Testumgebung, Testanwendungen, Testabläufen, Testabdeckung und der einzelnen Testfälle für des SIM API für Java Card nach GSM 03.19. Die beschriebenen Tests zielen ausnahmslos auf die informationstechnischen Aspekte einer Java Card für GSM. Diese Norm zeigt sehr gut und ausführlich, wie Tests für eine Java Card beschrieben, aufgebaut und durchgeführt werden können.

GSM 11.14 Version 8.8.0 : 2001

Digital cellular telecommunications system (Phase 2+); Specification of the SIM Application Toolkit for the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface

Definition und ausführliche Beschreibung des SIM Application Toolkits für SIMs. SAT beschreibt eine Schnittstelle zwischen Mobiltelefon und SIM zur teilweisen Steuerung des Mobiltelefons von Zusatzanwendungen der SIM aus. Die Norm führt proaktive Kommandos für die SIM ein und definiert viele neue Kommandos im Zusammenhang mit der Steuerung des Mobiltelefons, wie beispielsweise Displayausgabe, Tastaturabfrage oder Versenden von Kurznachrichten. Das UMTS-Äquivalent zu dieser Norm ist die TS 31.111.

GSM 11.17, Version 7.0.2 : 1998

Digital cellular telecommunications system (Phase 2+) (GSM); Subscriber Identity Module (SIM) conformance test specification

Festlegung von Testumgebung, Testausrüstung, Testhierarchie und der einzelnen Testfälle für den Test von SIMs. Die beschriebenen Tests zielen ausnahmslos auf die elektrischen und informationstechnischen Aspekte. Dazu sind Tests von der elektrischen Versorgung über Datentübertragung, Dateiverwaltung, Kommandos bis hin zu typischen Abläufen in der GSM-Anwendung im Detail festgelegt. Diese Norm zeigt sehr gut und ausführlich, wie SIM Tests beschrieben, aufgebaut und durchgeführt werden können. Das UMTS-Äquivalent zu dieser Norm ist die TS 31.122.

GSM 11.18 Version 7.0.1 : 1998

Digital cellular telecommunications system (Phase 2+); Specification of the 1.8 Volt Subscriber Identity Module – Mobile Equipment (SIM – ME) interface

GSM 11.19 Version 7.0.3 : 1998

Digital cellular telecommunications system (Phase 2+) (GSM) – Specification of the Cordless Telephony System Subscriber Identity Module for both Fixed Part and Mobile

IEEE 828 : 1990

Standard for Software Configuration Management Plans

IEEE 1363 : 2000

Standard Specifications for Public-Key Cryptography

☒ *Dies ist eine sehr ausführliche und umfassende Norm, welche beinahe alle Aspekte – von der Schlüsselgenerierung bis zur Verwendung bei digitaler Signatur, Schlüsselaustausch und Verschlüsselung bei asymmetrischen Kryptoalgorithmen – abdeckt.*

ISO 639 : 1988

Codes for the representation of names of languages

ISO 639

Codes for the representation of names of

– 1 : 2001

Part 1: Alpha-2 code

– 2 : 1998

Part 2: Alpha-3 code

ISO/IEC 646 : 1991

Information technology – ISO 7-bit coded character set for information interchange

ISO 3166

Codes for the representation of names of countries and their subdivisions

– 1 : 1997

Part 1: Country codes

– 2 : 1998

Part 2: Country subdivision code

– 3 : 1999

Part 3: Code for formerly used names of countries

ISO 4217 : 1995

Codes for the representation of currencies and funds

ISO 4909 : 2000

Bank cards – Magnetic stripe data contents for track 3

ISO/IEC 7501

Identification cards – Machine readable travel documents

– 1 : 1997

Part 1: Machine readable passport

– 2 : 1997	Part 2: Machine readable visa
– 3 : 1997	Part 3: Machine readable official travel documents
ISO 7810 : 1995	Identification cards – Physical characteristics <i>Beschreibung der wichtigsten physikalischen Eigenschaften einer Karte ohne Chip und Definition der Kartengrößen ID-1, ID-2 und ID-3.</i>
ISO 7811	Identification cards – Recording technique <i>Diese Normenreihe ist eine wichtige Referenz für die mechanischen Aspekte von Karten und legt für diesen Bereich die wesentlichen Kartenelemente in ihrer Ausführung fest.</i>
– 1 : 1995	Part 1: Embossing <i>Exakte Definition der zehn Ziffern sowie der grundlegenden Beschriftungsart für die Hochprägung von Karten.</i>
– 2 : 2001	Part 2: Magnetic stripe – low coercivity <i>Definition der Größe und Position des Magnetstreifens einer Karte. Außerdem sind in dieser Norm die physikalischen Eigenschaften des magnetisierbaren Materials und die Codierung der Zeichen auf dem Magnetstreifen festgelegt.</i>
– 3 : 1995	Part 3: Location of embossed characters on ID-1 cards <i>Definition von möglichen Positionen der Hochprägung auf ID-1 Karten.</i>
– 4 : 1995	Part 4: Location of read-only magnetic tracks – Tracks 1 and 2 <i>Definition der Position der nur lesbaren Spuren 1 und 2 auf einer ID-1 Karte.</i>
– 5 : 1995	Part 5: Location of read-write magnetic track – Track 3 <i>Definition der Position der schreib- und lesbaren Spur 3 auf ID-1 Karten.</i>
– 6 : 2001	Part 6: Magnetic stripe – High coercivity
– 7 WD : 2001	Part 7: Magnetic stripe – High coercivity high density
ISO 7812	Identification cards – Identification of issuers
– 1 : 2000	Part 1: Numbering system <i>Spezifikation eines Nummerierungsschemas für die Herausgeber von ID-Karten.</i>
– 2 : 2000	Part 2: Application and registration procedures <i>Festlegung der Registrierungsinstanz und eines Formulars für die Registrierung von Anwendungen. Enthält außerdem den Algorithmus für die Prüfsumme nach Luhn (Modulo 10 Prüfsumme).</i>
ISO 7813 : 2001	Identification cards – Financial transaction cards <i>Definition der grundlegenden physikalischen Eigenschaften, Größe und Hochprägung der ID-1 Karte nach ISO 7810 für Karten im Bereich des Zahlungsverkehrs. Auch sind hier die Dateninhalte von Spur 1 und 2 des Magnetstreifens definiert.</i>
ISO/IEC 7816	Identification cards – Integrated circuit(s) cards with contacts ☒ <i>Dies ist die wichtigste ISO-Normenreihe für Mikrocontroller-Chipkarten. Die ersten drei Teile fokussieren vor allem auf die Karten- und Chiphardware. Die nachfolgenden Teile legen alle Mechanismen und Eigenschaften von Anwendungen und Betriebssystemen für Chipkarten sowie der dazugehörigen Informationstechnik fest.</i>
– 1 : 1998	Part 1: Physical characteristics <i>Definition der physikalischen Eigenschaften einer Karte mit einem kontaktbehafteten Chip, sowie der dafür anzuwendenden Tests.</i>

