gematik
Gesellschaft für Telematikanwendungen der Gesundheitskarte mbH

Einführung der Gesundheitskarte

Spezifikation der elektronischen Gesundheitskarte

Teil 1: Kommandos, Algorithmen und Funktionen der Betriebssystem-Plattform

Version:	1.2.0
Stand:	24.08.2007
Status:	freigegeben

Dokumentinformationen

Änderungen zur Vorversion

Diese Version unterscheidet sich von der Version 1.1.0 dadurch, dass zum Stichtag 21.08.2007 alle SRQs, die Auswirkungen auf dieses Dokument besitzen, eingearbeitet wurden (siehe Dokumentenhistorie).

Inhaltliche Änderungen gegenüber der letzten freigegebenen Version sind gelb markiert. Sofern ganze Kapitel eingefügt wurden, wurde zur besseren Lesbarkeit lediglich die Überschrift durch gelbe Markierung hervorgehoben.

Referenzierung

Die Referenzierung in weiteren Dokumenten der gematik erfolgt unter:

[gemSpec_eGK_P1] gematik (24.08.2007): Einführung der Gesundheitskarte – Die Spezifikation elektronische Gesundheitskarte; Teil 1 – Kommandos, Algorithmen und Funktionen des Kartenbetriebssystems Version 1.2.0, www.gematik.de

Dokumentenhistorie

Version	Stand	Kap./ Seite	Grund der Änderung, besondere Hinweise	Bearbeitung
V0.1	08.06.05		1. von der gematik herausgegebene Version	gematik
V0.2	01.07.05		- kleinere redaktionelle Änderungen	gematik
V0.3	14.07.05		- Tab. 6 range for public exponents récised - Tab. 7 value of P2 for file commands récised - Selection status after execution of file commands récised - Access rule for LOAD APPLICATION récised - Usage of signature counters added - New clause 10 related to security status and situation after DF selection - A Key may be used for several purposes - DEACTIVATE RECORD récised - Status codes for CREATE FILE modified	gematik
V 0.4	16.08.05		Übersetzung in die deutsche Sprache Anpassung ICCSN an europäische Resolution 190 Entfernen der Kommandos „LOAD APPLICATION“ und DEACTIVATE RECORD“ Kennzeichnung der Kommandos für Schrei-	gematik

Version	Stand	Kap./ Seite	Grund der Änderung, besondere Hinweise	Bearbeitung
			ben/Nachladen Bibliographie gelöscht	
V 0.9	31.08.05		neue Einleitung nur Änderungskennzeichnung Einarbeitung Kommentare Präzisierung Algorithmen und Hash-Verfahren Überarbeitung Kapitel Zulassung und Evaluierung	gematik
V0.99	31.10.05		Dokumentenmodell aktualisiert Einarbeitung Kommentare gSP3 Kennzeichnung offener Punkte Einarbeiten Kommandos LOAD APPLICATION, DEACTIVATE RECORD Einarbeiten CVC-Verfahren Präzisierung QES-Komplettierung Ergänzung Anhang mit Authentisierungsverfahren Einarbeitung von Änderungen aus Workshop 25.10.05 Einarbeitung von Änderungen aus Workshop BÄK 26.10.05 Zeichnung E.6 ergänzt im Vor-Kommentierungsverfahren veröffentlicht	gematik
V1.0	05.12.05		- Verhalten der eGK bei Kommandoabbruch ergänzt - PIN-Management präzisiert - MSE-Kommando präzisiert - Anhang E ergänzt - nur ein kartenbasiertes CV-Zertifikat mit OID für Authentisierung mit/ohne TC (wie in HPC und SMC) - Client/Server-Authentisierungsverfahren präzisiert Parameter für „GENERATE ASYMMETRIC KEY PAIR“ überarbeitet redaktionelle Änderungen	gematik
V1.1	07.02.06		Kleinere Korrekturen aus dem Kommentierungsverfahren im Zeitraum 05.12.05-16.01.06 (siehe gelbe Markierungen), redaktionelle Änderungen	gematik
1.1.1	26.07.07		Einarbeitung SRQs • gelbe Markierungen entfernt Folgende SRQs eingearbeitet • SRQ_0509 PSO Decipher • SRQ_0511 public Exponent	gematik, AFI

Version	Stand	Kap./ Seite	Grund der Änderung, besondere Hinweise	Bearbeitung
			• SRQ_0513 Tag Reihenfolge MSE • SRQ_0601 Verkettung PSO Verify Cert • SRQ_0602 ASS bei MSE Restore • SRQ_0603 Einfluss MSE Set auf andere DF • SRQ_0605 zusätzliche Statusworte • SRQ_0608 Bufferlänge mit und ohne SM • SRQ_0609 ab wann ist TC eingerichtet • SRQ_0611 SM und Notwendigkeit CT • SRQ_0612 SM-Layer • SRQ_0613 PIN falsch im letzten Versuch • SRQ_0614 Kommandos innerhalb TC • SRQ_0615 Statuswort deaktiviertes EF • SRQ_0616 Statuswort deaktiviertes EF • SRQ_0617 Statuswort ungültiger SM Key • SRQ_0619 Statuswort PSO Verify Cert • SRQ_0620 Change Ref Data Varianten • SRQ_0621 Get und Put Data • SRQ_0622 PSO Encipher • SRQ_0623 Nutzungszähler PUK • SRQ_SD05_0280 Nutzungszähler PUK • SRQ_SD05_0290 Statuswörter bei Doppelfehler Folgende SRQs NICHT eingearbeitet, da sie keine Auswirkungen auf den normativen Teil haben • SRQ_0600 FCP Daten • SRQ_0604 DSI Formate • SRQ_0607 ´61 XX´ • SRQ_0610 SM und Notwendigkeit MAC • SRQ_SD05_0120 LCS und SE	
1.2.0	24.08.07		freigegeben	gematik

Inhaltsverzeichnis

1 Erläuterungen zum Inhalt der Dokumente

Die Dokumentation für die elektronische Gesundheitskarte besteht aus mehreren technischen Spezifikationen, ergänzenden Dokumenten und organisatorischen Festlegungen. Die ergänzenden Dokumente definieren die in den Spezifikationen beschriebenen Verfahren sowie die Handhabung der Zertifikate.

Die vorliegende Spezifikation wird auch in englischer Sprache veröffentlicht. In Zweifelsfällen ist die deutsche Version verbindlich.

1.1 Technische Spezifikationen zur eGK

- **Die Spezifikation der elektronischen Gesundheitskarte**
  Teil 1: Kommandos, Algorithmen und Funktionen der Betriebssystem-Plattform

  Im Teil 1 werden die Basiskommandos, die Grundfunktionen des Betriebssystems sowie die grundlegenden Sicherheitsfunktionen und -algorithmen (hard facts) detailliert beschrieben.

  Diese Spezifikation ist Grundlage der Entwicklung der Kommandostrukturen und –funktionen für eGK-konforme Chipkartenbetriebssysteme; sie ist somit die Grundarchitektur für die ROM-Maske des Halbleiters.

- **Die Spezifikation der elektronischen Gesundheitskarte**
  Teil 2: Anwendungsspezifische Strukturen

  Im Teil 2 werden die anwendungsspezifischen Strukturen der eGK beschrieben. Dieser Teil enthält die Spezifikationen für die Dateistrukturen der Pflichtanwendungen und der zugehörigen Datenelemente, die bei der Initialisierung und Personalisierung in die eGK geladen werden.

  Insbesondere sind hierin die Dateistrukturen der Anwendungen „Versichertenmanagement“, „elektronisches Rezept“, „qualifizierte Signatur“ und „eSign Anwendung“ spezifiziert. Dazu gehören entsprechende statische Daten sowie die Strukturen und Datencontainer für Zertifikate und Schlüsselelemente.

- **Die Spezifikation der elektronischen Gesundheitskarte**
  Teil 3: Äußere Gestaltung

  Der Teil 3 beschreibt die äußere Gestaltung der eGK. Hier werden die Bereiche auf der eGK festgelegt, in denen das Lichtbild des Versicherten sowie seine Unterschrift vorgesehen sind. Die Kartenrückseite wird entsprechend den Vorgaben für die europäische Krankenversicherungskarte definiert.

1.2 Ergänzende Dokumente zur eGK

- **Datendefinition für die Datenübergabe**

  Es werden die Daten aufgeführt und beschrieben, die zur Herstellung einer eGK benötigt und die dem Kartenhersteller / Personalisierer übertragen werden. Die Rollen im Herstellungsprozess werden beispielhaft dargestellt.

- **Speicherbedarf der eGK**

  Die eGK stellt einen begrenzten Speicherplatz für die Speicherung von Daten, Zertifikaten und Schlüsseln zur Verfügung. Der sich aus den verschiedenen Anforderungen und Spezifikationen ergebende Netto-Bedarf wird tabellarisch dargestellt.

- **Festlegungen zu den X.509-Zertifikaten der Versicherten**

  Die Inhalte der personenbezogenen X.509-Zertifikate zur Authentifizierung, Verschlüsselung und qualifizierten Signatur werden detailliert dargestellt. Das Dokument trifft die erforderlichen Festlegungen zur Versichertenidentität, zur Schlüsselverwendung und zur Zertifikatsvalidierung.

- **Aktivierung der Signaturzertifikate in der eGK für qualifizierte elektronische Signaturen**

  Beschreibung der erforderlichen technischen Festlegungen zum nachträglichen Laden von qualifizierten Signaturzertifikaten auf die eGK unter Berücksichtigung der gesetzlichen Vorgaben.

- **Festlegung einer einheitlichen X.509-Zertifikatsinfrastruktur für die Telematik im Gesundheitswesen**

  Das Konzept zur flexiblen und vertrauenswürdigen Einbindung der verschiedenen Public-Key-Infrastrukturen durch die Schaffung einer „Trust Service List“ wird beschrieben. Diese ermöglicht eine zentrale Sammlung und Verteilung der Root-Zertifikate unter Einhaltung eines einheitlichen Sicherheitsniveaus.

1.3 Das Schalenmodell der Dokumentenstruktur

Die Dokumente der technischen Spezifikation zur elektronischen Gesundheitskarte eGK werden nach einem Schalenmodell gegliedert. Dieses Modell ist modular aufgebaut und strukturiert die Dokumente und Prozesse, die sowohl für die Herstellung der Karte als auch für die nachfolgende Initialisierung und Personalisierung relevant sind.

Der innere Bereich (Teil1) beinhaltet die Spezifikation, die aufgrund ihres Reifegrades als Basis für die Ausschreibung von Betriebssystem, Mikroprozessorchip und Kartenkörper ge-

eignet ist. Es handelt sich um die Basiskommandos, Sicherheitsfunktionen und - algorithmen sowie Grundfunktionen des Betriebssystems (sog. hard facts). Diese Spezifikation ist Grundlage der Entwicklung der Kommandostrukturen und –funktionen für eGK-konforme Chipkartenbetriebsysteme. Weiterhin definiert sie die ROM-Maske für den Halbleiter (Fertigungsmuster des Halbleiters). Teil 1 beinhaltet auch die Spezifikationen für die Sicherheitsfunktionen und kryptografischen Algorithmen der eGK.

Die nächste Schale, der Teil 2, enthält die Spezifikationen für die Dateistrukturen der Anwendungen und der zugehörigen Datenelemente. Diese werden bei der Initialisierung und Personalisierung auf die eGK gebracht. In Teil 2 sind insbesondere die Dateistrukturen der Anwendungen Versichertenmanagement, elektronisches Rezept, Notfalldaten, Protokollierung, qualifizierte Signatur und eSign Anwendung enthalten. Dazu gehören entsprechende statische Daten sowie die Strukturen und Datencontainer.

In der äußeren Schale sind die Spezifikationen für die Personalisierung enthalten. Diese beschreiben die Verfahren der Personalisierung, die zu personalisierenden Daten sowie sicherheitstechnische und organisatorische Vorraussetzungen hierfür.

Die ISO-konformen Dateistrukturen der eGK Anwendungen können ggf. nach Produktion der Karte verändert werden. Spezifikationen für weitere bzw. zukünftige Anwendungen, zugehörige Datenstrukturen und Datenelemente auf Basis der eGK können nach Fertigstellung in Teil 2 bzw. die äußere Schale eingestellt werden. Es besteht die Möglichkeit, zusätzliche Schalen zur Aufnahme weiterer Spezifikationen zu definieren.