- 2 : 1999 Part 2: Dimensions and location of the contacts
Definition der Größe und Position der Kontaktelemente einer Chipkarte sowie der möglichen Anordnung von Chip, Magnetstreifen und Hochprägung. Auch ist die Messmethode für die Position der Kontakte auf der Chipkarte beschrieben.
- 3 : 1997 Part 3: Electronic signals and transmission protocols
☒ *Die wichtigste ISO-Norm für die elektrischen Rahmenparameter einer Mikrocontroller-Chipkarte. In dieser Norm sind alle grundlegenden elektrischen Eigenschaften wie Spannungsversorgung (5 Volt und 3 Volt), Anhalten der Taktfrequenz und Resetverhalten (Kalt- und Warmreset) festgelegt. Außerdem sind die Datenelemente, der Aufbau und die möglichen Abläufe von ATR und PTS definiert. Ein großer Teil dieser Norm befasst sich noch mit den Grundlagen der Datenübertragung auf physikalischer Ebene (z. B. Teiler) und die Definition der beiden Übertragungsprotokolle $T = 0$ und $T = 1$ mit ausführlichen Beispielen des Kommunikationsablaufs.*
- 4 : 1995 Part 4: Inter-industry commands for interchange
☒ *Die wichtigste ISO-Norm auf Anwendungsebene für Chipkarten. Definition der Dateiorganisation, Dateistrukturen, Sicherheitsarchitektur, TPDUs, APDUs, Secure Messaging, Returncodes und der logischen Kanäle. Den größten Teil nimmt eine ausführliche Beschreibung der Kommandos für Chipkarten ein. Außerdem werden die grundlegenden Mechanismen von Chipkarten für allgemeine Industrieanwendungen beschrieben.*
- 4 Amd. 1 : 1997 Part 4 – Amendment 1: Use of secure Messaging
- 5 : 1994 Part 5: Numbering system and registration procedure for application identifiers
Definition des Nummerierungsschemas für die eindeutige Identifizierung von nationalen und internationalen Anwendungen in Chipkarten. Außerdem werden die exakte Datenstruktur der AIDs definiert und die Registrierungsprozedur für Anwendungen erklärt.
- 5 Amd. 1 : 1996 Part 5 – Amendment 1: Registration of identifiers
- 6 CD : 2001 Identification cards – Integrated circuit(s) cards with contacts – Part 6: Inter-industry data elements
Definition von Datenelementen (data objects – DO) und dazugehöriger TLV-Kennzeichnung für allgemeine Industrieanwendungen. Die dazugehörigen TLV-Strukturen sind ebenfalls erläutert sowie die Prozeduren für das Auslesen von Datenobjekten aus Chipkarten.
- 7 : 1999 Part 7: Interindustry commands for Structured Card Query Language (SCQL)
Definition von zusätzlichen Kommandos für Chipkarten in Ergänzung zu ISO/IEC 7816-4. In dieser Norm sind die Prinzipien einer an SQL angelehnten Datenbank für Chipkarten definiert. Ebenso legt sie die Kommandos für die dazugehörigen SCQL-Zugriffe auf Chipkarten fest.
- 8 : 1999 Part 8: Security related interindustry commands
Dieser Teil der Normenreihe ist zur Gänze dem Thema sicherheitsrelevante Funktionen und Kommandos gewidmet. Die Norm definiert ergänzend zu ISO/IEC 7816-4 zusätzliche Mechanismen für Secure Messaging. Des Weiteren sind umfangreiche Kommandos für kryptografische Funktionen, wie beispielsweise digitale Signatur, Hash-Berechnung, MAC-Berechnung, Ver- und Entschlüsselung von Daten beschrieben.

- 9 : 2000 Part 9: Enhanced interindustry commands
In dieser in drei Teile gegliederten Norm ist am Beginn der Lebenszyklus von Chipkarten-Anwendungen auf Dateiebene in Form von Zuständen beschrieben. Der nächste große Abschnitt beschäftigt sich mit der Beschreibung von Zugriffskontrollobjekten (access control objects – ACO), die für die Reglementierung von Dateizugriffen verwendet werden können. Der dritte umfangreiche Teil der Norm definiert Suchkommandos für Dateiinhalte und die für die Verwaltung von Anwendungen notwendigen Administrationskommandos zum Erzeugen und Löschen von Dateien.
- 10 : 1999 Part 10: Electronic signals answer to reset for synchronous cards
Dieser Teil ist für Speicherkarten das Gegenstück zu Teil 3 dieser Normenreihe. Hier werden die wesentlichen elektrischen Rahmenparameter für Speicherkarten festgelegt. Auch sind die Datenelemente, der Aufbau und die möglichen Abläufe des ATR für synchrone Karten definiert.
- 11 CD : 2000 Part 11: Personal verification through biometric methods
Dieser Normenteil definiert Kommandos zur biometrischen Benutzeridentifizierung sowie dazugehörige Datenelemente. Des Weiteren werden im Anhang überblickshaft der Ablauf für die Aufnahme der biometrischen Daten in die Chipkarte (enrollment) aufgezeigt und ein Szenario für die Überprüfung dieser biometrischen Daten beschrieben.
- 15 CD : 2001 Part 15: Cryptographic Information Application
Dieser Teil der Normenreihe nutzt den PKCS #15 Standard als Grundlage und definiert darauf aufbauend alle notwendigen Datenelemente für eine interoperable Chipkarte für digitale Signatur. Die Norm enthält eine Beschreibung aller für eine Signaturkarte notwendigen Datenobjekte, Verzeichnisse und Dateien sowie eine ASN.1-Beschreibung aller in den Dateien gespeicherten Zertifikate, Schlüssel und sonstiger Verwaltungsinformationen.
- ISO 8372 : 1987 Information processing – Modes of Operation for a 64-Bit Block Cipher Algorithm
☒ *Definition der vier Modi für Verschlüsselungsalgorithmen mit 64 Bit Blockgröße (z. B.: DES): Electronic Codebook (ECB), Cipher Block Chaining (CBC), Output Feedback (OFB) und Cipher Feedback (CFB). Die in ANSI X 3.106 und FIPS 81 beschriebenen Modi für Blockverschlüsselungsverfahren stellen eine Untermenge dieser Norm dar.*
- ISO 8583 : 1993 Financial transaction card originated messages – Interchange message specifications
Norm für die Datenübertragung zwischen Terminal und Host. In einer angelehnten Form funktioniert auch in Deutschland die Kommunikation zwischen Terminals für Debitkarten und dem Hintergrundsystem.
- ISO 8583 Financial transaction card originated messages – Interchange message specifications
- 1 CD : 1998 Part 1: Messages, data elements and code values
- 2 : 1998 Part 2: Application and registration procedures for Institution Identification Codes (IIC)
- 3 : 1988 Part 3: Maintenance procedures for messages, data elements and code values
- ISO 8730 : 1990 Banking – Requirements for message authentication
Grundlagen für die Sicherung von Daten bei der Übertragung d. h. Erzeugung und Prüfung von MACs. Im Anhang dazu befinden sich ausführliche Zahlenbeispiele, sowie die Beschreibung eines Pseudozufallszahlengenerators mit dem DES.

ISO 8731	Banking – Approved algorithms for message authentication
– 1 : 1987	Part 1: DEA <i>Sehr kurze Norm, in der der DEA für MAC-Berechnung als geeignet bezeichnet wird. Außerdem wird kurz die Paritätsberechnung für DES-Schlüssel beschrieben.</i>
– 2 : 1992	Part 2: Message authenticator algorithm <i>Definition eines schnellen Algorithmus für die MAC-Berechnung im Bankenbereich. Im Anhang befinden sich Zahlenbeispiele sowie eine exakte Beschreibung des Algorithmus.</i>
ISO 8732 : 1988	Banking – Key management <i>Umfangreiche Norm über die Grundlagen und Verfahren des Schlüsselmanagements zwischen zwei oder mehreren beteiligten Instanzen mit Hilfe von symmetrischen Kryptoalgorithmen.</i>
ISO/IEC 8824	Information technology – Abstract Syntax Notation One (ASN.1) <i>Definition der grundlegenden Codierungsregeln von ASN.1.</i>
– 1 : 1998	Part 1: Specification of basic notation
– 1 : 1998 / Amd 1 : 2000	Part 1 – Amendment 1: Relative object identifiers
– 1 : 1998 / Amd 2 : 2000	Part 1 – Amendment 2: ASN.1 Semantic Model
– 1 : 1998 / Amd 3 : 2000	Part 1 – Amendment 3: XML value notation
– 1 : 1998 / Amd 4 : 2000	Part 1 – Amendment 4: Version number support
– 2 : 1998	Part 2: Information object specification
– 2 : 1998 / Amd 1 : 2000	Part 2 – Amendment 1: ASN.1 semantic model
– 2 : 1998 / Amd 2	Part 2 – Amendment 2: XML value notation
– 3 : 1998	Part 3: Constraint specification
– 4 : 1998	Part 4: Parameterization of ASN.1 specifications
– 4 : 1998 / Amd 1 : 2000	Part 4 – Amendment 1: ASN.1 semantic model
ISO/IEC 8825	Information technology – Open Systems Interconnection – Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1) <i>Definition der Datenbeschreibungssprache ASN.1.</i>
– 1:1998	Part 1: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)
– 1:1998 / Amd 1:2000	Part 1 – Amendment 1: Relative object identifiers
– 2:1998	Part 2: Specification of Packed Encoding Rules (PER)
– 2:1998 / Amd 1:2000	Part 2 – Amendment 1: Relative object identifiers
– 3 FCD : 2001	Part 3: Specification of Encoding Control Notation (ECN)
– 3 : FCD / Amd 1 : 2001	Part 3 – Amendment 1: ASN.1 extensibility notation
– 4 : WD 2000	Part 4: XML Encoding Rules (XER)
ISO/IEC 8859 - 1 : 1998	Information technology – 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1
ISO/IEC 9075	Information technology – Database languages – SQL <i>Definition der Datenbankabfragesprache Structured Query Language (SQL), welche eine Obermenge der chipkartenbasierten Datenbankabfragesprache SCQL ist.</i>
– 1 : 1999	Part 1: Framework (SQL/Framework)
– 1 : 1999 / Amd 1 : 2001	Part 1 – Amendment 1: On-Line Analytical Processing (SQL/OLAP)
– 2 : 1999	Part 2: Foundation (SQL/Foundation)
– 2 : 1999 / Amd 1 : 2001	On-Line Analytical Processing (SQL/OLAP)
– 3 : 1999	Part 3: Call-Level Interface (SQL/CLI)