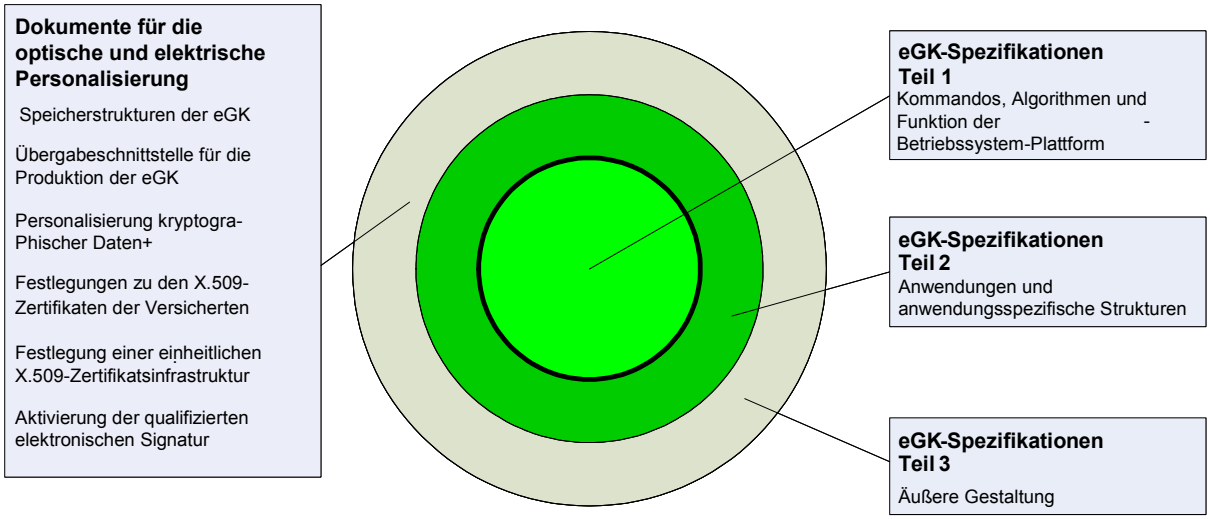

Abbildung 1 - Dokumentenstruktur

2 Einführung

2.1 Zielsetzung und Einordnung des Dokumentes

Diese Spezifikation definiert die Anforderungen an die Funktionalität einer Betriebssystem-Plattform für die elektronische Gesundheitskarte (eGK), die internationalen Standards entsprechen und die internationale und europäische Interoperabilität sicherstellen.

Im Einzelnen werden auf der Basis von [ISO7816-4], [ISO7816-8] und [ISO7816-9] Kommandos und Optionen beschrieben, die von der eGK unterstützt werden müssen.

2.2 Zielgruppe

Das Dokument richtet sich an Kartenhersteller.

2.3 Geltungsbereich

Der Inhalt des Dokumentes ist verbindlich für die Erstellung elektronischer Gesundheitskarten.

2.4 Arbeitsgrundlagen

Die Ausarbeitung steht in engem Zusammenhang mit der Spezifikation des Heilberufsausweises [HBA].

2.5 Abgrenzung des Dokumentes

Vergleiche hierzu Kap. 1.3.

2.6 Notation:

Hexadezimalzahlen werden mit Hochkomma dargestellt, z. B. '30'.

3 Kommandos

3.1 Konventionen und Parameter

Die folgende Tabelle beschreibt die Kommandos und Optionen, die vom Betriebssystem der Chipkarte zur Nutzung als eGK unterstützt werden müssen, wobei die Unterstützung weiterer Kommandos und Optionen nicht ausgeschlossen wird. Die obligatorisch zu unterstützenden Status-Codes sind in Anhang C beschrieben.

In allen Fällen, bei denen keine andere Festlegung vorliegt, werden Hexadezimalzahlen in der Big Endian Darstellung verwendet, d.h. das höchstwertige Byte (MSB) wird zuerst übertragen.

Das CLA-Byte der nachfolgenden Kommandos ist nach den in Kapitel 5.1.1 von ISO/IEC 7816-4 beschriebenen Konventionen zu codieren.

Tabelle 1 – Auswahl-Kommandos

INS	Name	P1	P2	C-Data Feld	R-Data Feld
'A4'	SELECT (File oder Application)	- '02': Wählt EF unter aktuellem DF - '04': Wählt DF über den Namen	- '04': Gibt FCP zurück - '0C': keine Daten in Antwort	P1 = '02': File ID P1 = '04': AID	FCP, wenn P2 = '04' und Le vorhanden
'A4'	SELECT (Root) siehe Anmerkung	- '00' (native Karten) - '04' (GP-Karten)	- '04': Gibt FCP zurück - '0C': keine Daten in Antwort	P1 = '00': '3F00' P1 = '04': nicht vorhanden	FCP, wenn P2 = '04' und Le vorhanden

Konventionen:

- Root ist bei nativen Karten das MF
- Root ist bei Global-Platform-orientierten Karten die "Default Selected Application"
- Das Kommando SELECT mit P1 = '00' und P2 = '04' muss immer im Klartext gesendet werden. Das COS KANN ein derartiges Kommando innerhalb eines TC akzeptieren oder ablehnen.

Tabelle 2 – Kommandos für die Bearbeitung von Daten

INS	Name	P1 - P2	C-Data Feld	R-Data Feld
‘B0’	READ BINARY	- Short EFID und offset - Offset	nicht vorhanden	Daten vorhanden, Le-Behandlung: siehe Kapitel 14
‘D6’	UPDATE BINARY	- Short EFID und offset - Offset	Daten, die geschrieben werden sollen	Absent

Wenn Short EFID benutzt wird, setzt das Kommando bei erfolgreicher Kommandoausführung das adressierte EF als aktuelles EF. Falls das Kommando mit Short EFID mit einem Fehler beendet wird und das adressierte EF bereits selektiert war, dann bleibt es aktuelles EF.

Falls das Kommando mit Short EFID mit einem Fehler beendet wird und das adressierte EF noch nicht selektiert war, ist der Kartenzustand unbestimmt, d.h. die externe Welt kann weder davon ausgehen, dass das adressierte EF selektiert wurde, noch kann sie davon ausgehen, dass ein zuvor selektiertes anderes EF immer noch aktuelles EF ist.

Ein Anwendungssystem soll daher bei einem nachfolgenden EF-Lese- oder Schreibbefehl das betreffende EF durch Setzen der SFID in dem Kommando selektieren.

Tabelle 3 – Kommandos für die Bearbeitung von Records

INS	Name	P1	P2	C-Data Feld	R-Data Feld
’B2’	READ RECORD	- Record-Nr.	- mit/ohne Short EFID - Record Nr. in P1	nicht vorhanden	- Le = '00': vollständiger Record, d.h. die Daten des Records ohne Rec-Nr. und Länge
‘DC’	UPDATE RECORD	- Record-Nr.	- mit/ohne Short EFID - Record Nr. in P1	Daten, die geschrieben werden sollen (die neue Länge bei variabler Record-Länge ist Lc)	nicht vorhanden
’E2’	APPEND RECORD	- ’00’	- mit/ohne Short EFID	Record, der hinzugefügt werden soll	nicht vorhanden
’A2’	SEARCH RECORD	- Record-Nr.	- xxxxx100 (einfache Suche vorwärts mit/ohne SFID xxxxx)	Suchzeichenfolge	- nicht vorhanden oder Record-Nummern
'06'	DEACTIVATE RECORD (siehe Anhang C)				

Wenn Short EFID benutzt wird, setzt das Kommando bei erfolgreicher Kommandoausführung das adressierte EF als aktuelles EF. Falls das Kommando mit Short EFID mit einem

Fehler beendet wird und das adressierte EF bereits selektiert war, dann bleibt es aktuelles EF.

Falls das Kommando mit Short EFID mit einem Fehler beendet wird und das adressierte EF noch nicht selektiert war, ist der Kartenzustand unbestimmt, d.h. die externe Welt kann weder davon ausgehen, dass das adressierte EF selektiert wurde, noch kann sie davon ausgehen, dass ein zuvor selektiertes anderes EF immer noch aktuelles EF ist.

Ein Anwendungssystem soll daher bei einem nachfolgenden EF-Lese- oder Schreibbefehl das betreffende EF durch Setzen der SFID in dem Kommando selektieren.

Tabelle 4 – Kommandos für die grundlegenden Sicherheitsfunktionen

INS	Name	P1	P2	C-Data Feld	R-Data Feld
'88’	INTERNAL AUTHENTICATE siehe Anmerkung 1	- '00'	- '00'	siehe Anhang E	
'84’	GET CHALLENGE siehe Anmerkung 2	- '00'	- '00'	nicht vorhanden (Le = 8)	8 Byte Zufallszahl
'82’	EXTERNAL AUTHENTICATE siehe Anmerkung 1	- '00'	- '00'	siehe Anhang E	nicht vorhanden
'82’	MUTUAL AUTHENTICATE siehe Anmerkung 1	- '00'	- '00'	siehe Anhang E	
'20’	VERIFY siehe Kapitel 6	- '00'	PIN-Referenz entsprechend [ISO7816-4], Tabelle 65	PIN im Format 2 PIN Block, siehe Kapitel 6	nicht vorhanden
'24’	CHANGE REF. DATA siehe Kapitel 6	- '00' - '01'	siehe VERIFY	siehe Kapitel 6	nicht vorhanden
'2C’	RESET RETRY COUNTER siehe Kapitel 6	- '00': Rücksetz-Code || neue PIN - '01': Rücksetz-Code	siehe VERIFY	1 oder 2 Format 2 PIN-Blöcke, siehe Kapitel 6	nicht vorhanden
'22’	MANAGE SE siehe u. a. Konventionen	- b8-b5 entsprechend [ISO7816-4], Tabelle 78 -SET -RESTORE	- SEID (Restore) - 'A4’: AT - 'B4’: CCT - 'B6’: DST - 'B8’: CT - 'AA’: HT	- nicht vorhanden (Restore) - DO '80’: AlgID - DO '83’: KID.SK - DO '83’: KID.PuK - DO '84’: KID.PrK	nicht vorhanden

Anmerkungen:

(1) Die Kommandos INTERNAL AUTHENTICATE, EXTERNAL AUTHENTICATE und MUTUAL AUTHENTICATION werden ggf. vor der Etablierung eines Trusted Channels verwendet, verwenden also aus Sicht des IFD immer Klartext. Verschlüsselte Daten sind damit für das IFD transparent (siehe Anhang E).
(2) Das Kommando GET CHALLENGE muss immer im Klartext gesendet werden. Das COS KANN GET CHALLENGE innerhalb eines TC akzeptieren oder ablehnen.

Für MSE gelten folgende Konventionen:

- Option RESTORE: Das Kommando MSE mit Option RESTORE MUSS immer im Klartext gesendet werden. Dabei wird nur die SE–Nummer des aktuellen DFs gesetzt. Die SE–Nummern aller anderen DFs (auch das MF ist ein DF) MÜSSEN unverändert bleiben. Das COS KANN ein mit SM gesichertes MSE RESTORE Kommando akzeptieren oder ablehnen. Der Sicherheitszustand des aktuellen DF (auch das MF ist ein DF) KANN verloren gehen oder erhalten bleiben. Die Außenwelt MUSS durch entsprechende Gestaltung der APDUs dafür sorgen, dass beide Varianten umgesetzt werden.

- Option SET: Das COS MUSS folgende Kommandodaten in der abgegebenen Reihenfolge akzeptieren:
  KeyID_symmetrischerSchlüssel
  KeyID_privaterSchlüssel
  KeyID_privaterSchlüssel || AlgID
  KeyID_öffentlicher Schlüssel
  KeyID_öffentlicher Schlüssel || AlgID
  KeyID_privaterSchlüssel || KeyID_öffentlicher Schlüssel
  KeyID_privaterSchlüssel || KeyID_öffentlicher Schlüssel || AlgID
  Das COS KANN andere Datenobjekte oder Reihenfolgen akzeptieren.
  Das COS KANN andere Datenobjekte oder Reihenfolgen als fehlerhaft ablehnen.

Werden mit einem Kommando zwei Schlüsselreferenzen gesetzt, dann bezieht sich eine evtl. vorhandene AlgID immer auf den geheimen Schlüssel. Ansonsten sind 2 getrennte MSE-Kommandos zu senden.

Ist das CRT das Hash Template HT, dann ist im Datenfeld die AlgID des Hash-Algorithmus anzugeben.

Tabelle 5 – Kommandos für Sicherheitsfunktionen

INS	Name	P1	P2	C-Data Feld	R-Data Feld
'46'	GENERATE ASYMMETRIC KEY PAIR siehe u. a. Konventionen	- '82': Generieren des asymmetrischen Schlüsselpaares und Ausgabe des Public Key - '83': Auslesen eines Public Key - '86': Generieren des asymmetrischen Schlüsselpaares ohne Ausgabe des Public Key	- '00'	nicht vorhanden	- Falls P1 = '82' oder '83' (Le muss vorhanden sein): Public Key, siehe u. a. Konventionen - Falls P1 = '86' (Le darf nicht vorhanden sein): keine Daten
'2A'	PSO: COMPUTE DS	- '9E'	- '9A'	- nicht vorhanden, d.h. Hash-Wert wurde zuvor mit PSO:HASH erzeugt - vorhanden und Signatur soll mit DSI gemäß [PKCS#1] erzeugt werden: DigestInfo gemäß [PKCS#1] Kapitel 9.2 - vorhanden und Signatur soll mit DSI gemäß [ISO9796-2] erzeugt werden: HashWert *Anmerkung: Die Länge des Signatur-Inputs (Digestinfo bzw. Hash-Wert) darf höchstens 40% der Moduluslänge des Signaturschlüssels betragen.*	Signatur

INS	Name	P1	P2	C-Data Feld	R-Data Feld
'2A'	PSO: HASH	- '90'	- 'A0'	- DO '90' (Hash-Zwischenwert = bisher berechneter Hash-Wert (x Byte) || Länge der gehashten Bits (y Byte)) || DO '80' (letzter Textblock), siehe u. a. Konventionen	nicht vorhanden
'2A'	PSO: VERIFY CERTIFICATE	- '00'	- 'AE'	- DO '5F37' || DO '5F38' ('5F37' = Signatur des Zertifikats, '5F38' = Public Key Remainder, siehe Anhang B)	nicht vorhanden
'2A'	PSO: DECIPHER	- '80'	- '86'	- Daten, die entschlüsselt werden sollen, siehe u. a. Konventionen	- entschlüsselte Daten

3.1.1 Konventionen für Schlüsselgenerierung:

- Die KeyId des privaten Schlüssels des zu generierenden Schlüsselpaars muss zuvor mit dem MSE-Kommando selektiert werden (CRT ist DST für die Generierung eines QES-Schlüsselpaars).
- Mit der KeyId sind die Schlüsselparameter in einer ausgegebenen Karte fest verknüpft.
- Die Schlüsselparameter müssen den Bedingungen von [ALGCAT] genügen.
- Die PuK-Daten für Public Key RSA sind [ISO7816-8]-konform nach einer explizit oder implizit COS-intern vorhandenen "extended headerlist" zu codieren: '7F49'-L-'81'-L-'xx ...xx' || '82 0x .... ' = DO Public Key Data Objects (DO Modulus || DO Public Exponent, z. B. 65537)
- Die vorgenannten Konventionen sind spätestens ab 2007 zu unterstützen.

3.1.2 Konventionen für Hashing:

Tabelle 6 – Werte für die Hash-Berechnung (relevant für Kommando PSO:HASH)

Hash Algorithmus	Länge des bisher berechneten Hash-Wertes (x Byte)	Länge des Hash-Ausgabewertes in Byte	Länge des Zählers (y Byte), der die Anzahl der bereits gehashten Bits angibt	Blocklänge des Hash-Alg. in Byte	OID
SHA-1 (Algorithmus muss in der eGK vorhanden sein)	20	20	8	64	{1 3 14 3 2 26}
SHA-256 (Realisierung in der eGK empfohlen)	32	32	8	64	{2 16 840 1 101 3 4 2 1}

Falls die Daten, die gehasht werden sollen, kürzer als die Blocklänge sind, soll die Länge des DO mit dem Tag ‘90’ auf Null gesetzt werden.

3.1.3 Konventionen für DECIPHER:

Der Verschlüsselungsalgorithmus ist RSA. Das erste Byte der Kommandodaten ist der Padding-Indikator mit dem Wert PI = '81'. Es folgen N=Moduluslänge Bytes. Die Antwortdaten MÜSSEN kleiner gleich N-11 Byte sein.

Tabelle 7 – Format für Key Encipherment Input

PI	Key Encipherment Input	Spezifikation
'81'	'00 02' || RND (alle Byte ungleich Null, Anzahl von Schlüssellänge abhängig) || '00' (Separator) || Zu entschlüsselnde Daten	[PKCS#1], Clause 7.2.1, "EME-PKCS1v1_5"

Tabelle 8 – Kommandos für das Kartenmanagement

INS	Name	P1	P2	C-Data Feld	R-Data Feld
'EA'	LOAD APPLICATION siehe [ISO7816-13] und Anmerkung. 1	- '00'	- '00'	DO mit dem Kommando, das auszuführen ist, tag '52'	Antwort-Daten oder nicht vorhanden
'E0'	CREATE FILE (DF) siehe Anmerkung 2 und 3	- '38'	- '00': keine Info	File-Kontroll-Parameter	nicht vorhanden
'E0'	CREATE FILE (EF) siehe Anmerkung 3	- '00'	- '00': keine Info	File-Kontroll-Parameter	nicht vorhanden
'E4'	DELETE FILE	- '00': lösche aktuelles DF (das DF sollte vorher ausgewählt werden, und es sollte kein aktuelles EF geben; dies muss durch die externe SW sichergestellt werden) - '02': lösche EF im aktuellen DF	- '00': keine Info	- P1 = '00': nicht vorhanden - P1 = '02': EFID	nicht vorhanden
'04'	DEACTIVATE FILE siehe Anmerkung 4	- '00': deaktiviere aktuelles DF (das DF sollte vorher ausgewählt werden, und es sollte kein aktuelles EF geben; dies muss durch die externe SW sichergestellt werden) - '02': deaktiviere angegebenes EF im aktuellen DF	- '00': keine Info	- P1 = '00': nicht vorhanden - P1 = '02': File ID	nicht vorhanden
'44'	ACTIVATE FILE siehe Anmerkung 4 und Anhang C	- "00': aktiviere aktuelles DF (das DF sollte vorher ausgewählt werden, und es sollte kein aktuelles EF geben; dies muss durch die externe SW sichergestellt werden) - '02': aktiviere angegebenes EF im aktuellen DF	- '00': keine Info	- P1 = '00': nicht vorhanden - P1 = '02': File ID	nicht vorhanden

Der Datei-Selektions-Status nach Ausführung der Kommandos CREATE FILE, ACTIVATE FILE und DEACTIVATE FILE ist nicht festgelegt, d.h., es wird als Zustand „nicht selektiert“ angenommen.

Anmerkungen:

(1) *Dieses Kommando ist kein "muss", wird aber empfohlen (der Gebrauch ist in Teil 2 der eGK-Spezifikation beschrieben; das Kommando dient als Rahmen für Kommandos wie CREATE FILE, ACTIVATE FILE usw., es kann aber auch zum Laden von „application images" benutzt werden, d.h. File-Kontroll-Blöcke und Daten können in den Speicher geschrieben werden, ohne Kommandos wie CREATE FILE nutzen zu müssen). Für das Kommando LOAD APPLICATION mit seinen eingebetteten Einzel-Kommandos wird jeweils nur eine Zugriffsregel angewendet. Die Menge der zulässigen eingebetteten Kommandos ist beschränkt (z.B. ist DELETE FILE in dieser Kommandosequenz nicht erlaubt). Anlegen einer neuen Anwendung und ggf. Anlegen eines neuen Files in einer existierenden Anwendung; Modifikationen sind auf die neue Anwendung bzw. das neue File beschränkt.*

(2) *Der Parameter P1 = '00' sollte nur gesetzt werden, wenn ein EF erzeugt wird. Andererseits ist die eGK gemäß* [ISO7816-9] *nicht verpflichtet, das Kommando abzulehnen, wenn der Parameter P1 = '00' zusammen mit dem File-beschreibenden Byte (File Descriptor Byte) mit Wert '38' genutzt wird, das im Datenfeld des Kommandos übertragen wird.*

(3) *Die Unterstützung des Kommandos ist erforderlich, die genaue Struktur der File Control Parameter bleibt aber herstellerspezifisch.*

(4) *Dieses Kommando wird nicht nur für das Kartenmanagement benötigt.*

3.2 Behandlung von Kommandos nach Interrupt

Wenn durch ein Kommando Daten der Chipkarte im nicht-flüchtigen Speicher geändert werden sollen und das Kommando bei seiner Ausführung unterbrochen wird, so dass die Daten nur teilweise geschrieben wurden, muss durch interne Recovery-Mechanismen sichergestellt werden, dass die Daten, die im nicht-flüchtigen Speicher durch das Kommando verändert werden sollten, vor der Ausführung des nächsten Kommandos in den Zustand vor der Kommandoausführung (Roll-Back) oder in den Zustand nach der Kommandoausführung (Roll-Forward) versetzt werden.

Für die Kommandos

- - UPDATE BINARY
- - UPDATE RECORD
- - APPEND RECORD
- - CHANGE REFERENCE DATA
- - GENERATE ASYMMETRIC KEY PAIR
- - LOAD APPLICATION
- - CREATE FILE und
- - ACTIVATE FILE

muss bei der Unterbrechung der Schreibvorgänge eines Kommandos ein Roll-Back ausgeführt werden. Hiervon müssen nur solche Daten ausgenommen werden, die der Erkennung und Protokollierung von Sicherheitsverstößen (z.B. Bedien- oder Fehlbedienzähler) dienen. Für diese muss ein Roll-Forward erfolgen.

Für die Kommandos

- - DEACTIVATE RECORD
- - DEACTIVATE FILE und
- - DELETE FILE

muss ein Roll-Forward erfolgen.

Wenn die Unterbrechung eines Kommandos vor dem Beginn der Schreibprozesse erfolgt, darf kein Roll-Forward ausgeführt werden.

Wenn die Unterbrechung eines Kommandos erst während der Ausgabe von Antwortdaten erfolgt, darf kein Roll-Back ausgeführt werden. Ein unterbrochenes GENERATE ASYM-METRIC KEY PAIR kann dazu führen, dass der betreffende Schlüssel nicht verwendbar ist

4 File-Verwaltung

Die File-Verwaltung soll wie in [ISO7816-4] beschrieben aufgebaut sein und die folgenden Funktionalitäten umfassen:

- Schachtelungstiefe: MF-Ebene und DF-Ebene (tiefere DF-Ebenen sind optional)
- EFs mit transparenten Strukturen
- EFs mit linearer Struktur und Records festgelegter Länge
- EFs mit linearer Struktur und Records variabler Länge
- EFs mit zyklischer Struktur
- Records mit einer Länge von 1 Byte bis zu 255 Byte
- Anzahl von Records, die in einem File möglich sind: mindestens 254
- EFs mit Short EF Identifier (nutzbar nur im zugehörigen DF oder der zugehörigen Anwendung)
- DF mit Application Identifier (AID), siehe Tabelle 1.

Mindestens die folgenden File-Kontroll-Parameter sind funktional zu unterstützen (empfohlene FCP-Kodierung: siehe Tabelle 12 in [ISO7816-4]):

- Anzahl der Daten-Bytes (EF) ohne Strukturinformation: siehe DO mit Identifier (tag) '80'
- File-Descriptor, siehe DO mit Identifier (tag) '82'
- File-Identifier, siehe DO mit Identifier (tag) '83'
- AID, siehe DO mit Identifier (tag) '84' (dieses DO kann in FCP zweimal vorkommen, z.B. eine Gesundheitsanwendung kann durch einen nationalen und einen internationalen AID gekennzeichnet sein)
- Short EF Identifier, siehe DO mit Identifier (tag) '88'
- Lebenszyklus-Status-Byte, siehe DO mit Identifier (tag) '8A'
- Sicherheits-Attribut, bezugnehmend auf das erweiterte Format (siehe Tabelle 25 von [ISO7816-4]), siehe DO mit Identifier (tag) '8B')
- Ein Template für Sicherheitsattribute für Daten-Objekte, siehe DO mit Identifier (tag) 'A0'

5 Sicherheitsattribute und Zugriffsregeln

Da Smartcards persönliche Sicherheitsinstrumente sind, ist die Nutzung von Sicherheitsattributen ein unentbehrlicher Bestandteil jeder Karte. Speziell für die eGK ist eine differenzierte und granulare Zuweisung der Zugangsberechtigungen und der Sicherheitsbedingungen unbedingt notwendig.

Das Sicherheitsmanagement muss folg. Funktionalitäten unterstützen (Kodierung: siehe [ISO7816-4]):

- erweitertes Format
- Zugriffs-Modus-Byte für DFs
- Zugriffs-Modus-Byte für EFs
- Zugriffs-Modus-Byte für DOs
- Zugriffs-Modus-DOs mit Identifier (tag) '80' bis '8F'
- Datenobjekte mit Sicherheitsbedingungen, siehe Tabelle 23 von [ISO7816-4]; Unterstützung des Bytes für die Sicherheitsbedingungen (Identifier (tag) '9E'), NOT-Template (Identifier (tag) 'A7') und CRT mit Identifier (tag) 'B6', das asymmetrische SM anzeigt, ist optional
- EF für die betreffenden Zugriffsregeln (EF.ARR)

Karten, die EF.ARR nicht unterstützen, müssen einen Mechanismus mit äquivalenter Funktionalität bereitstellen.

Ein EF.ARR, das zur Root gehört, ist unterhalb der Root angesiedelt. Ein EF.ARR, das zu einer bestimmten Anwendung gehört, ist unterhalb des betreffenden DF angelegt.

Zugriffsregeln in einem EF.ARR können geändert oder ergänzt werden, falls die Sicherheitsbedingungen das zulassen. Es kann notwendig sein, Zugriffsregeln, die nicht änderbar sein sollen, in einem EF.ARR, und solche, die änderbar sein sollen, in einem anderen EF.ARR zu speichern. Daher müssen Sicherheitsumgebung, Record-Nummer und File-Identifier in den Datenobjekten für Sicherheitsattribute spezifiziert werden können, die sich auf das erweiterte Format beziehen (siehe [ISO7816-4], Tabelle 25).

Kommandos sollen nur ausgeführt werden, wenn die Sicherheitsbedingungen entsprechend der Zugriffsregel für diese Operation erfüllt sind oder das Kommando durch eine implizite Konvention immer erlaubt ist. Folgende Kommandos sind im Klartext immer erlaubt, d.h. falls kein TC etabliert wurde, sind diese Kommandos im Sinne der Zugriffsbedingung "Always" möglich:

- SELECT
- MSE
- GET CHALLENGE
- PSO: HASH
- INT. / EXT. AUTHENTICATE mit Public Keys, die über VERIFY CERTIFICATE importiert wurden

In einem Trusted Channel sind folgende Kommandos (im SM-Modus) immer erlaubt:

- SELECT (EF)
- MSE Operation SET
- PSO: HASH

Falls die Karte einen Dual-Interface-Chip enthält (kontaktbasiertes Interface nach [ISO7816-3] und RF-Interface), muss das COS in der Lage sein, die Einschränkung der Nutzbarkeit einer Anwendung auf ein bestimmtes Interface kontrollieren zu können (siehe [ISO7816-4]).