– 4 : 1999	Part 4: Persistent Stored Modules (SQL/PSM)
– 5 : 1999	Part 5: Host Language Bindings (SQL/Bindings)
– 5 : 1999 / Amd 1 : 2001	On-Line Analytical Processing (SQL/OLAP)
– 9 : 2001	Part 9: Management of External Data (SQL/MED)
– 10 : 2000	Part 10: Object Language Bindings (SQL/OLB)
– 11 : CD 2001	Part 11: Information and definition schemas (SQL/schemata)
– 12 : AWI 2000	Part 12: Replication
– 13 : FCD 2001	Part 13: Java Routines and Types (SQL/JRT)
– 14 : WD 2001	Part 14: XML-Related Specifications (SQL/XML)
ISO/IEC 9126	Software engineering – Product quality
– 1 : 2001	Part 1: Quality model
– 2 : CD 2001	Part 2: External metrics
– 3 : CD 2001	Part 3: Internal metrics
– 4 : CD 2001	Part 4: Quality in use metrics
ISO 9564	Banking – Personal Identification Number management and security
– 1 : 1991	Part 1: PIN protection principles and techniques <i>Grundlagen der PIN-Auswahl, des PIN-Managements und des Schutzes der PIN für allgemeine Bankanwendungen. In den Anhängen sind unter anderem generelle Forderungen an Eingabegeräte für PINs definiert. Auch sind dort Vorschläge für das Layout der entsprechenden Tastaturen vorhanden. Des Weiteren befinden sich im Anhang Hinweise zum Löschen von sensiblen Daten auf verschiedenen Medien wie Magnetband, Papier oder Halbleiterspeicher.</i>
– 2 : 1991	Part 2: Approved algorithm(s) for PIN encipherment <i>Sehr kurze Norm, die den DES als Algorithmus zur PIN-Verschlüsselung definiert.</i>
– 3 : 2002	Part 3: PIN protection requirements for offline PIN handling in ATM and POS systems
ISO/IEC 9646-3 : 1998	Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 3: The Tree and Tabular Combined Notation (TTCN) <i>Umfangreiche Norm, welche eine allgemeine Hochsprache für die Festlegung von Tests beschreibt. TTCN wird vereinzelt im Umfeld von Chipkartentests eingesetzt.</i>
ISO/IEC 9796	Information technology – Security techniques – Digital signature scheme giving message recovery <i>Definition von Verfahren zur Erstellung und Überprüfung digitaler Signaturen mit Nachrichtenrückgewinnung. Im Anhang befinden sich mehrere Zahlenbeispiele zur Schlüsselerzeugung, Signaturerstellung und Prüfung der Signatur.</i>
– 1 : 1999	Part 1: Mechanisms using redundancy
– 2 : 1997	Part 2: Mechanisms using a hash-function
– 3 : 2000	Part 3: Discrete logarithm based mechanisms
ISO/IEC 9797	Information technology – Security techniques – Message Authentication Codes (MACs)
– 1 : 1999	Part 1: Mechanisms using a block cipher
– 2 : 1999	Part 2: Mechanisms using a dedicated hash-function

ISO/IEC 9798	Information technology – Security techniques – Entity authentication ☒ <i>Normenreihe, die im Detail die unterschiedlichen kryptografischen Techniken für die Authentisierung von einer, zwei oder drei beteiligten Instanzen beschreibt. Diese Normenreihe ist die wichtigste Referenz zum Thema Authentisierung.</i>
– 1 : 1997	Part 1: General <i>Festlegung der Bezeichnungen und Notationen für die weiteren Teile der Normenreihe.</i>
– 2 : 1999	Part 2: Mechanisms using symmetric encipherment algorithms <i>Festlegung von Authentisierungsverfahren, die auf symmetrischen Kryptotalgorithmen basieren.</i>
– 3 : 1998	Part 3: Mechanisms using digital signature techniques <i>Festlegung von Authentisierungsverfahren, die auf digitalen Signaturen basieren.</i>
– 4 : 1999	Part 4: Mechanisms using a cryptographic check function <i>Festlegung von Authentisierungsverfahren, die auf kryptografischen Prüffunktionen basieren.</i>
– 5 : 1999	Part 5: Mechanisms using zero knowledge techniques <i>Festlegung von Authentisierungsverfahren, die auf Zero-Knowledge-Verfahren basieren.</i>
ISO 9807 : 1991	Banking and related financial services – Requirements for message authentication (retail)
ISO/IEC 9979 : 1999	Information technology – Security techniques – Procedures for the registration of cryptographic algorithms
ISO 9992	Financial Transaction Cards – Messages between the Integrated Circuit Card and the Card Accepting Device
– 1 : 1990	Part 1: Concepts and structures
– 2 : 1998	Part 2: Functions, messages (commands and responses), data elements and structures <i>Definition von Kommandos, Abläufen und Datenelementen für Chipkarten im Zahlungsverkehr. Enthält die Definition der Kennzeichen (tags) für Datenelemente im Zahlungsverkehr und viele Querverweise auf bestehende Normen der ISO/IEC 7816-Reihe.</i>
– 4 DIS : 1993	Part 4: Common data for interchange
– 5 CD : 1991	Part 5: Organization of data elements
ISO/IEC 10 116 : 1997	Information technology – Security techniques – Modes of operation for an n-bit block cipher algorithm <i>Beschreibung der vier üblichen Modi (ECB, CBC, CFB, OFD), mit denen ein Blockverschlüsselungsalgorithmus betrieben werden kann. Im Anhang befinden sich detaillierte Anmerkungen zur jeweiligen Anwendung der vier Verschlüsselungsmodi sowie in einem weiteren Anhang die entsprechenden Zahlenbeispiele dazu.</i>
ISO/IEC 10 118	Information technology – Security techniques – Hash functions <i>Allgemeine Grundlagen zu Hash-Funktionen, sowie dazugehörige Padding-Methoden.</i>
– 1 : 2000	Part 1: General
– 2 : 2000	Part 2: Hash functions using an n-bit block cipher algorithm <i>Definition von Hash-Funktionen, die einen Blockverschlüsselungsalgorithmus als Grundlage benutzen. Es ist ein Algorithmus mit einfacher und einer mit doppelter Schlüssellänge beschrieben. Im Anhang befindet sich jeweils ein dazu passendes Zahlenbeispiel auf der Basis des DES.</i>
– 3 : 1998	Part 3: Dedicated hash functions
– 4 : 1998	Part 4: Hash-functions using modular arithmetic