6 PIN Management

Die eGK muss zum Zwecke der Benutzerverifikation PIN Objekte speichern können. Wie die PIN Objekte gespeichert werden, ist herstellerspezifisch. An der Schnittstelle zur eGK sind folgende Attribute eines PIN Objektes sichtbar:

Referenzwert: Dieser repräsentiert das dem Benutzer bekannte Geheimnis. Als Referenzwert wird bei Gesundheitsanwendungen nur eine Ziffernfolge (Persönliche Identifikations-Nummer PIN) verwendet. Eine PIN wird zur Personalisierungszeit meistens als Transport-PIN eingetragen (die Konstruktion der Transport-PIN ist herstellerspezifisch). Eine PIN kann durch das Kommando CHANGE REFERENCE DATA verändert werden (siehe dort). Falls die Zugriffsregeln dies zulassen, kann ein existierender Referenzwert durch eine neue PIN mit dem Kommando RESET RETRY COUNTER ersetzt werden.

PIN-Referenz und PIN-Typ: Die PIN-Referenz besteht aus 1 Byte. Es wird zwischen globalen und DF-spezifischen (lokalen) PINs unterschieden, was Einfluss auf das Codierungsschema der PIN hat (siehe [ISO7816-4]). Eine PIN-Referenz wird bei der Personalisierung eingetragen. Mindestens 3 PINs müssen in einem DF koexistent unterstützt werden können.

Übertragungsformat: Dieses legt das Format fest, wie die dem Benutzer bekannte PIN im Rahmen der Kommandos VERIFY, CHANGE REFERENCE DATA und RESET RETRY COUNTER zur Karte transportiert werden. Dieses Attribut wird spätestens zur Personalisierungszeit herstellerspezifisch eingetragen. Es ist nicht gefordert, dass dieses Attribut veränderbar ist. Die eGK muss nur das Format 2PIN Block unterstützen.

“Format 2 PIN Block” entsprechend [ISO9564-1] für eine 5-stellige PIN (Beispiel):

C	L	P	P	P	P	P	F	F	F	F	F	F	F	F	F

C = Kontrollfeld, Wert 2

L = Länge der PIN in BCD

P = PIN-Ziffer in BCD

F = Füllwert mit dem Wert ‘F’

Das so kodierte Format besteht immer aus 8 Byte. Bei einer PIN-Änderung bzw. beim Setzen einer neuen PIN muss die eGK die Korrektheit des Formats prüfen.

Mindestlänge: Auf dieses Attribut wird im Rahmen der Kommandos CHANGE REFERENCE DATA und RESET RETRY COUNTER zugegriffen (siehe dort). Dieses Attribut wird spätestens zur Personalisierungszeit herstellerspezifisch eingetragen. Der zu unterstützende Wertebereich umfasst das Intervall [4, 12].

Maximallänge: Die Maximallänge beträgt 12 Ziffern. Soll die Maximallänge z.B. auf den Wert 8 begrenzt werden, dann ist dies außerhalb der Karte durch die Anwendung zu kontrollieren.

Ausgangswert des Wiederholungszählers: Auf dieses Attribut wird im Rahmen der Kommandos VERIFY, CHANGE REFERENCE DATA und RESET RETRY COUNTER zugegriffen (siehe dort). Dieses Attribut wird spätestens zur Personalisierungszeit herstellerspezifisch eingetragen. Der Wertebereich muss mindestens das Intervall [1, 15] umfassen. Üblicher Wert ist 3.

Wiederholungszähler: Auf dieses Attribut wird im Rahmen der Kommandos VERIFY, CHANGE REFERENCE DATA und RESET RETRY COUNTER zugegriffen (siehe dort). Dieses Attribut wird spätestens zur Personalisierungszeit herstellerspezifisch eingetragen. Der Wertebereich muss mindestens das Intervall [1, 15] umfassen. Das Management des Wiederholungszählers erfolgt durch eine Zustandsmaschine:

- ist der Wert des Wiederholungszählers Null, ist die PIN blockiert (siehe Rücksetzcode)
- ist der Wert des Wiederholungszählers nicht Null und die eingegebene PIN richtig, dann wird der Wiederholungszähler auf den Ausgangswert gesetzt und ein PIN-spezifischer Sicherheitszustand gesetzt ("Präsentation der PIN mit PIN-Referenz 'xx' erfolgreich")
- ist der Wert des Wiederholungszählers nicht Null und die eingegebene PIN falsch, dann wird der Wiederholungszähler dekrementiert (verbleibende Versuche sind mit dem Status-Code '63C*x'* anzuzeigen). Falls der Wiederholungszähler vor Kommandoausführung auf eins stand und die PIN falsch ist, dann SOLL als Statuswort ’63 C0’ verwendet werden. Im beschriebenen Fall KANN ’69 83’ verwendet werden.
- ein Syntax-Fehler beim Übertragungsformat (Status-Code '6A80') darf keinen Einfluss auf den Wiederholungszähler haben

Rücksetzcode: Auf dieses Attribut wird im Rahmen des Kommandos RESET RETRY COUNTER zugegriffen (siehe dort). Dieses Attribut wird zur Personalisierungszeit herstellerspezifisch eingetragen. Der Rücksetzcode wird im selben Format übertragen, wie die PIN. Der Rücksetzcode mit mindestens 8, maximal 12 Ziffern. Das Vorhandensein eines Rücksetzcodes ist abhängig von den gewünschten PIN-Attributen.

Nutzungsbegrenzung des Rücksetzcode: Auf dieses Attribut wird im Rahmen des Kommandos RESET RETRY COUNTER zugegriffen (siehe dort). Dieses Attribut wird spätestens zur Personalisierungszeit herstellerspezifisch eingetragen. Der Wertebereich muss mindestens das Intervall [1, 15] umfassen.

Bei richtiger PIN-Eingabe setzen das VERIFY- und das CHANGE REFERENCE DATA-Kommando einen PIN-spezifischen Sicherheitszustand: "Präsentation der PIN mit PIN-Referenz 'xx' erfolgreich".

Ein CHANGE REFERENCE DATA-Kommando mit einer Transport-PIN muss keinen Sicherheitsstatus setzen, d.h. ein Anwendungssystem sollte daher immer vor Zugriff auf PIN-geschützte Funktionen oder Daten ein VERIFY-Kommando senden.

Für die Nutzung des privaten Schlüssels für qualifizierte elektronische Signaturen muss es möglich sein, eine PIN-Eingabe vor jeder Berechnung einer Signatur zu fordern.

Die Unterstützung dedizierter Werte für die Zahl der Signaturen ist optional (z.B. eine neue PIN-Eingabe nach der Berechnung von 6 Signaturen). Die Einstellung gilt nur für die aktuelle Sitzung, d.h. nach einem Reset gilt immer wieder die gleiche Ausgangssituation (keine PIN eingegeben und keine Nutzung des privaten Signaturschlüssels).

Wenn das COS ein Transportschutzverfahren unterstützt, bei welchem nur eine PIN übergeben wird, dann MUSS das COS folgende Varianten von CHANGE REFERENCE DATA unterstützen:

CHANGE REFERENCE DATA mit P1=´01´, P2 = PIN Referenz, Datenfeld der Kommandonachricht newPIN, Le abwesend

Das COS MUSS in jedem Fall folgende Variante von CHANGE REFERENCE DATA unterstützen:

CHANGE REFERENCE DATA mit P1=´00´, P2 = PIN Referenz, Datenfeld der Kommandonachricht oldPIN || newPIN, Le abwesend

Die Informationen oldPIN und newPIN MÜSSEN stets im Format-2-PIN Block übertragen werden.

Das COS KANN weitere Varianten für CHANGE REFERENCE DATA unterstützen oder ablehnen.

Im Rahmen eines RESET RETRY COUNTER Kommandos

- SOLL der Nutzungszähler der PUK heruntergezählt werden, wenn die Länge der übergebenen PUK nicht mit der Länge der in der Karte gespeicherten PUK übereinstimmt. In diesem Fall MUSS die Karte mit ´63Cx´ antworten, wobei ´x´ entweder die Anzahl der noch möglichen Nutzungen oder der noch möglichen Fehlversuche angibt.

- KANN der Nutzungszähler KANN unverändert bleiben, wenn die Länge der übergebenen PUK nicht mit der Länge der in der Karte gespeicherten PUK übereinstimmt. In diesem Fall MUSS die Karte mit ´6A80´ antworten.

7 Sicherheitsumgebungen (Security Environments)

Die Sicherheitsumgebungen (SE) definieren Zugriffsrechte auf Datenobjekte und Schlüssel. In unterschiedlichen SE können unterschiedliche Zugriffsrechte für dasselbe Datenobjekt oder denselben Schlüssel definiert werden.

Das SE-Management muss die folgenden Funktionalitäten unterstützen:

- SE #1 als voreingestelltes SE
- mindestens 3 verschiedene SEs im selben DF (Unterstützung von MSE RESTORE)

Wenn auf MF-Ebene ein SE explizit oder implizit ausgewählt wurde, dann bleibt dieses auch dann aktiv, wenn ein anderes DF selektiert wird.

8 Secure Messaging

Die zu unterstützenden Datenobjekte und SM-Funktionen sind in Anhang D beschrieben.

9 Sicherheitsstatus und Situation nach Anwendungsselektion

Ein Sicherheitszustand in der eGK kann außer durch eine PIN-Präsentation (siehe Kapitel 8) auch durch andere Verfahren gesetzt werden

Die folgenden Verfahren, Sicherheitszustände zu setzen, müssen unterstützt werden:

- Status nach erfolgreicher PIN-Präsentation (Status für mindestens 1 globale und 2 DF-spezifische PINs)
- Status nach erfolgreicher Schlüssel-Präsentation (Status für mindestens 3 globale und 3 DF-spezifische Schlüssel), ggf. einschließlich der 7-Byte-langen CHA-Kennung)

Nach einer erfolgreichen DF-Auswahl (Anwendungsselektion wird immer ohne SM durchgeführt) ist folgende Situation gegeben:

- der globale Sicherheitsstatus bleibt erhalten
- ein vorheriger DF-spezifischer Sicherheitsstatus geht verloren (Ausnahme: das neu selektierte DF ist dasselbe wie das vorherige)
- globale Schlüssel-Referenzen und das implizit oder explizit gesetzte SE auf MF-Level bleiben erhalten
- DF-spezifische Schlüssel-Referenzen und das DF-spezifische SE gehen verloren, falls ein neues DF gewählt wird. (Ausnahme: das neu selektierte DF ist dasselbe wie das vorherige: dann gehen die DF-spezifische Schlüssel-Referenz und das DF-spezifische SE möglicherweise verloren)

10 Algorithmen und Schlüsselreferenzen

Die sicherheitskritischen kryptografischen Berechnungen zur Erzeugung und Prüfung von elektronischen Signaturen und zur Verschlüsselung werden aus Sicherheitsgründen in der eGK ausgeführt.

Dazu muss die eGK folgende Algorithmen unterstützen:

- Die öffentlichen Exponenten aller RSA Schlüssel, die innerhalb der eGK verwendet werden MÜSSEN im Intervall [65.537, 4.294.967.295] = [´1 0001´, ´FFFF FFFF´] liegen. Das COS KANN andere öffentliche Exponenten akzeptieren oder ablehnen.
- RSA mit mindestens 1024 bit Schlüssellänge für das CV-Zertifikats-Management
- RSA mit Schlüssellängen, die den aktuellen Anforderungen des Algorithmenkataloges entsprechen [ALGCAT]
- RSA Signaturformate (Padding erfolgt durch die Karte)
- Padding nach [PKCS#1] Kapitel 9.2 (für Signaturen bezogen auf Schlüssel mit X.509-Zertifikaten, Kommandos PSO: COMPUTE DS und INT. AUTHENTICATE)
- Padding nach [ISO9796-2] (mit Zufallszahl für Signaturen bezogen auf Schlüssel mit X.509-Zertifikaten und ohne Zufallszahl bei der Nutzung von CV-Zertifikaten, Kommandos PSO: COMPUTE DS und PSO: VERIFY CERTIFICATE)
- SHA-1 (Nutzung in Übereinstimmung mit [ALGCAT] gefordert; siehe auch Tabelle 6)
- 3DES (Triple DES), Nutzungs-Modi siehe Anhang D und Anhang E, Kapitel E.4 und E.5
- Asymmetrische Authentifizierungs-Verfahren mit und ohne Einrichtung eines Trusted Channel, siehe Anhang E.2 und E.3
- Symmetrische Authentifizierungs-Verfahren mit Einrichtung eines Trusted Channel, siehe Anhang E.4
- Einseitiges Challenge/Response-Verfahren mit 3DES für externe Authentisierung, Anhang E.5
- Asymmetrische Client/Server-Authentisierung mittels X.509-Zertifikaten, Anhang E.6.