ISO 10202	Financial Transaction Cards – Security Architecture of Financial Transaction Systems using Integrated Circuit Cards
– 1 : 1991	Part 1: Card life cycle
– 2 : 1996	Part 2: Transaction process
– 3 : 1998	Part 3: Cryptographic key relationship
– 4 : 1996	Part 4: Secure application modules
– 5 : 1998	Part 5: Use of algorithms
– 6 : 1994	Part 6: Card holder verification
– 7 : 1998	Part 7: Key Management <i>Definition von allgemeinen Mechanismen für Schlüsselmanagement und zur Schlüsselableitung. Es werden sowohl symmetrische als auch asymmetrische Verfahren beschrieben.</i>
– 8 : 1998	Part 8: General principles and overview
ISO/IEC 10 373	Identification cards – Test methods
	☒ <i>Grundlegende Norm zum Test von Karten. Genaue Beschreibung von Testmethoden für Kartenkörper und Kartenkörper in Verbindung mit dem implantierten Chip. Die einzelnen Tests werden mit vielen erklärenden Zeichnungen detailliert beschrieben.</i>
– 1 : 1998	Part 1: General characteristics tests
– 2 : 1998	Part 2: Cards with magnetic stripes
– 3 : 2001	Part 3: Integrated circuit(s) cards with contacts and related interface devices <i>Festlegung von Testumgebung, Testmethoden und Testabläufen für elektrische Tests von kontaktbehafteten Chipkarten. Dazu sind Abläufe für die Prüfung der Kontaktpositionen, der elektrischen Versorgung und der Datenübertragung bei ATR, PTS und den Übertragungsprotokollen im Detail festgelegt.</i>
– 4 CD : 1998	Part 4: Contactless integrated circuit cards
– 5 : 1998	Part 5: Optical memory cards
– 6 : 2001	Part 6: Proximity cards
– 7 : 2001	Part 7: Vicinity cards
ISO/IEC 10 536	☒ Identification cards – Contactless integrated circuit(s) cards – Close-coupled cards
	<i>In dieser Norm werden kontaktlose Chipkarten beschrieben, deren Einsatzgebiet sich auf den direkten Kontakt mit dem Terminal beschränkt.</i>
– 1 : 2000	Part 1: Physical characteristics <i>Definition der physikalischen Eigenschaften von kontaktlosen Chipkarten sowie der dazugehörigen Testmethoden.</i>
– 2 : 1995	Part 2: Dimension and location of coupling areas <i>Spezifikation der Abmessungen und Lage der Koppelflächen für kontaktlose Karten und den Betrieb in Kartenterminals mit Kartenschlitz oder an der Oberfläche.</i>
– 3 : 1996	Part 3: Electronic signals and reset procedures <i>Definition der elektrischen Signale der induktiven und kapazitiven Koppellemente zwischen Terminal und Chipkarte.</i>
– 4 CD : 1997	Part 4: Answer to reset and transmission protocols <i>Festlegungen der Datenübertragung auf physikalischer Ebene, des Aufbaus und der Datenelemente von ATR und PTS für kontaktlose Chipkarten. Definition des Übertragungsprotokolls T = 2 mit vielen Beispielszenarien für den Protokollablauf.</i>

ISO/IEC 10646	Information technology – Universal Multiple-Octet Coded Character Set (UCS)
– 1 : 2000	Part 1: Architecture and Basic Multilingual Plane
– 2 : 2001	Part 2: Supplementary Planes
ISO 11 568	Banking – Key management
– 1 : 1994	Part 1: Introduction to Key Management
– 2 : 1994	Part 2: Key management techniques for symmetric ciphers
– 3 : 1994	Part 3: Key life cycle for symmetric ciphers
– 4 : 1998	Part 4: Key management techniques for public key cryptosystems
– 5 : 1998	Part 5: Key life for public key cryptosystems
– 6 : 1998	Part 6: Key management schemes
ISO/IEC 11 693 : 2000	Identification cards – Optical memory cards
ISO/IEC 11 694	Identification cards – Optical memory cards and devices – Linear recording method
– 1 : 2000	Part 1: Physical characteristics
– 2 : 2000	Part 2: Dimensions and location of the accessible optical area
– 3 : 2001	Part 3: Optical properties and characteristics
– 4 : 1996	Part 4: Logical data structures
ISO/IEC 11 770	Information technology – Security techniques – Key management
– 1 : 1996	Part 1: Framework
– 2 : 1996	Part 2: Mechanisms using symmetric techniques
– 3 : 1999	Part 3: Mechanisms using asymmetric techniques
ISO/IEC 12 207 : 1995	Information technology – Software life cycle processes
ISO/IEC 13 239 : 2000	Information technology – Telecommunications and information exchange between systems – High-level data link control (HDLC) procedures
ISO 13 491	Banking – Secure cryptographic devices
– 1 : 1998	Part 1: Concepts, requirements and evaluation methods
– 2 : 2000	Part 2: Security compliance checklists for devices used in magnetic stripe card systems
ISO/IEC 13 888	Information technology – Security techniques – Non-repudiation
– 1 : 1997	Part 1: General
– 2 : 1998	Part 2: Mechanisms using symmetric techniques
– 3 : 1997	Part 3: Mechanisms using asymmetric techniques
ISO/IEC 14 443	☒ Identification cards – Contactless integrated circuit(s) cards – Proximity cards
	<i>In dieser Norm werden kontaktlose Chipkarten beschrieben, die bis zu einigen 10 cm Abstand zu Terminals verwendet werden können.</i>
– 1 : 2000	Part 1: Physical characteristics
– 2 : 2001	Part 2: Radio frequency power and signal interface
– 3 : 2001	Part 3: Initialization and anticollision
– 4 : 2001	Part 4: Transmission protocol
ISO/IEC 14 888	Information technology – Security techniques – Digital signature with appendix
	<i>Diese Norm spezifiziert grundlegende Mechanismen und Vorgehensweisen für digitale Signaturen mit Anhang. Die Norm ist unabhängig von einem bestimmten asymmetrischen Kryptoalgorithmus.</i>

– 1 : 1998	Part 1: General
– 2 : 1999	Part 2: Identity-based mechanisms
– 3 : 1998	Part 3: Certificate-based mechanisms
ISO/IEC 15 292 : 2001	Information technology – Security techniques – Protection Profile registration procedures
ISO/IEC 15 408	Information technology – Security techniques – Evaluation criteria for IT security
– 1 : 1999	Part 1: Introduction and general model
– 2 : 1999	Part 2: Security functional requirements
– 3 : 1999	Part 3: Security assurance requirements
ISO/IEC 15 693	Identification cards – Contactless integrated circuit(s) cards – Vicinity cards <i>In dieser Norm werden kontaktlose Chipkarten beschrieben, die in bis zu einem Meter Abstand zu Terminals verwendet werden können.</i>
– 1 CD : 2000	Part 1: Physical characteristics
– 2 WD : 2000	Part 2: Air interface and initialization
– 3 WD : 2001	Part 3: Anticollision and transmission protocol
– 4 WD : 1996	Part 4: Extended command set and security features
ISO 15 782	Banking – Certificate management for financial services
– 1 DIS : 2000	Part 1: Public Key Certificates
– 2 : 2001	Part 2: Certificate extensions
ISO/IEC 15 946	Information technology – Security techniques – Cryptographic techniques based on elliptic curves
– 1 FDIS : 2001	Part 1: General
– 2 FDIS : 2001	Part 2: Digital signatures
– 3 FDIS : 2001	Part 3: Key establishment
– 4 CD : 2000	Part 4: Digital signatures giving message recovery
ISO 17 090	Public key infrastructure
– 1 CD : 2001	Part 1: Framework and overview
– 2 CD : 2001	Part 2: Certificate profile
– 3 CD : 2001	Part 3: Policy management of certification authority
ITU X.509 : 2000	Information Technology – Open Systems Interconnection – The Directory: Authentication Framework ☒ <i>Diese Norm legt Aufbau und Codierung von Zertifikaten fest. Sie stellt weltweit die meistbenutzte Grundlage für Zertifikatsstrukturen dar und ist identisch mit der ISO/IEC 9594-8.</i>
Java Card 2.1.1 : 2000	☒ <i>Dieser Industriestandard ist die Grundlage der Java Card; er wird vom Java Card Forum erstellt und von der Firma Sun veröffentlicht. Alle drei Standards ergänzen sich und decken die verschiedenen Implementierungs-Aspekte einer Java Card ab.</i>
Application Programming Interface	<i>Dieser Standard spezifiziert die gesamte Schnittstelle, das API, die einem Applet auf einer Java Card zur Verfügung steht. Das Dokument umfasst im Wesentlichen eine umfangreiche Aufstellung aller Klassen und Interfaces des Java Card API.</i>
Runtime Environment (JCRE) Specification	<i>Dieser Standard spezifiziert die Java Card Laufzeitumgebung, die im Wesentlichen aus der Virtuellen Maschine und dem Java Card API besteht. Das Dokument beschreibt im Detail die Themen Lebensdauer der Virtuellen Maschine und Lebensdauer von Applets, Selektion von Applets, transiente Objekte, Sharing von Objekten, Transaktionen und Atomität von Transaktionen sowie Installation von Applets.</i>

Virtual Machine Specification

Dieser Standard spezifiziert die Virtuelle Maschine der Java Card. Er umfasst den detaillierten Aufbau der Virtuellen Maschine, den Befehlsatz der Virtuellen Maschine und das Format der CAP-Dateien.

Multifunktionale Karten Terminals MKT Spezifikation, Version 1.0 : 1999

Die von Teletrust Deutschland herausgegebene MKT Spezifikation ist in Deutschland der Quasi-Standard für die Anbindung von Terminals an PCs.