Tabelle 9 – AlgIDs für Hash Funktionen und Signaturalgorithmen für PSO: COMPUTE DS

AlgID	Bedeutung	Unterstützung in eGK
'1x' (default)	SHA-1 (160 bit)	Algorithmus und Padding
'2x'	RIPEMD160	
'3x'	SHA-2 (224 bit)	
'4x'	SHA-2 (256 bit)	mindestens Padding
'5x'	SHA-2 (384 bit)	
'6x'	SHA-2 (512 bit)	
'x0'	Hash Algorithmus x (selektierbar mit MSE SET HT)	mindestens SHA-1
'x1'	RSA mit DSI entsprechend [ISO9796-2] RND mit Hash- Algorithmus x	DSI ist von COS zu unterstützen, siehe Anmerkung
'x2'	RSA mit DSI entsprechend PKCS #1 Hash-Algorithmus x	DSI ist von COS zu unterstützen, siehe Anmerkung

Anmerkung: In dem X.509-Zertifikatsdokument wird nur PKCS#1 verwendet.

Jeder Schlüssel hat mindestens eine Nutzung. Bestimmte Schlüssel können auch mehrere Zwecke haben, soweit dies aus Anwendungssicht wünschenswert und sicherheitstechnisch zulässig ist (z.B. Nutzung des Schlüssels mit unterschiedlichen Padding-Verfahren). Mindestens 2 Nutzungsarten sollen prinzipiell unterscheidbar sein.

Ein kryptografischer Schlüssel wird entweder über einen 1-Byte langen Key Identifier oder über einen Schlüsselnamen referenziert. Bei den Schlüsselnamen ist eine Länge von mindestens 12 Byte zu unterstützen. Bei PINs werden nur 1-Byte lange PIN-Referenzen verwendet.

11 Control Reference Templates

Die folgenden CRTs müssen unterstützt werden:

- CRT zur Authentifizierung, Identifier (tag) 'A4'
- CRT für die kryptografische Prüfsumme, Identifier (tag) 'B4'
- CRT für die Hash-Berechnung, Identifier (tag) 'AA'
- CRT für elektronische Signatur, Identifier (tag) 'B6'
- CRT für die Vertraulichkeit, Identifier (tag) 'B8'

Innerhalb eines CRT müssen mindestens die folgenden DO unterstützt werden:

- Referenz für den kryptografischen Mechanismus (AlgID), Identifier (tag) '80'
- Schlüssel-Referenzen, Identifier (tags) '83' und '84'
- Nutzungsbezeichner, Identifier (tag) '95'

12 Technische Charakteristika und Übertragungsverfahren

Eine eGK ist eine Chipkarte normaler Größe (ID1-Karte) und muss mindestens die Klasse AB (5V-3V) unterstützen. Die Abmessungen und die Anordnung der Kontakte müssen [ISO7816-2] entsprechen.

Das Übertragungsverfahren ist gemäß [ISO7816-3] zu implementieren und muss folgende Funktionalitäten unterstützen:

- Übertragungsprotokoll T = 1
- Wird ein Übertragungsblock zur Karte gesendet, muss das NAD-Byte den Wert '00' haben. Die Karte muss jedoch das NAD-Byte nicht prüfen.
- S-Block ABORT wird normalerweise nicht verwendet, kann aber von der Karte zum Abbruch einer zu langen Chain bei Nicht-Beachtung der I/O-Puffergrößen eingesetzt werden
- Protokoll-Parameter-Auswahl (PPS), Unterstützung des aushandelbaren Modus
- Frequenz-Umsetzungs-Faktor (FI) und Baud-Raten-Anpassungs-Faktor (DI) mit FI/DI- und Fi/Di-Werten gemäß Tabelle 10
- Größe der Informationsfelder: IFSC = 254 Byte, IFSD = 254 Byte
- ATR-Kodierung in Übereinstimmung mit [ISO7816-3], wobei die Anforderungen dieses Kapitels berücksichtigt werden müssen

Tabelle 10 – FI/DI-Werte, von denen einer im ATR angezeigt werden muss

Fi/Di	FI/DI (TA1)	Baudrate in kbps	MHz
372/12	18	115,2 / 161,3	3,5712 / 5
512/16	95	156,2	5
512/32	96	312,5	5

Um Abwärtskompatibilität zu erreichen, müssen die Karten auch die entsprechenden Werte in Tabelle 11 unterstützen, die in einem PPS-Verfahren mit einem Karten-Terminal genutzt werden, das noch keine Baudraten von 115 kbps und höher unterstützt.

Tabelle 11 – Zusätzliche FI/DI-Werte, die in einem PPS-Verfahren unterstützt werden müssen

TA1 in ATR	Fi/Di	FI/DI (TA1 in PPS)	Baudrate in kbps	MHz
TA1 = 18	372/2	12	19,2	3,5712 / 5
TA1 = 18	372/4	13	38,4	3,5712 / 5
TA1 = 95 oder 96	512/2	92	19,5	5
TA1 = 95 oder 96	512/4	93	39,1	5
TA1 = 95 oder 96	512/8	94	78,1	5

13 Verkettete Kommandos

Bei folgenden Kommandos ist das Verketten zu unterstützen (Bit b5 in CLA):

- LOAD APPLICATION

Falls Kommandos wie oben dargestellt verkettet und mittels SM abgesichert werden sollen, dann ist jede Kommando-APDU und jede Response-APDU der Kette gemäß den im Anhang D dargestellten Regeln separat abzusichern.

14 Längen-Behandlung (Lc und Le)

14.1 Short Length

Entsprechend [ISO7816-4] bedeutet:

- Lc ≠ '00': Daten im Kommandodatenfeld der angegebenen Länge
- Le ≠ '00': Daten der angegebenen Länge sollen im Antwortdatenfeld zurückgeliefert werden
- Le = '00':

  (1) Antwortdaten haben eine Länge ≤ 256 Byte
  Daten der Länge ≤ 256 Byte werden zurückgeliefert.

  (2) Daten in einer transparenten Datei haben eine Länge > 256 Byte.
  Daten der Länge 256 Byte werden zurückgeliefert

  (3) Antwortdaten haben aufgrund von SM eine Länge > 256 Byte.
  In diesem Fall kommt nur die Verwendung von “extended length” in Betracht.

14.2 Extended Length

Die Unterstützung von "extended Length" ist im ATR in den Card Capabilities anzuzeigen. Darüber hinaus muss in eGKs mit "extended Length" ein EF.ATR vorhanden sein, der mindestens folgendes Datenobjekt mit vier eingebetteten Datenobjekten (Tag '02’ = Integer value, Längenfeld 1 Byte mit Wert '02' bzw. '03', Wertfeld = Anzahl der Bytes der APDU) enthält, siehe Tabelle 12.

Tabelle 12 – Datenobjekt Input/Output-Puffer-Größen

Tag	Länge	Wert
'E0'	'xx'	'02'-L-'xx...xx' || '02'-L-'xx...xx' ||'02'-L-'xx...xx'||'02'-L-'xx...xx' = - DO max. Anzahl Byte Kommando-APDU falls kein SM || - DO max. Anzahl Byte Antwort-APDU falls kein SM || - DO max. Anzahl Byte Kommando-APDU falls SM || - DO max. Anzahl Byte Antwort-APDU falls SM

Anmerkung: Die Kodierung ist [ECC-2]*-kompatibel.*

Bezüglich der Behandlung der Längenfelder gelten die gleichen Konventionen, wobei jedoch statt 256 der max. Wert 65536 Byte mit den Einschränkungen gemäß den verfügbaren I/O-Puffer-Größen zu verwenden ist.

15 Personalisierung, Kartenmanagement und Nachladen

Die Komplettierung der Anwendung zur Erstellung qualifizierter elektronischer Signaturen ist auf der Basis der in Kapitel 3 beschriebenen Kommandos zu unterstützen (genauer Ablauf wird in Teil 2 der eGK-Spezifikation beschrieben).

Die Kommandofolgen für das Kartenmanagement, das Anwendungs-Management und der Personalisierungsprozess bleiben herstellerspezifisch.

Der Nachlade-Prozess muss einen Authentifizierungs-Prozess und die Einrichtung eines Trusted Channel unterstützen (siehe auch Secure Messaging).

Folgende Funktionalitäten sind nach Herausgabe der eGK unterstützen:

- Hinzufügen eines EF innerhalb eines vorhandenen DF
- Hinzufügen eines DF mit seiner Unterstruktur
- Schlüsselimport
- Schlüsselerzeugung, siehe Tabelle 5

Die COS-Plattform muss auch das Löschen von DFs und EFs unterstützen.

Bezogen auf das Speicherplatz-Management muss der Speicher für eine existierende Anwendung (DF-Bereich) dynamisch erweitert werden können, wenn ein EF erzeugt wird, solange freier physikalischer Speicher vorhanden ist. Dies bedeutet, dass während der Personalisierung die Speichergröße für die Gesundheitsanwendung nicht festgelegt wird.

Zusätzlich kann die Betriebssystem-Plattform auch die Zuordnung einer festen Speichergröße für eine Anwendung zulassen.

16 Evaluierung und Zulassung der eGK

16.1 Evaluierung der eGK

Eine Evaluierung der eGK auf der Basis der endgültigen Version des Protection Profiles der eGK ist notwendig.

16.2 Zulassung der eGK

Die Zulassung der eGK erfolgt gemäß den Prüfvorschriften der gematik.

Anhang A
(normativ)
Status Codes

A.1 Allgemeine Anforderungen

Die folgenden Tabellen beschreiben die Fehlerbedingungen, die durch das Betriebssystem der eGK erkannt, und die zugehörigen Status Codes, die zurückgemeldet werden müssen. Die Fehlermeldungen und ihre Bedeutung stimmen mit [ISO7816-4] überein.

Zusätzliche spezifische Fehlermeldungen und Kodierungen können vorhanden sein, sind aber nicht Gegenstand dieser Spezifikation.

Die Reihenfolge von Prüfungen in der Karte ist nicht festgelegt. Das bedeutet: Wenn ein Kommando mehrere Fehler enthält, dann MUSS das Statuswort irgendeinen dieser Fehler anzeigen.

A.2 Status Codes

Tabelle A 1 beschreibt allgemeine Fehlermeldungen, die unterstützt werden müssen. Tabellen A 2 bis A 26 beschreiben Kommando-spezifische Status Codes, die unterstützt werden müssen.

Tabelle A 1 – allgemeine Fehlermeldungen

Error Condition	Status Code
Ausführungsfehler	’64 00’
Speicherfehler beim Lesen oder Schreiben von Daten	‘65 81’
Lc ist nicht erlaubt für diese Kommando-Variante	’67 00’
Lc stimmt nicht mit der Länge der Kommandodaten überein	’67 00’
Lc oder Le sind vorhanden, obwohl sie fehlen sollten	’67 00’
Lc oder Le fehlen, obwohl sie vorhanden sein sollten	’67 00’
Logische Kanäle werden nicht unterstützt	’68 81’
Die Verkettung von Kommandos wird nicht unterstützt. Anstelle dieses Status Codes kann auch '6E00' gesendet werden.	’68 84’
Die Sicherheitsbedingungen sind nicht erfüllt	’69 82’
Das erwartete SM DO fehlt	‘69 87’
Nicht korrektes SM DO	’69 88’
Nicht korrekte Parameter P1 – P2	‘6A 86’
INS wird nicht unterstützt	’6D 00’

Error Condition	Status Code
Die Klasse wird nicht unterstützt	‘6E 00’
SM–Handling: Reference data not usable	’69 84’
SM–Handling: Condition of use not satisfied	’69 85’
SM–Handling: Ungültige Schlüssel	’69 82’ oder ’69 84’ oder ’6A 88’

Tabelle A 2 - Status Codes für SELECT

SELECT	Status Code
File (DF oder EF), das ausgewählt werden soll, nicht gefunden	‘6A 82’
Das ausgewählte File ist deaktiviert	SOLL: ‘62 83’ KANN:’90 00’

Tabelle A 3 – Status Codes für READ BINARY

READ BINARY	Status Code
EOF vor dem Lesen von Ne erreicht	’62 82’
File nicht transparent	‘69 81’
Kommando-Option ohne SFI und aktuell kein EF ausgewählt	’69 86’
File, auf das sich SFI bezieht, nicht gefunden	‘6A 82’
Offset >= File-Größe	‘6B 00’
File ist deaktiviert	SOLL: ’62 83’ KANN: ‘6A 82’

Tabelle A 4 – Status Codes für UPDATE BINARY

UPDATE BINARY	Status Code
Das File ist nicht transparent	‘69 81’
Kommando-Option ohne SFI und aktuell kein EF ausgewählt	’69 86’
File, auf das sich SFI bezieht, nicht gefunden	‘6A 82’
Offset + Lc > File-Größe	‘6A 87’
Offset >= File-Größe	‘6B 00’

Tabelle A 5 – Status Codes für READ RECORD

READ RECORD	Status Code

READ RECORD	Status Code
Ausgewählter Record deaktiviert	’62 83’
Das File ist transparent	‘69 81’
Kommando-Option ohne SFI und aktuell kein EF ausgewählt	’69 86’
Das File, auf das sich SFI bezieht, wurde nicht gefunden	‘6A 82’
Der Record wurde nicht gefunden	‘6A 83’

Tabelle A 6 – Status Codes für UPDATE RECORD

UPDATE RECORD	Status Code
Ausgewählter Record deaktiviert	’62 83’
Das File ist transparent	‘69 81’
Kommando-Option ohne SFI und aktuell kein EF ausgewählt	’69 86’
Das File, auf das sich SFI bezieht, wurde nicht gefunden	‘6A 82’
Der Record wurde nicht gefunden	‘6A 83’
Nicht genug Speicherplatz	‘6A 84’

Tabelle A 7 – Status Codes für APPEND RECORD

APPEND RECORD	Status Code
Das File ist transparent oder enthält bereits die maximale Anzahl von Records	‘69 81’
Kommando-Option ohne SFI und aktuell kein EF ausgewählt	’69 86’
Das File, auf das sich SFI bezieht, wurde nicht gefunden	‘6A 82’
nicht genug Speicherplatz	‘6A 84’

Tabelle A 8 – Status Codes für SEARCH RECORD

SEARCH RECORD	Status Code
Suchzeichenfolge nicht gefunden	’62 82’
Das File ist transparent	‘69 81’
Kommando-Option ohne SFI und aktuell kein EF ausgewählt	’69 86’
Das File, auf das sich SFI bezieht, wurde nicht gefunden	‘6A 82’
Der Record wurde nicht gefunden	‘6A 83’

Tabelle A 9- Status Codes for DEACTIVATE RECORD

DEACTIVATE RECORD	Status Code
Ausgewählter Record ist deaktiviert	‘6283’
Das File ist transparent	‘69 81’
Kommando-Option ohne SFI und aktuell kein EF ausgewählt	’69 86’
Das File, auf das sich SFI bezieht, wurde nicht gefunden	‘6A 82’
Der Record wurde nicht gefunden	‘6A 83’

Tabelle A 10 – Status Codes für INTERNAL AUTHENTICATE

INTERNAL AUTHENTICATE	Status Code
Die referenzierten Daten (besonders den Schlüssel) nicht gefunden, siehe Anmerkung	‘6A 88’

Anmerkung
Eine falsche Schlüsselreferenz wird ggf. schon vom Kommando MANAGE SE festgestellt.

Tabelle A 11 – Status Codes für EXTERNAL AUTHENTICATE

EXTERNAL AUTHENTICATE	Status Code
Die referenzierten Daten (besonders den Schlüssel) nicht gefunden, siehe Anmerkung unter Tabelle A.12	‘6A 88’

Tabelle A 12 - Status Codes für MUTUAL AUTHENTICATE

MUTUAL AUTHENTICATE	Status Code
Die referenzierten Daten (besonders den Schlüssel) nicht gefunden, siehe Anmerkung unter Tabelle A.12	‘6A 88’

Tabelle A 13 – Status Codes für VERIFY

VERIFY	Status Code
Authentifikations-Methode blockiert	’69 83’
Die PIN-Verifizierung ist fehlgeschlagen (PIN-Wert falsch), noch erlaubte Versuche in ’X’	’63 C*x*’
Format-Fehler im Format 2 PIN-Block in den Kommmando-Daten	‘6A 80’
Referenz-Daten (speziell die PIN) nicht gefunden	‘6A 88’

Tabelle A 14 – Status Codes für CHANGE REFERENCE DATA

CHANGE REFERENCE DATA	Status Code
Die PIN-Verifizierung ist fehlgeschlagen (PIN-Wert falsch), noch erlaubte Versuche in ’X’	’63 C*x*’
Authentifikations-Methode blockiert	’69 83’
Format-Fehler im Format 2 PIN-Block in den Kommmando-Daten	‘6A 80’
Referenz-Daten (speziell die PIN) nicht gefunden	‘6A 88’

Tabelle A 15 – Status Codes für RESET RETRY COUNTER

RESET RETRY COUNTER	Status Code
Die Verifizierung des Resetting Codes ist fehlgeschlagen (Wert falsch), noch erlaubte Versuche in 'x'	'63 C*x*'
Authentifikations-Methode blockiert	'69 83'
Format-Fehler im Format 2 PIN-Block in den Kommando-Daten	'6A 80'
Referenz-Daten (speziell die PIN) nicht gefunden	'6A 88'

Tabelle A 16 – Status Codes für MANAGE SE

MANAGE SE	Status Code
Die referenzierten Daten (besonders den Schlüssel) nicht gefunden, siehe Anmerkung	'6A 88'

Anmerkung
Eine gesetzte Schlüsselreferenz wird ggf. erst bei dem Kommando ausgewertet, das sie benutzt.

Tabelle A 17 – Status Codes für GENERATE ASYMMETRIC KEY PAIR

GENERATE ASYMMETRIC KEY PAIR	Status Code
Referenz-Daten (speziell der Schlüssel) nicht gefunden, siehe Anmerkung unter Tabelle A.12	'6A 88'

Tabelle A 18 - Status Codes für PSO: COMPUTE DS

PSO: COMPUTE DS	Status Code
Referenz-Daten (speziell der Schlüssel) nicht gefunden, siehe Anmerkung unter Tabelle A.12	'6A 88'

Tabelle A 19 – Status Codes für PSO: HASH

PSO: HASH	Status Code
Format-Fehler in den Kommando-Daten	‘6A 80’

Tabelle A 20 – Status Codes für VERIFY CERTIFICATE

PSO: VERIFY CERTIFICATE	Status Code
Format-Fehler in den Kommando-Daten	‘6A 80’
Referenz-Daten (speziell der Schlüssel) nicht gefunden, siehe Anmerkung unter Tabelle A.12	‘6A 88’
Längenfehler in mindestens einem TLV Objekt	‘6A 85’

Tabelle A 21 – Status Codes für PSO: DECIPHER

PSO: DECIPHER	Status Code
Im Fall von RSA: Eingabewert außerhalb des zugelassenen Bereiches	‘6A 80’
Referenz-Daten (speziell der Schlüssel) nicht gefunden, siehe Anmerkung unter Tabelle A.12	‘6A 88’

Tabelle A 22 - Status Codes for LOAD APPLICATION

LOAD APPLICATION	Status Code
Return Code des eingebetteten Kommandos	‘xxxx’

Tabelle A 23 – Status Codes für CREATE FILE

CREATE FILE	Status Code
Kommando unverträglich mit Dateistruktur	’69 81’
Nicht genug Speicherplatz	‘6A 84’
Das File existiert schon	‘6A 89’
DF-Name existiert schon	'6A 8A'

Anmerkung: '6981' kann anstelle von '6A89' und '6A8A' vorkommen.

Tabelle A 24 – Status Codes für DELETE FILE

DELETE FILE	Status Code
Das File, das gelöscht werden soll, wurde nicht gefunden	‘6A 82’

Tabelle A 25 – Status Codes für DEACTIVATE FILE

DEACTIVATE FILE	Status Code
Das File (EF) wurde nicht gefunden	‘6A 82’

Anmerkung: '6A82' kann bei einem DF nicht vorkommen, weil DF zuvor selektiert werden muss.

Tabelle A 26 – Status Codes für ACTIVATE FILE

ACTIVATE FILE	Status Code
Das File (EF) wurde nicht gefunden, siehe Anmerkung unter Tab. A.28	‘6A 82’

Anhang B
(normativ)
Von der Karte prüfbare Authentisierungs-Zertifikate (CV-Zertifikate)

B.1 Prinzipieller Aufbau

Der prinzipielle Aufbau eines karten-prüfbaren Zertifikates (CV-Zertifikat) wird in der folgenden Tabelle gezeigt. Die Reihenfolge der Datenelemente kann - wie in [ISO7816-8] beschrieben - durch eine Header-Liste definiert werden (ein Header besteht aus tag und length eines TLV-Objekts). Dies erfordert eine definierte Länge jedes Datenelementes.

Tabelle B 1 – Zertifikatsinhalt und Header-Liste

Zertifikatsinhalt	Zertifikats-Profil-Identifier (1B)	Kennung der CA/ CA-Schlüssels (8B)	Kennung des Zertifikats-Inhabers/-Schlüssels (12B)	Autorisierung des Zertifikats-Inhabers (7B)	OID.PuK (x B)	PuK (modulus tag '81', exponent tag '82') (x B)
Inhalt der Header-Liste	'5F29 01'	'42 08'	'5F20 0C'	'5F4C 07'	'06 0x'	'7F49 xx || 81 xx || 82 xx'

B.1.1 Certificate Profile Identifier

Der "Certificate Profile Identifier (CPI)" hat den Zweck, die genaue Struktur eines CV-Zertifikates anzuzeigen. Er kann als ein Identifier einer Karten-internen Header-Liste betrachtet werden, in der die Verkettung von Datenelementen einschließlich ihrer Länge beschrieben wird, so dass z.B. ein öffentlicher Schlüssel (PuK) in einem Zertifikat von der das Zertifikat verifizierenden Karte gefunden werden kann.

B.1.2 Kennung der CA bzw. des CA-Schlüssels

Zur Prüfung eines CV-Zertifikats wird der öffentliche Schlüssel der CA benötigt, die das CV-Zertifikat signiert hat. Für die Bereitstellung der Kennung (Referenz) dieses Schlüssels wird das Datenobjekt bzw. Datenelement „Certification Authority Reference (CAR)" benutzt. Die CAR besteht aus:

- dem CA-Namen (Ländercode entsprechend [ISO3166-1] (2 Bytes, DE = Deutschland), gefolgt von einem Acronym der CA (3 Bytes, ASCII-Zeichen)) und
- einer Erweiterung für die Schlüssel-Referenzierung (3 Bytes)

Tabelle B 2 – Struktur der Certification Authority Referenz (benutzt als CA-Schlüssel-Identifier)

CA Name (5 Byte)	Erweiterung für die Schlüssel-Referenzierung (3 Byte)

Die Erweiterung hat den folgenden Aufbau:

Tabelle B 3 – Struktur der Erweiterung für die Schlüssel-Referenzierung

Service-Indikator (1 BCD)	Feld für CA-spezifische Info (nicht festgelegt) (1 BCD)	Algorithmen-Referenz (2 BCD)	Datum (die letzten beiden Ziffern des Jahres der CA-Schlüsselerzeugung) (2 BCD)

Das Feld „Service-Indikator“ hat den Wert 1 = Authentifizierung der Geschäftseinheit gemäß der Schlüsselverwendung in X.509v3-Zertifikaten.

Das Feld für CA-spezifische Info kann mit einem Wert nach Belieben der entsprechenden CA belegt werden.

Das Feld „Algorithmen-Referenz“ kann von einer CA individuell belegt werden, um verschiedene Public-Key-Algorithmen zu unterscheiden.

Das Feld „Datum“ enthält die letzten beiden Ziffern des Jahres, in dem das Schlüsselpaar zur Signierung der Zertifikate generiert wurde. Falls mehr als ein Schlüsselpaar erzeugt wurde, können diese durch Nutzung des Feldes für CA-spezifische Info unterschieden werden.

B.1.3 Kennung des Zertifikats-Inhabers bzw. Zertifikat-Inhaber-Schlüssels

Bei der Durchführung des CV-basierten Authentisierungsverfahrens wird die Kennung des zertifizierten öffentlichen Schlüssels benötigt. Für die Bereitstellung der Kennung wird das Datenobjekt bzw. das Datenelement „Certificate Holder Reference (CHR)“ benutzt. Das „Subject“, also der Zertifikats-Inhaber, dem der Schlüssel zugeordnet ist, ist entweder eine CA (CV-Zertifikat ausgestellt von der Root-CA für eine CA) oder die betreffende eGK. Daher werden 2 Varianten für die Konstruktion der Schlüsselreferenz benötigt, wie in den Tabellen B. 4 und B.5 beschrieben:

Tabelle B 4 – Struktur der “Certificate Holder Reference”, wenn der Zertifikatsinhaber eine CA ist

Füllbytes (4 Byte)	CA Name (5 Byte)	Erweiterung für die Schlüsselreferenzierung (3 Byte)

Ein Füllbyte ist als ‘00’ kodiert.

Die „Erweiterung für die Schlüsselreferenzierung“ hat dieselbe Struktur wie in Tabelle B.3 gezeigt. Das Feld „Datum“ enthält die letzten beiden Ziffern des Jahres, in welchem der öffentliche Schlüssel, der im Zertifikat (z.B. PuK.CA.CS_AUT) zertifiziert wird, veröffentlicht wurde.

Tabelle B 5 - Struktur der „Certificate Holder Reference“, wenn der Zertifikatsinhaber die Karte selbst ist

'0000' (2 Byte)	ICCSN (10 Byte)

B.1.4 Certificate Holder Authorization

Die „Certificate Holder Authorization (CHA)“ hat den Zweck, die Zugriffsrechte des Karteninhabers anzuzeigen, d.h., die Zugriffsrechte des Karteninhabers in Bezug auf Daten, die in einer anderen Karte gespeichert sind. Die Bedeutung von CHA kann mit einem rollenbasierten Gruppenschlüssel bei Nutzung symmetrischer Verschlüsselungsverfahren verglichen werden.

Die CHA besteht aus

- einem Präfix, der die Instanz bezeichnet, die das Profil zuweist oder hält, und
- der Kennzeichnung des Profils des Zertifikatsinhabers.