- Teil 1 MKT-Basiskonzept
- Part 2 CT-ICC-Interface – MKT-Schnittstelle für kontaktorientierte Chipkarten mit synchroner und asynchroner Übertragung
- Part 3 CT-API 1.1 – Anwendungsunabhängiges Card Terminal Applikation Programming Interface
- Part 4 CT-BCS – Anwendungsunabhängiges Card Terminal Basic Command Set
- Part 5 Chipkarten mit synchroner Übertragung – ATR und Datenbereiche
- Part 6 Chipkarten mit synchroner Übertragung – Übertragungsprotokolle
- Part 7 Chipkarten mit synchroner Übertragung – Anwendung von Interindustry Commands

PKCS *Die Public Key Cryptography Standards sind von der Firma RSA Inc. veröffentlichte Industriestandards mit Fokus auf den Einsatz von asymmetrischen Kryptoalgorithmen.*

- PKCS #1 V 2.1 : 2001 RSA Encryption Standard
☒ *Dieser PKCS-Standard beschreibt die Funktionsweise der Ver- und Entschlüsselung mit dem RSA-Algorithmus.*
- PKCS #3 V 1.4 : 1993 Diffie-Hellman Key-Agreement Standard
Dieser PKCS-Standard beschreibt die Funktionsweise eines Schlüsselaustauschverfahrens zwischen zwei Instanzen nach dem Verfahren von Diffie-Hellman.
- PKCS #5 V 2.0 : 1999 Password-Based Cryptography Standard
Dieser PKCS-Standard enthält Empfehlungen zur Realisierung von Verschlüsselung, Schlüsselableitung und MAC-Bildung auf der Grundlage von Schlüsseln, die aus Passwörtern gebildet werden.
- PKCS #11 V 2.11 : 2001 Cryptographic Token Interface Standard
☒ *Dies ist die weltweite De-facto-Norm für eine API zum Aufruf von kryptografischen Funktionen. Die API hat den Namen „Cryptoki“ (cryptographic token interface) und beinhaltet Funktionen wie RC2, RC4, RC5, MD5, SHA-1, DES, Triple-DES, IDEA, RSA, DSA, MAC-Berechnung und Schlüsselerzeugung für die unterschiedlichen Kryptoalgorithmen.*
- PKCS #13 V 1.0 : 1998 Elliptic Curve Cryptography Standard
- PKCS #14 V 1.0 Pseudorandom Number Generation Standard
- Proposal : 1998
Dieser nicht fertig gestellte und nicht sehr umfangreiche PKCS-Standard enthält Vorschläge zur Konzeption von Pseudozufallsgeneratoren.
- PKCS #15 V 1.1 : 2000 Cryptographic Token Information Format Standard
☒ *Dies ist die weltweite De-facto-Norm der notwendigen Datenelemente einer interoperablen Chipkarte für digitale Signatur. Der Standard enthält eine Beschreibung aller für eine Signaturkarte notwendigen Verzeichnisse und Dateien sowie eine ASN.1-Beschreibung aller in den Dateien gespeicherten Zertifikate, Schlüssel und sonstiger Verwaltungsinformationen.*

OCF - API Docs V1.2 : 2001

OCF - Programmer's Guide V 1.2 : 2001

Open Platform Card Specification 2.1 : 2001

☒ *Dies ist die wichtigste Spezifikation zum Thema Verwaltung von Applikationen auf Multiapplikations-Chipkarten. Die sehr umfangreiche Spezifikation enthält eine detaillierte Darstellung der Software- und Sicherheits-Architektur von Multiapplikations-Chipkarten sowie eine ausführliche Beschreibung von dazu notwendigen Kommandos. Der Anhang enthält unter anderem die Spezifizierung einer API für das Applikationsmanagement auf Java Cards, was zum De facto-Standard bei diesem Typ von Chipkarten avanciert ist.⁷⁴*

RFC 1319 : 1992

The MD2 Message-Digest Algorithm

RFC 1320 : 1992

The MD4 Message-Digest Algorithm

RFC 1321 : 1992

The MD5 Message-Digest Algorithm

RFC 1750 : 1994

Randomness Recommendations for Security

Darstellung der Funktionsprinzipien von unterschiedlichen Zufallszahlengeneratoren. Aufbauend darauf werden Empfehlungen für Verfahren zur Konzeption einer hochwertigen Pseudozufallszahlengenerierung für PCs gegeben.

RFC 2706 : 1999

ECML V1: Field Names for E-Commerce

SET Book 1, Version 1.0 : 1997

Secure Electronic Transaction Specification, Book 1: Business Description

SET Book 2, Version 1.0 : 1997

Secure Electronic Transaction Specification, Book 2: Programmer's Guide

SET Book 3, Version 1.0 : 1997

Secure Electronic Transaction Specification, Book 3: Formal Protocol Definition

TIA/EIA/IS-820 : 2000

Removable User Identity Module (R-UIM) for TIA/EIA Spread Spectrum Standards

TIA/EIA/IS-820-1 : 2001

Removable User Identity Module (R-UIM) for TIA/EIA Spread Spectrum Standards, Addendum 1

TIA/EIA/IS-839 : 2000

R-UIM Overview, Operation, and File Structure Support in TIA/EIA-136

TR 33.900, V 1.2.0 : 2000

3rd Generation Partnership Project; Technical Specification Group SA WG3; A Guide to 3rd Generation Security Architecture
Überblick zu Sicherheitsarchitektur, vorhandenen Sicherheitsfunktionen und möglichen Angriffsszenarien bei 3G-Mobilfunknetzen. Die sicherheitstechnischen Details sind in den entsprechenden Normen TS 33.102, TS 33.120 und TS 21.133 beschrieben.

TS 21.111, Version 4.0.0 : 2001

3rd Generation Partnership Project; Technical Specification Group Terminals; USIM and IC card requirements
Kurze Norm mit den Festlegungen der grundsätzlich notwendigen Funktionalitäten eines Sicherheitsmoduls, d. h. einer USIM, für ein UMTS-Mobilfunknetzwerk. Diese Norm ist das UMTS-Äquivalent zur GSM-Norm GSM 02.17.

⁷⁴ Siehe auch Abschnitt 5.11 „Open Platform“.

- TS 21.133, Version 4.0.0 : 2001
Universal Mobile Telecommunications System (UMTS); 3G Security; Security Threats and Requirements
- TS 22.038, Version 4.1.0 : 2001
3rd Generation Partnership Project; Technical Specification Group Terminals; USIM/SIM Application Toolkit (USAT, SAT); Service description; Stage 1
- TS 22.112, Version 5.0.0 : 2001
Technical Specification; 3rd Generation Partnership Project; Technical Specification Group Terminals; USAT Interpreter – Stage 1
- TS 23.038, V 4.3.0 : 2001
3rd Generation Partnership Project; Technical Specification Group Terminals; Alphabets and language-specific information
Festlegung der Zeichen-Codierung bei SMS, USSD und der bei UMTS benutzten Zeichensätze.
- TS 23.040, V 4.3.0 : 2001
3rd Generation Partnership Project; Technical Specification Group Terminals; Technical realization of the Short Message Service (SMS)
- TS 23.048, Version 5.1.0 : 2001
3rd Generation Partnership Project; Technical Specification Group Terminals; Security Mechanisms for the (U)SIM application toolkit; Stage 2
- TS 31.102, Version 4.2.0 : 2001
3rd Generation Partnership Project; Technical Specification Group Terminals; Characteristics of the USIM Application
Spezifikation der logischen Eigenschaften der Chipkarten-Anwendung USIM, wobei diese durch eine Beschreibung der Schnittstelle zwischen USIM und dem UMTS-Mobiltelefon realisiert ist. Die TS 31.102 beinhaltet eine detaillierte Beschreibung aller Dateien mit ihren Datenelementen, die Definition einiger weniger USIM-spezifischen Kommandos und eine Aufstellung von Kommandosequenzen für typische Abläufe. Zusammen mit der TS 31.101 ist diese Norm das UMTS-Äquivalent zur GSM-Norm GSM 11.11.
- TS 31.110, Version 4.0.0 : 2001
3rd Generation Partnership Project; Technical Specification Group Terminals; Numbering system for telecommunication IC card applications
Diese Norm wird in Zukunft als TS 101.220 veröffentlicht.
- TS 31.111, Version 4.4.0 : 2001
3rd Generation Partnership Project; Technical Specification Group Terminals; USIM Application Toolkit (USAT)
Definition und ausführliche Beschreibung des USIM Application Toolkits für USIMs. USAT beschreibt eine Schnittstelle zwischen Mobiltelefon und USIM zur teilweisen Steuerung des Mobiltelefons von Zusatzanwendungen der USIM aus. Die Norm führt proaktive Kommandos für die USIM ein und definiert viele neue Kommandos im Zusammenhang mit der Steuerung des Mobiltelefons, wie beispielsweise Displayausgabe, Tastaturabfrage oder Versenden von Kurznachrichten. Das GSM-Äquivalent zu dieser Norm ist die GSM 11.14.
- TS 31.112, Version 5.0.0 : 2001
3rd Generation Partnership Project; Technical Specification Group Terminals; USAT Interpreter Architecture Description; Stage 2
- TS 31.113, Version 5.0.0 : 2001
3rd Generation Partnership Project; Technical Specification Group Terminals; USAT Interpreter Byte Codes