Tabelle B 6 – Struktur der „Certificate Holder Authorization”

Präfix (6 Byte)	Profil-ID (1 Byte)

Das Präfix kann aus einer AID (6 MSB) oder einer weltweit eindeutigen Kennziffer der entsprechenden Instanz bestehen. Verschiedene Gruppen oder verschiedene Instanzen von Zertifikatsinhabern dürfen nicht dieselbe CHA haben.

Die folgenden Tabellen zeigen Beispiele für Profil-Kennziffern (die vollständigen Tabellen sind in [gemSpec_eGK_P2] zu finden).

Tabelle B 7 – Beispiel CHA Profil-ID Kodierung (Zertifikatsinhaber = CA)

CHA Profil- ID	Besitzer
‘00’	CA (keine Autorisierung, wird in höheren Ebenen von Zertifikaten oder in Cross-Zertifikaten genutzt)

Anmerkung: ‘00’ bedeutet: keine Zugriffsrechte auf Daten in den Zielkarten (HBA oder SMC oder eGK)

Tabelle B 8 – Beispiel CHA-Profil-ID Kodierung (Zertifikatsinhaber = HBA)

CHA Profil- ID	Besitzer
‘01’	eKiosk
‘02’	z.B. HPC Arzt oder HBC Zahnarzt oder HPC Psychotherapeut
‘03’	z.B. Apotheker.
….	…..

Tabelle B 9 – Beispiel CHA-Profil-ID Kodierung (Zertifikatsinhaber = eGK)

CHA Profil-ID	Besitzer
‘00’	eGK Karte

Anmerkung: ‘00’ bedeutet: keine Zugriffsrechte auf Daten in den Zielkarten (HBA oder SMC)

B.1.5 „Object Identifier“ für die Signatur-Algorithmen des Zertifikats-Inhabers

Der „Object Identifier“ muss in Übereinstimmung mit [DIN66291-1] ausgewählt werden.

B.1.6 Öffentlicher Schlüssel des Zertifikatsinhabers

B.1.6.1 Prinzipieller Aufbau

Der öffentliche Schlüssel in einem Zertifikat besteht aus einer Verkettung von Parametern. Diese Parameter, die durch ein kontext-spezifisches Tag gekennzeichnet werden, gehören zu dem DO PK (Tag '7F49’, konstruiert) und müssen als Octet-String kodiert werden.

In der das CV-Zertifikat verifizierenden Einheit (z.B. in einem HBA, einer SMC oder einer eGK) kann das Vorhandensein eines solchen Parameters und seine Länge in einer passenden Kopfzeile beschrieben werden, die durch das CPI angezeigt wird.

B.1.6.2 Öffentlicher Schlüssel RSA

- Tag ‘81’: Modulus
- Tag ‘82’: Public exponent

B.1.7 Kodierung der CV-Zertifikate

Die folgende Tabelle zeigt die Kodierung der CV-Zertifikate, die von der eGK zu unterstützen sind.

Tabelle B 10 – CPI-Werte und CV-Feld-Werte

CPI (1 Byte)	CAR (8 Byte)	CHR (12 Byte)	CHA (7 Byte)	OID	PK	Bemerkung
'03'	...	...	Präfix (6 Byte) || 'xx'	'2B2403040 20201' (7 Byte)	Modulus (128 Byte) || Expo-nent (4 Byte)	CVC für eine CA, ausgestellt von einer Root-CA (Signierter Schlüssel ist ein Zertifikats-schlüssel zur Verifizierung weiterer Schlüssel in einer Zertifikatskette; gehasht wird der Zertifikatsinhalt)
'04'	...	...	Präfix (6 Byte) || 'xx'	'2B2403050 203' (6 Byte)	Modulus (128 Byte) || Expo-nent (4 Byte)	CVC für eine eGK, HPC oder SMC, ausgestellt von einer CA (signierter Schlüssel ist für Authentisierungsverfahren mit und ohne TC Etablierung, siehe Anmerkung sowie E.2 und E.3)

Anmerkung:
Zur Differenzierung zwischen den beiden Verfahren soll der zugehörige PrK mit zwei Key Identifier und jeweils eindeutigem Verwendungszweck versehen werden (es ist erlaubt, den Schlüssel doppelt abzulegen, falls das COS nur eine KeyID pro Schlüssel verwalten kann).

Tabelle B 11 – Object Identifier

OID- Kodierung	OID - Nummer	OID- Name	OID Registration Authority
'2B240304020201'	{1 3 36 3 4 2 2 1}	sigS_ISO9796-2Withsha1 = signature scheme with RSA signature and DSI according to [ISO9796-2] and SHA-1	TeleTrusT
'2B2403050203'	{1 3 36 3 5 2 3}	authS_ISO9796-2Withrsa_mutual = authentication scheme with RSA signature and DSI ac-cording to [ISO9796-2] and SHA-1 for a mutual authenti-cation with or without estab-lishment of a Trusted Channel	TeleTrusT
TeleTrusT-OID Registration Authority: www.teletrust.de/anwend.asp?Id=30200&Sprache=D_&HomePG=0			

B.2 Struktur und Inhalt eines CV-Zertifikats-Files

Ein CV-Zertifikats-EF enthält ein zusammengesetztes Zertifikats-Datenobjekt mit demTag ’7F21’ (RSA-Zertifikat mit message recovery), siehe folgende Tabellen B.12 und B13.

Tabelle B 12 – Struktur und Inhalt eines EF, das ein CV-Zertifikat mit CPI = ‘03’ enthält

tag	L	Wert		
'7F21'	'81CE'	CV-Zertifikat (206 byte)		
		tag	L	Wert
		'5F37'	'8180'	SIG.CA (128 byte)
				Digital Signature Input für SIG.CA (‘6A’ …’BC’):
				'6A' = Padding entsprechend [ISO9796-2]
				'03' = CPI
				'xx..xx' = CAR (8 byte)
				'xx..xx' = CHR (12 byte)
				'xx..xx' = CHA (7 byte)
				'xx..xx' = OID (7 byte)
				'xx..xx' = PK part 1 (erster Teil des Modulus, 71 byte)
				'xx..xx' = Hash (20 byte, Hash Input: DEs CPI … PK, siehe Tabelle B.8)
				'BC' = Trailer
		'5F38'	'3D'	'xx..xx' = PK-Rest (Rest des Modulus, gefolgt vom Exponenten ‘00010001’, 61 byte)
		'42'	'08'	'xx..xx' = CAR (8 byte)

Tabelle B 13– Struktur und Inhalt eines EF, das ein CV-Zertifikat mit CPI = ‘04’ enthält

tag	L	Wert		
'7F21'	'81CD'	CV-Zertifikat (205 byte)		
		tag	L	Wert
		'5F37'	'8180'	SIG.CA (128 byte)
				Digital Signature Input für SIG.CA (‘6A’ …’BC’):
				'6A' = Padding entsprechend [ISO9796-2]
				'04' = CPI
				'xx..xx' = CAR (8 byte)
				'xx..xx' = CHR (12 byte)
				'xx..xx' = CHA (7 byte)
				'xx..xx' = OID (6 byte)
				'xx..xx' = PK part 1 (erster Teil des Modulus, 72 byte)
				'xx..xx' = Hash (20 byte, Hash Input: DEs CPI … PK, siehe Tabelle B.8)
				'BC' = Trailer
		'5F38'	'3C'	'xx..xx' = PK-Rest (Rest des Modulus, gefolgt vom Exponenten ‘00010001’, 60 byte)
		'42'	'08'	'xx..xx' = CAR (8 byte)

B.3 CVC-Handling

Für das CVC-Handling gelten ergänzend zu den Ausführungen in B.1 und B.2 folgende Konventionen:

Die Prüfung einer Zertifikatskette beginnt immer mit der Selektion des Root PuK, dessen KeyReference 8 Byte lang ist.

Die Zertifikatskette umfasst üblicherweise 2 CVCs. Bei Verwendung eines Cross-Zertifikats umfasst die Zertifikatskette 3 CVCs.

Die eGK muss sich bei CVC mit PuK-Nutzung "CertSign" den zertifizierten PuK und dessen Schlüsselreferenz merken.

Die eGK muss sich bei CVC mit PuK-Nutzung "Authentication" den zertifizierten PuK, dessen Schlüsselreferenz und die CHA merken.

Bei erfolgreicher Authentisierung der Gegenseite (z.B. HPC) wird ein Sicherheitsstatus gesetzt, der die betreffende CHA als erfolgreich präsentiert kennzeichnet.

Anhang C
(normativ)
Kommando DEACTIVATE RECORD

Das Kommando DEACTIVATE RECORD (CLA = ’00’) versetzt einen Record eines linear fixen oder linear variablen EF in einen deaktivierten Status; der betroffene Record wird durch P1 beschrieben. Auf deaktivierte Records darf nicht mit den Kommandos READ RECORD, WRITE RECORD, UPDATE RECORD und ERASE RECORD zugegriffen werden können (Hinweis: WRITE RECORD und ERASE RECORD werden in dieser Spezifikation nicht verwendet, können aber optional Bestandteil der COS-Funktionalität sein). Falls ein derartiges Kommando benutzt wird, soll dieses Kommando mit dem Rückantwort-Code ’6283’ (ausgewählter Record ist deaktiviert) beendet werden. Deaktivierte Records sollen übersprungen werden, wenn ein SEARCH RECORD-Kommando ausgeführt wird. Falls der adressierte Record schon deaktiviert ist oder in einem Bereich schon deaktivierte Records vorhanden sind, sollen trotzdem die Status-Bytes ‘9000’ zurückgegeben werden.

Zur Aktivierung aller deaktivierten Records muss ein ACTIVATE FILE-Kommando genutzt werden. Falls das EF sich schon vorher im Status “deaktiviert” befand, werden das EF und auch alle Records aktiviert.

Tabelle C 1 – Kommando zum Deaktivieren eines Records

Kommando zum Deaktivieren eines Records					
INS	Name	P1	P2	C-Data Feld	R-Data Feld
'06'	DEACTIVATE RECORD*	Record-Nr.	- b8-b4: Short EFID oder 0000 (= aktuelles File) - b3-b1= 100: deaktiviere Record P1	nicht genutzt	nicht genutzt

** keine Patente bekannt, INS-Code vorgeschlagen durch DIN NI-17.4, Kommando bei ISO zur Standardisierung beantragt*

Zur Nutzung des DEACTIVATE RECORD-Kommandos muss die Karte intern den Status jedes Records speichern (record life cycle state RLCS). Der RCLS muss mindestens die folgenden Zustände unterscheiden:

- aktiviert, d.h. auf den Record kann gemäß den gültigen Zugriffsberechtigungen zugegriffen werden (normaler Gebrauch)
- Deaktiviert, d.h., auf den Record kann nicht zugegriffen werden (versteckt).

Es wird empfohlen, den Gebrauch des RLCS auf die Record-basierten Files zu beschränken, bei denen dieses Kommando benötigt wird.

Es sollte möglich sein, die An- oder Abwesenheit des RLCS innerhalb des FCP des zugehörigen EF durch ein Objekt namens „profile indicator“ anzuzeigen.

Tabelle C 2 – DO Profil-Indikator (vorgeschlagen durch DIN NI-17.4 als Zusatz zu ISO/IEC 7816-4, Tabelle 12 – File control parameter data objects)

Tag	Länge	Wert	angewendet auf
'8F'	1	Profil-Indikator, siehe Tab. C.3	EF mit linear or fixed Records, einmal

Wenn vorhanden, soll der Profil-Indikator in Tag '8F' in den „File Control Parameter" des betreffenden EF gemäß Tabelle C.3 interpretiert werden. Falls der Profil-Indikator im FCP fehlt, soll die Karte den Default-Wert '00' (EF hat keine Records mit Record-Lebenszyklus) nutzen, soweit dies nicht anders spezifiziert ist.

Tabelle C 3 – Kodierung des Profil-Indikators

b8	b7	b6	b5	b4	b3	b2	b1	Bedeutung
0	x	x	x	x	x	x	x	Nutzung für ISO/IEC
							0	Das File hat keine Records mit Record-Lebenszyklus
							1	Das File hat Records mit Record-Lebenszyklus
1	x	x	x	x	x	x	x	Private Nutzung
Jeder andere Wert ist für eine zukünftige Nutzung durch ISO/IEC JTC 1/SC 17 reserviert								

Anhang D
(normativ)
Secure Messaging

D.1 Secure Messaging mit Trusted Channel

Um Daten und Kommandos zwischen einer externen Komponente und der eGK sicher austauschen zu können, müssen diese Daten und Kommandos verschlüsselt ausgetauscht werden. Dazu müssen sich die Komponenten zunächst authentifizieren und können danach Sitzungsschlüssel aushandeln. Mit diesen Schlüsseln wird die Kommunikation verschlüsselt. Dieses Verfahren wird Secure Messaging (SM) mit Trusted Channel genannt.

Das SM-Management muss die nachfolgend beschriebenen Funktionalitäten unterstützen.

D.1.1 SM-DOs

Tabelle D.1 zeigt die DOs, die von der eGK zu unterstützen sind (Teilmenge der SM-DOs aus [ISO7816-4]).