TS 31.114, Version 1.1.0 : 2002

3rd Generation Partnership Project; Technical Specification Group Terminals; USAT Interpreter Protocol and Administration

TS 31.121, Version 4.0.0 : 2001

3rd Generation Partnership Project; Technical Specification Group Terminals; UICC-Terminal Interface; USIM Application Test Specification

TS 31.122, Version 3.0.0 : 2000

3rd Generation Partnership Project; Technical Specification Group Terminals; USIM Conformance Test Specification
Festlegung von Testumgebung, Testausrüstung, Testhierarchie und der einzelnen Testfälle für das Testing von USIMs. Die beschriebenen Tests zielen ausnahmslos auf die elektrischen und informationstechnischen Aspekte. Dazu sind Tests von der elektrischen Versorgung über Datenübertragung, Dateiverwaltung, Kommandos bis hin zu typischen Abläufen in der UMTS-Anwendung im Detail festgelegt. Diese Norm zeigt sehr gut und ausführlich, wie USIM Tests beschrieben, aufgebaut und durchgeführt werden können. Diese Norm ist das USIM-Äquivalent zur SIM-Testnorm GSM 11.17.

TS 31.900, Version 3.1.0 : 2001

3rd Generation Partnership Project; Technical Specification Group Terminals; SIM/USIM Internal and External Interworking Aspects

TS 33.102, Version 4.1.0 : 2001

3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Security Architecture
Zentrale Norm für die gesamte Sicherheitsarchitektur eines UMTS-Mobilfunknetzwerks bezüglich Netzwerkzugriff, Authentisierung, Vertraulichkeit und Datenintegrität. Diese Norm enthält unabhängig von einem bestimmten Kryptoalgorithmus eine vollständige Beschreibung der Netzwerksicherheitsfunktionen, Authentisierungsprotokolle, Verschlüsselungsverfahren sowie der Erzeugung der Authentisierungsvektoren und der verwendeten Schlüsselableitungen.

TS 33.103, Version 4.1.0 : 2001

3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G security; Integration guidelines

TS 33.105, Version 4.1.0 : 2001

3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Cryptographic Algorithm Requirements

TS 33.120, Version 4.0.0 : 2001

Universal Mobile Telecommunications System (UMTS); 3G Security; Security Principles and Objectives

TS 35.205, Version 4.0.0 : 2001

Universal Mobile Telecommunications System (UMTS); 3G Security; Specification of the MILENAGE algorithm set: An example algorithm Set for the 3GPP Authentication and Key Generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 1: General

TS 35.206, Version 4.0.0 : 2001

Universal Mobile Telecommunications System (UMTS); 3G Security; Specification of the MILENAGE algorithm set: An example algorithm Set for the 3GPP Authentication and Key Generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 2: Algorithm Specification

TS 35.207, Version 4.0.0 : 2001

Universal Mobile Telecommunications System (UMTS); 3G Security; Specification of the MILENAGE algorithm set: An example algorithm Set for the 3GPP Authentication and Key Generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 3: Implementors' Test Data

TS 35.208, Version 4.0.0 : 2001

Universal Mobile Telecommunications System (UMTS); 3G Security; Specification of the MILENAGE algorithm set: An example algorithm Set for the 3GPP Authentication and Key Generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 4: Design Conformance

TS 35.909, Version 4.0.0 : 2001

Universal Mobile Telecommunications System (UMTS); 3G security; Report on the design and evaluation of the MILENAGE algorithm set; Deliverable 5: An example algorithm for the 3GPP Authentication and Key Generation functions

TS 42.009, V4.0.0 : 2001

3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Digital cellular telecommunications system (Phase 2+); Security aspects
Grundlagendokument mit einem Überblick über die wichtigen Sicherheitsaspekte eines PLMN.

TS 51.011, Version 4.2.0 : 2001

3rd Generation Partnership Project; Technical Specification Group Terminals; Specification of the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface

Diese Norm ist nach dem neuen ETSI-Nummern-schema die GSM 11.1.

TS 101.220, V 4.0.0 : 2001

Integrated Circuits Cards (ICC); ETSI numbering system for telecommunication application providers
Festlegung der AIDs, PLX und TAR für SIM, TETRA-SIM und USIM. Außerdem ist der Codierungsraum der PLX für die unterschiedlichen Arten von Zusatzanwendungen dieser Telekommunikations-Chipkarten definiert.

TS 102.221,
Version 4.4.0 : 2001

Smart cards; UICC-Terminal interface; Physical and logical characteristics

☒ *Spezifikation der physikalischen und logischen Eigenschaften einer USIM, wobei diese durch eine Beschreibung der Schnittstelle zwischen USIM und dem UMTS-Mobiltelefon realisiert ist. Die Spezifikation beinhaltet die Definition der Kartengrößen ID-1 und Plug-In, der mechanischen Rahmenparameter für die Karte und die Kontakte. Weiterhin ist in dieser Spezifikation die Festlegung aller elektrischen Rahmenwerte sowie Struktur und Dateninhalt von ATR und PPS enthalten, sowie die Definition der Übertragungsprotokolle, der Dateistrukturen, Sicherheitsmechanismen, Kommandos und Returncodes. Zusätzlich sind alle von einer bestimmten Telekommunikationsanwendung unabhängigen Dateien mit ihren Datenelementen sowie Kommandosequenzen aufgeführt. Diese Norm ist die Grundlage für ein Chipkarten-Betriebssystem für eine USIM und wird von der TS 31.102 mit allen anwendungsspezifischen Teilen für eine USIM ergänzt. Diese beiden Normen sind das UMTS-Äquivalent zur GSM-Norm GSM 11.11.*

TS 102.222,
Version 3.3.0 : 2001

Integrated Circuit Cards (ICC); Administrative commands for telecommunications applications

In dieser Spezifikation sind die administrativen Kommandos zur Verwaltung von Dateien sowie die dazugehörigen Sicherheitsbedingungen für Telekommunikations-Chip-karten festgelegt.

TS 102.223, Version 4.1.0 : 2001

Smart cards; Card Application Toolkit (CAT)

Definition und ausführliche Beschreibung eines generischen Application Toolkits für Telekommunikations-Chipkarten. CAT beschreibt eine Schnittstelle zwischen Mobiltelefon und Chipkarte zur teilweisen Steuerung des Mobiltelefons von Zusatzanwendungen der Chipkarte aus. Die Norm definiert die Kommandos im Zusammenhang mit der Steuerung des Mobiltelefons, wie beispielsweise Displayausgabe, Tastaturabfrage oder Versenden von Kurznachrichten. Auf dieser Grundlagennorm baut beispielsweise GSM 11.14 und TS 31.111 auf.

TS 102.224, Version 1.0.0 : 2001

Smart cards; Security mechanisms for the Card Application Toolkit; Functional requirements

TS 102.225, Version 1.0.0 : 2001

Smart cards; Secured packet structure for UICC applications

TS 102.226, Version 1.0.0 : 2001

Smart cards; Remote APDU Structure for UICC based Applications

TS 102.230, Version 4.0.0 : 2001

Smart cards; UICC-Terminal Interface; Physical, Electrical and Logical Test Specification

Festlegung der physikalischen und elektrischen Tests für UICCs. Weiterhin sind grundlegende Tests für die Kommunikationsverbindung zur UICC und Tests zu den beiden Übertragungsprotokollen $T = 0$ und $T = 1$ beschrieben.

TS 102.240, Version 1.0.0 : 2001

Smart Cards; UICC Application Programming Interface (UICC API); Service description

TS 102.241, Version 1.0.0 : 2001

UICC Application Programming Interface (UICC API); UICC API for Java Card

TS 102.241, Version 4.4.0 : 2002

Universal Mobile Telecommunications System (UMTS); Network architecture

Unicode Standard, Version 3.1.1 : 2001

Universal Serial Bus Specification, Revision 2.0, 2000

Diese sehr umfangreiche Spezifikation ist die Grundlage für die USB-Schnittstelle.

Wireless Application Protocol Identity Module Specification, Version 260 : Juli 2001

Spezifikation der physikalischen und logischen Eigenschaften einer WIM, der digitalen Signaturanwendung für Telekommunikations-Chipkarten. Es sind alle für eine WIM-Anwendung notwendigen Mechanismen, Kommandos, Datenelemente und Dateien aufgeführt.

16.6 Codierung von Datenobjekten

Das Kapitel 5 über ISO/IEC 7816-9-konforme Zugriffe auf Chipkarten-Ressourcen enthält weitere Tabellen über Kennzeichen (*tags*) für Datenobjekte.

16.6.1 Datenobjekte nach ISO/IEC 7816-4

Tabelle 16.2 Codierung einiger wichtiger Datenelemente nach ISO/IEC 7816-4.

Kennzeichen (tag)	Datenelement	Template	Länge [Byte]	Beschreibung
'62'	FCP (<i>file control parameters</i>) Template	---	---	Das FCP Template enthält Dateikontrollparameter (file control parameter – FCP).
'64'	FMD (<i>file management data</i>) Template	---	---	Das FMD Template enthält Dateiverwaltungsparameter (<i>file management data</i> – FMD).
'6F'	FCI (<i>file controll information</i>) Template	---	---	Das FMD Template enthält Dateikontrollparameter (<i>file control parameter</i> – FCP) und Dateiverwaltungsparameter (<i>file management data</i> – FMD).
'80'	Anzahl der Datenbytes in der Datei ohne Strukturinformation	'62'	2	für EFs der Struktur transparent
'81'	Anzahl der Datenbytes in der Datei mit Strukturinformation	'62'	2	für alle Dateien
'82'	Dateibeschreibung	'62'	1 ... 4	Dateizugriff °00-- ----°: Datei nicht shareable (d. h. kein gleichzeitiger Zugriff über verschiedene logische Kanäle) °01-- ----°: Datei shareable (d. h. gleichzeitiger Zugriff über verschiedene logische Kanäle möglich) Dateityp °--00 0---°: EF working °--00 1---°: EF internal °--11 1---°: DF EF Struktur °---- -000°: keine Information °---- -001°: transparent °---- -010°: linear fixed °---- -011°: linear fixed, Inhalt Simple-TLV codiert °---- -100°: linear variable °---- -101°: linear variable, Inhalt Simple-TLV codiert °---- -110°: cyclic °---- -111°: cyclic, Inhalt Simple-TLV codiert
'83'	FID	'62'	2	für alle Dateien

Tabelle 16.2 (Fortsetzung)

Kennzeichen (tag)	Datenelement	Template	Länge [Byte]	Beschreibung
'84'	DF Name	'62'	1 ... 16	für DFs
'86'	Sicherheitsattribute	'62'	variabel	
'88'	SFI (short file identifier)	'62'	1	Definition aus ISO/IEC 7816-9. für EFs SFI = b8 ... b4 °000°
'8A'	Status Indikator des Kartenlebenszyklus (life cycle status indicator, LCSi-tag)	'62'	1	Bitcodierung in ISO/IEC 7816-9 festgelegt. °0000 0000°: keine Information °0000 0001°: Creation State °0000 0011°: Initialisation State °0000 01x1°: operational state - activated °0000 01x0°: operational state - deactivated °0000 11xx°: termination state °yyyy xxxx°: y ≠ 0, proprietär

16.6.2 Datenobjekte nach ISO/IEC 7816-6

Tabelle 16.3 Codierung einiger wichtiger Datenelemente nach ISO/IEC 7816-6.

Kennzeichen (tag)	Datenelement	Template	Länge [Byte]	Beschreibung
'4F'	AID	'61' / '6E'	5 - 16	---
'50'	Anwendungsname	'61' / '6E'	variabel	
'59'	Verfallsdatum der Karte	'66'	3	Format: YYMMDD
'5B'	Name	'65'	max. 39	
'5F24'	Verfallsdatum der Anwendung	'6E'	3	Format: YYMMDD
'5F25'	Ausgabedatum der Karte	'6E'	3	Format: YYMMDD
'5F26'	Ausgabedatum der Anwendung	'66'	3	Format: YYMMDD
'5F28'	Länderkennung	'66'	2	Format: 3 Stellen, Codierung nach ISO 3166
'5F2A'	Währungskennung	'6E'	2	Format: 3 Stellen Codierung nach ISO 3166
'5F2B'	Geburtsdatum	'65'	4	Format: YYYYMMDD
'5F2B'	Geburtsdatum	'65'	4	Format: YYYYMMDD
'5F42'	Adresse	'65'	variabel	Adresse einer Person
'5F4D'	Chiphersteller	'66'	1	siehe Abschnitt 16.6.3 Datenobjekt für Chiphersteller nach ISO/IEC 7816-6

16.6.3 Datenobjekt für Chiphersteller nach ISO/IEC 7816-6

Tabelle 16.4 Codierung der Datenelemente für Chiphersteller nach ISO/IEC 7816-6. Diese Tabelle zeigt einen guten Überblick über die Hersteller von Chipkarten-Mikrocontroller. Das Kennzeichen (tag) für Chiphersteller ist '5F4D'.

Codierung	Firma
'01'	Motorola
'02'	ST Microelectronics
'03'	Hitachi
'04'	Philips Semiconductors
'05'	Infineon Semiconductors
'06'	Cyline
'07'	Texas Instrument
'08'	Fujitsu
'09'	Matsushita Electronic
'0A'	NEC
'0B'	Oki Electric
'0C'	Toshiba
'0D'	Mitsubishi Electric
'0E'	Samsung Electronics
'0F'	Hyundai Electronics Industries
'10'	LG-Semiconductors
'11'	Emosyn-EM Microelectronics
'12'	Inside Technologies
'13'	ORGA Kartensysteme
'14'	Sharp Corporation
'15'	ATMEL
'16'	EM Microelectronic-Marin
'17'	KSW Microtec
'19'	Xicor

16.7 Registrierungsstellen für RID

Das für die Registrierung eines RID benötigte Formular befindet sich im Anhang der ISO/IEC 7816-5. Der Antrag für eine internationale RID verläuft in der Regel über die jeweilige nationale Registrierungsinstanz und ist kostenpflichtig. Die Adressen der nationalen Registrierungsinstanz sowie die entsprechenden Registrierungsverfahren für RIDs können in der Regel bei der jeweiligen nationalen Normungsinstanz erfragt werden.

Tabelle 16.5 Registrierungsstellen für RIDs nach ISO/IEC 7816-5.

Bereich	Organisation
International	TeleDanmark KTAS att/ ISO/IEC 7816-5 Registration Authority Teghlolmsgade 1 1790 Copenhagen V Dänemark
Deutschland	RID German National Registration Authority c/o GMD Bruno Struif Rheinstraße 75 D-64 295 Darmstadt

16.8 Ausgewählte RIDs

Die folgende Tabelle zeigt einige Beispiele veröffentlichter RIDs bzw. AIDs. Leider werden die RIDs von den Registrierungsstellen vertraulich behandelt, so dass diese Liste nicht sehr umfangreich ist.

Tabelle 16.6 Auswahl von RIDs typischer Chipkarten-Anwendungen bzw. Instanzen, die Chipkarten einsetzen.

Chipkarten-Anwendung	AID (<i>application identifier</i>) = RID PIX
3GPP (UICC, USIM, USAT)	RID = 'A0 00 00 00 87' PIX = spezifisch für Kartenherausgeber
cc-Karte mit Chip für Deutschland	RID = 'D2 76 00 00 25' PIX = '45 50 01 00'
ETSI RID (SIM, SAT, Java Card SIM API, TETRA)	RID = 'A000000009'
FINEID (finnischer Personalausweis)	RID = 'A0 00 00 00 63' PIX = '50 4B 43 53 2D 31 35'
Giesecke & Devrient	RID = 'D2 76 00 00 05'
PKCS #15	RID = 'A0 00 00 00 63' PIX = '50 4B 43 53 2D 31 35' = "PKCS-15"
WIM	RID = 'A0 00 00 00 63' PIX = '57 41 50 2D 57 49 4D' = "WAP-WIM"
Wolfgang Rankl	RID = 'D2 76 00 00 60'

16.9 Veranstaltungen

Die folgenden Veranstaltungen und Messen haben mindestens einen ihrer Schwerpunkte im Bereich der Chipkarten oder auf einem den Karten verwandten Gebiet. Die aufgeführten Termine und Orte sind typische Werte der letzten Jahre, die sich aber je nach Veranstalter in der Zukunft verschieben können. Man sieht an dieser Liste, dass ein interessierter Reisender durchaus die Möglichkeit hätte, jeden Monat in einem anderen Land eine Veranstaltung zu besuchen.

Tabelle 16.7 Auswahl von jährlichen stattfindenden Veranstaltungen mit dem Schwerpunkt Chipkarten und Kryptografie

Name der Messe/Veranstaltung	Ort	Zeitraum
Asia Crypt	Asien	Herbst
Card Tech/Secure Tech [CTST]	USA	September
Cards Africa	Südafrika, Johannesburg	November
Cards Asia	Singapore	Februar
Cards Australia	Australien, Melbourne	August
Java One	San Francisco, USA	Juni
Cards Latin America	Chile, Santiago de Chile	Juli
Cartes	Frankreich, Paris	Oktober
CeBit	Deutschland, Hannover	März
Crypto	USA, Kalifornien, Santa Barbara	Sommer
Euro Crypt	Europa	Frühjahr
GSM World Congress	Cannes, Frankreich	Februar
OmniCard	Deutschland, Berlin	Januar
Smart Card	Großbritannien, London	Februar

16.10 World-Wide-Web-Adressen

Die folgende Liste von World-Wide-Web-Adressen erhebt keinen Anspruch auf Vollständigkeit und sollte als Querschnitt durch die unterschiedlichen Firmen und Institutionen gesehen werden, die auf dem Gebiet der Chipkarten aktiv sind. Die genannten Adressen sind auch als Startpunkte für weitere Recherchen gedacht. Aufgrund der Hypertextstruktur von HTML-Dokumenten befinden sich an vielen der angegebenen Adressen Verbindungen zu anderen interessanten Dokumenten und Plätzen im World Wide Web. Größere Sammlungen von Adressen sind als „Link-Farm“ explizit gekennzeichnet.

Es darf bei der Benutzung dieser Liste jedoch nicht vergessen werden, dass die angegebenen Internet-Adressen aufgrund der im Internet herrschenden Dynamik sehr schnell überholt sind. Dies ist auch der Grund, warum keine Referenzen auf einzelne Dokumente gemacht wurden, sondern maximal Unterverzeichnisse angegeben sind. Selbst diese werden aber oft bei Reorganisationen der Webserver geändert, weshalb es sich im Zweifelsfalle empfiehlt, die Adresse bis zum Organisations- bzw. Länderkennzeichen (*.com, *.de, ...) zu benutzen und dann von dieser aus die jeweils gültigen Verzeichnisse auf dem Webserver manuell über die Einstiegsseite zu selektieren.

In der Regel kann über die Homepage einer Firma auch deren Postanschrift sowie die Telefonnummer eines Ansprechpartners herausgefunden werden, weshalb es kein explizites Verzeichnis von Firmenadressen gibt. Bei konkretem Bedarf an einer bestimmten Information geben wir prinzipiell den Rat, mit den entsprechenden Suchbegriffen und der Zuhilfenahme einer leistungsfähigen Suchmaschine das World Wide Web zu durchkäm-

men. Man kann dann auf jeden Fall sicherstellen, über einen aktuellen Querschnitt an Informationen für die Recherche zu verfügen.

Die Zuordnung der Adressen bzw. Firmen wurde aufgrund des hauptsächlichlichen Tätigkeitsbereiches getroffen. Viele der aufgeführten Firmen haben jedoch Aktivitäten in mehreren aufgeführten Gebieten; diese sind in der Regel explizit angegeben. Soweit sinnvoll, ist zusätzlich noch das Land der Firma bzw. der Institution genannt. Eine in unregelmäßigen Zeitabständen aktualisierte Liste, deren Hyperlinks man von einem Browser aus folgen kann, findet sich bei [Rankl].

Tabelle 16.8 Überblick über die verwendeten Kategorien bei der Liste der Web-Adressen.

Kategorie	Erklärung
Angriffe	Angriffe auf Chipkarten, Chipkartenterminals, Sicherheitsmodule u. Ä.
Betriebssysteme	Betriebssysteme für Chipkarten
Hersteller von Chipkarten-Mikrocontrollern	Hersteller von Halbleitern für Chipkarten, Speicherchips und/oder Mikrocontroller
Kartenherausgeber	Herausgeber von Karten und/oder Chipkarten
Kartenhersteller	Hersteller von Karten mit und/oder ohne Chip
Kryptografie	Kryptografie im Umfeld von Chipkarten
Link-Farm	Sammlung von Links auf weitere Adressen im Internet
Maschinen für Kartenfertigung	Maschinen und Geräte für die Kartenfertigung
Normen	Normung im Umfeld von Chipkarten und Kryptografie
Patente	Patente im Umfeld von Chipkarten
Sicherheitstechnik	Sicherheitstechnik im Umfeld von Chipkarten
Software	PC-Software für Chipkarten, Chipkartensimulationen
Terminalhersteller	Hersteller von Terminals für Karten mit und ohne Chip
Universität	Universität oder Institute von Universitäten
Veranstaltungen	Veranstaltungen, Kongresse und Konferenzen im Umfeld von Chipkarten
Verlag	Zeitschriften und Bücher über Chipkarten

[3GPP]

3GPP

Normen

<http://www.3gpp.org/>

[3GPP2]

3GPP2

Normen

<http://www.3gpp2.org/>

[a la Card]

a la Card, Deutschland

Verlag im Chipkartenbereich

<http://www.alacard.de/>

[AC]	Austria Card, Deutschland Kartenhersteller http://www.austriacard.at/
[ACG]	ACG, Deutschland Chiphändler http://www.acg.de/
[Activ Card]	Activ Card, USA Kartenhersteller http://www.activcard.com/
[Aliro]	Aliro Ltd., Israel Sicherheitstechnik http://www.aliro.com/
[AM]	American Magnetics, USA Terminalhersteller http://www.magstripe.com/
[AmEx]	American Express, USA Kartenherausgeber http://www.americanexpress.com/
[Anderson]	Ross Anderson's Home Page, Großbritannien Informationen über Angriffe auf Chipkarten http://www.cl.cam.ac.uk/users/rja14/
[ANSI]	ANSI, USA Normen http://www.ansi.org/
[ARM]	ARM Ltd., Großbritannien Prozessor Core für Chipkarten-Mikrocontroller http://www.arm.com/
[Ascom]	Ascom AG, Schweiz Kryptografie, IDEA http://www.ascom.ch/
[ASM]	ASM Lithography, Niederlande Maschinen für die Halbleiterproduktion http://www.asml.com/
[Atmel]	Atmel, USA Chipkarten-Mikrocontroller http://www.atmel.com/
[Basiccard]	Basic Card Chipkarten-Betriebssystem http://www.basiccard.com/
[BdB]	Bundesverband deutscher Banken (BdB), Deutschland http://www.bdb.de/

[Bluetooth]	Bluetooth Informationen über Bluetooth http://www.bluetooth.com/
[BSI]	Bundesamt für Sicherheit in der Informationstechnik (BSI), Deutschland Informationen zum Thema Sicherheit http://www.bsi.bund.de/
[Card Forum]	Card Forum, Deutschland Verlag im Chipkartenbereich http://www.card-forum.com/
[Cardshow]	The Smart Card Cyber Show, Frankreich http://www.cardshow.com/
[CC]	Common Criteria http://www.commoncriteria.org/
[CCC]	Chaos Computer Club e.V., Deutschland Angriffe auf Chipkarten und Kryptoalgorithmen http://www.ccc.de/
[CDG]	CDMA Development Group (CDG) Informationen über CDMA http://www.cdg.org/
[CEN]	CEN Normen http://www.cenorm.be/
[CEPS]	CEPSCO Informationen über CEPS http://www.cepsco.com/
[Certicom]	Certicom Corp., Kanada Kryptografie, ECC http://www.certicom.ca/
[Counterpane]	Counterpane, USA Kryptografie http://www.counterpane.com/
[CR]	Cryptography Research Angriffe http://www.cryptography.com/
[CTST]	CardTech/SecurTech Konferenz, USA Veranstaltungen über Chipkarten http://www.ctst.com/
[Dai Nippon]	Dai Nippon Printing Co. Ltd., Japan Chipkartenhersteller http://www.dnp.co.jp/