Tabelle D 1 - SM Daten-Objekte

tag	Bedeutung
'81'	Klartextdaten (Plain Value PV), wenn INS gerade ist (zu schützen durch CC)
'B3'	Klartextdaten (Plain Value PV), wenn INS ungerade ist (zu schützen durch CC)
'97'	Le (zu schützen durch CC)
'99'	Status-Information (zu schützen durch CC)
'8E'	Kryptografische Prüfsumme
'87'	PI || Kryptogramm, wenn INS gerade ist (zu schützen durch CC)
'85'	Kryptogramm, wenn INS ungerade ist (zu schützen durch CC)

Für Kryptogramme ist der Padding-Indikator immer auf '01' gesetzt, d.h. Padding gemäß [ISO7816-4] ('80'…'00').

D.1.2 Kommandos und Antworten mit SM

Nachdem die gegenseitige Authentifizierung erfolgreich abgeschlossen und ein Trusted Channel eingerichtet ist, müssen alle Kommandos und Antworten im SM-Modus übertragen werden. Wenn Sitzungsschlüssel für einen bestimmten logischen Kanal vereinbart sind und die Karte ein Kommando ohne SM für diesen Kanal erhält, werden die Sitzungsschlüssel für diesen logischen Kanal ungültig. Der Sicherheitszustand, der sich auf das Authentifizierungsverfahren mit der Schlüsselvereinbarung bezieht, darf nicht länger verwendbar sein. Das Kommando ohne SM soll ausgeführt werden, sofern die Zugriffsregeln dies gestatten.

Da der Kommando-Header in die CC-Berechnung einbezogen werden soll, müssen die Bits b4 und b3 des CLA-Bytes auf 1 gesetzt werden. Zur Vereinfachung werden alle Beispiele für den Kanal #0 ausgelegt. Damit ergibt sich die folgende Struktur für die Kommandos und Antworten:

Kommando :

'0C'	INS	P1-P2	Lc	TPV	LPV	PV	TLe	LLe	Le	TCC	Lcc	CC	Le'

Die DOs PV und Le sind konditional, d.h., diese DOs sind nur vorhanden, wenn das Kommando ohne SM ein Datenfeld beziehungsweise ein Le-Feld enthält. Der Teil der kryptografischen Prüfsumme (CC), der übertragen werden muss, ist 8 Bytes lang.

Antwort mit Daten:

TPV	LPV	PV	TCC	Lcc	CC	SW1-SW2

Antwort ohne Daten:

TSW	'02'	SW1-SW2	TCC	Lcc	CC	SW1-SW2

Das DO PV ist konditional, d.h. dieses DO ist genau dann vorhanden, wenn die ungesicherte Antwort Antwortdaten enthält. Das DO SW muss vorhanden sein, wenn die ungesicherte Antwortdaten keine Antwortdaten enthält oder das Kommando mit Fehler beendet wurde. Andernfalls darf es fehlen.

Wenn Kommandos mit Kommando- oder Antwortdaten mit SM übertragen werden, müssen die Daten im Datenfeld als Kryptogramm übertragen werden, falls die dazugehörenden Sicherheitsbedingungen dies erfordern. Falls die Sicherheitsbedingungen dies nicht explizit erfordern, KÖNNEN die Daten sowohl als Kryptogramm als auch als Plain DO mit CC gesichert übertragen werden. Beispiele für diese Kommandos sind:

- READ BINARY
- UPDATE BINARY
- VERIFY
- CHANGE RD
- RESET RC.

Damit ergibt sich die folgende Struktur für diese Kommandos und die dazugehörenden Antworten:

Kommando ohne Kryptogramm (z.B. READ BINARY):

'0C'	INS	P1-P2	Lc	TLe	LLe	Le	TCC	Lcc	CC	Le'

Antwort mit Kryptogramm:

TCG	LCG	PI, CG	TCC	Lcc	CC	SW1-SW2

Kommando mit Kryptogramm (z.B. VERIFY):

'0C'	INS	P1-P2	Lc	TCG	LCG	PI,CG	TCC	Lcc	CC	Le'

Antwort ohne Kryptogramm:

TSW	'02'	SW1-SW2	TCC	Lcc	CC	SW1-SW2

Anmerkung: Der Status-Code der Kommandoantwort muss identisch sein mit dem Status-Code, der durch CC geschützt wird.

D.1.3 Behandlung von SM-Fehlern

Wenn die eGK einen SM-Fehler während der Ausführung eines Kommandos entdeckt, müssen die Status-Bytes ohne SM zurückgegeben werden. In [ISO7816-4] werden die folgenden Status-Bytes definiert, um SM-Fehler anzuzeigen:

- 6987: erwartete SM-Datenobjekte fehlen
- 6988: SM-Datenobjekte sind fehlerhaft

Nachdem ein SM-Fehler festgestellt wurde, werden die SM-Schlüssel gelöscht. Der Sicherheitszustand, der sich auf das Authentifizierungsverfahren mit der Schlüsselvereinbarung bezieht, darf nicht länger verwendbar sein.

Handelt es sich nicht um einen SM-Fehler, dann MUSS dieser Fehlercode mit SM-DO (Tag '99') zurückzugeben werden. Der Trusted Channel MUSS dabei erhalten bleiben.

D.1.4 Padding bei der Berechnung von Prüfsummen

Es wird das Padding-Verfahren gemäß [ISO7816-4] (’80…00’) angewendet.

D.1.5 DES-Modus, Ausgangswert und Sendefolgezähler (Sequenzzähler)

D.1.5.1 Kryptogramme

Kryptogramme werden mit einem 16 Byte langen DES-3-Schlüssel im CBC-Modus mit dem Null-Vektor als Anfangswert gebildet. Das Padding des Eingangswertes soll gemäß [ISO7816-4] durchgeführt werden, siehe Kapitel 6.2.3.1.

D.1.5.2 Kryptografische Prüfsummen

Kryptografische Prüfsummen werden gemäß [ISO7816-4] wie folgt gebildet (die grundlegende Vorschrift ist, einen Retail-MAC gemäß ANSI X9.19 mit DES zu bilden):

- Erste Stufe: der Ausgangsprüfblock y_0 ist E(Ka, SSC).
- Folgende Stufen: die Prüfblöcke y_1, .. , y_n werden unter Nutzung von Ka berechnet.
- Letzte Stufe: die kryptografische Prüfsumme wird folgendermaßen vom letzten Prüfblock y_n berechnet: E(Ka, D(Kb, y_n)). Hierbei bedeutet E() Verschlüsselung mit DES, beziehungsweise D() Entschlüsselung mit DES.

Der Sendefolgezähler SSC muss jedes Mal um (+1) erhöht werden, bevor ein MAC berechnet wird; d.h., wenn der Anfangswert x ist, ist der Wert des SSC im nächsten Kommando x+1. Der SSC-Wert der ersten Antwort ist dann x+2.

Der Anfangswert für den SSC istSSC = RND.eGK (4 niedrigwertigste Bytes) || RND.SMC (4 niedrigwertigste Bytes).

D.2 Gebrauch von DES

Die folgende Abb. zeigt die Anwendung der Schlüssel beim DES-3-Verfahren (siehe [ISO11568-2]).

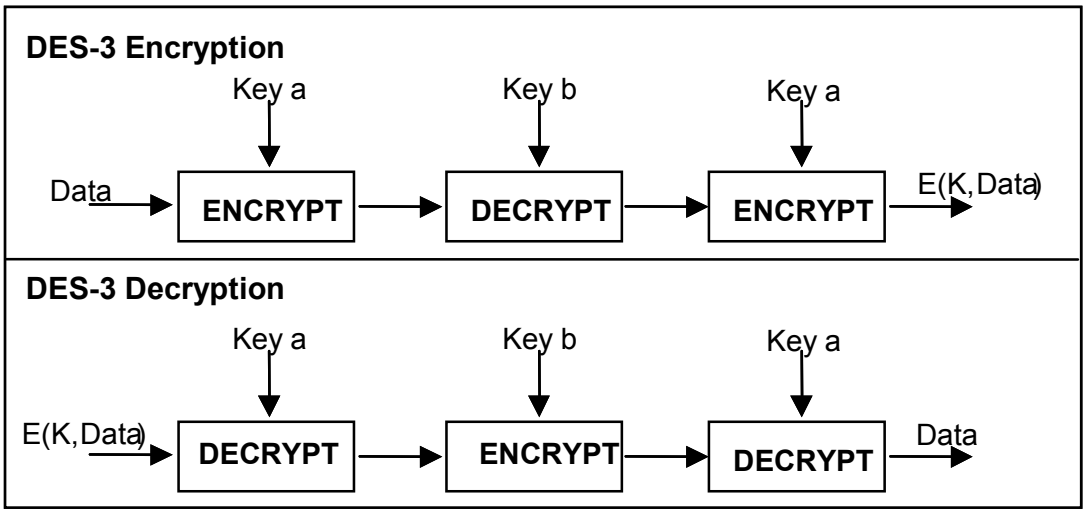

Abbildung D. 1 – DES-3-Verschlüsselung/Entschlüsselung

Der Retail-MAC wird wie in Abbildung D.2 dargestellt berechnet.

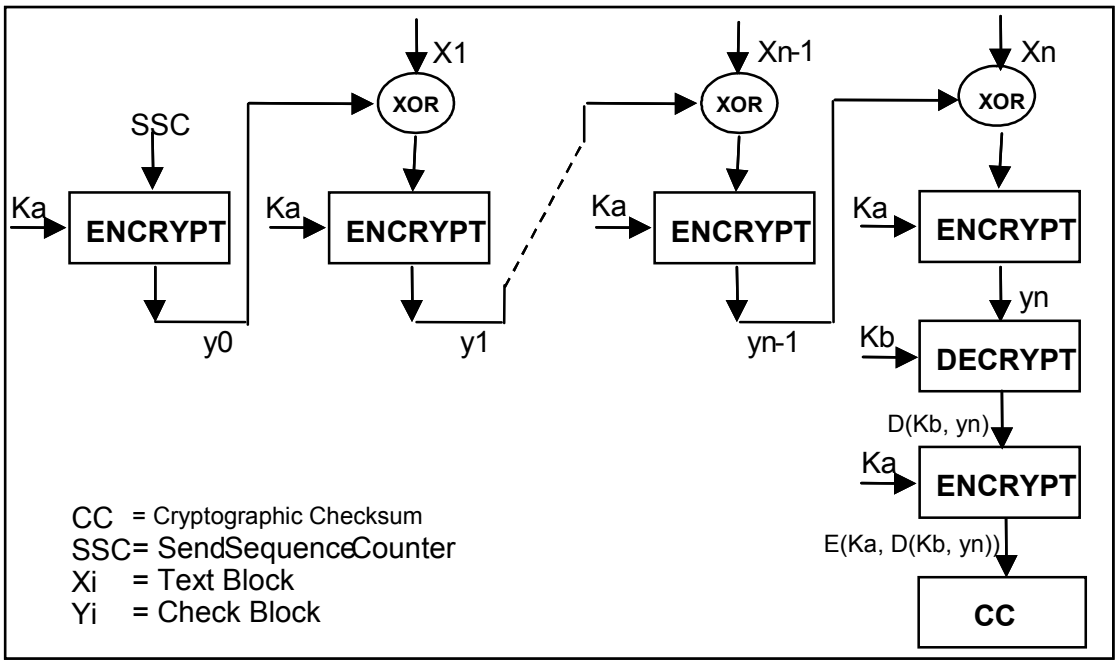

Abbildung D. 2 - Berechnung des Retail-MAC

D.3 SM-Schlüsselreferenzierung

Nach einer SM-Schlüsselvereinbarung sind die SM-Schlüssel implizit selektiert, d.h. es ist kein MSE-Kommando zur SM-Schlüsselselektion erforderlich.

Anhang E
(normativ)
Authentisierungsverfahren

E1. Notation für die folgenden Tabellen:

Tabelle E. 1 Notationen für die Beschreibung der Authentisierungsverfahren

Kommandodaten	< >
Antwortdaten	< >
Konkatenation	||
h(x)	Hash-Berechnung mit SHA1
ENC [key, data]	Kryptogramm
MAC [key, data]	Message Authentication Code
KD.xxx	Key Derivation Data, frei durch xxx wählbar
PRND	Pseudo Random Number
RND	Zufallszahl (8 byte)
SIG	Signatur
DS [key, DSI]	Signatur-Berechnung mit Verfahren entsprechend OID
DS-1	Signatur-Prüfung mit Verfahren entsprechend OID
ICCSN8.xxx	8 least significant Byte der ICCSN der Karte xxx

E.2 Asymmetrisches Authentisierungsverfahren ohne SM-Schlüsselvereinbarung

Abb. E.1 zeigt das asymmetrische Authentisierungsverfahren ohne SM-Schlüsselvereinbarung zwischen eGK und HPC. Das Verfahren wird auch zwischen eGK und SMC benutzt.

Vor dem Authentisierungsverfahren ist in der eGK der Schlüssel PrK.eGK.AUT und der Algorithmus mit der Referenz '1E' zu selektieren. Als Hash-Algorithmus h wird SHA-1, als SIG-Algorithmus RSA mit Schlüssellänge von 1024 Bit verwandt. Außerdem ist mittels CVC der PuK.HPC.AUT bzw. der PuK.SMC.AUT zu importieren. Das Verfahren ist aus zwei voneinander unabhängigen Authentisierungssequenzen zusammengesetzt, d.h. im ersten Schritt

authentisiert sich die eGK gegenüber HPC bzw. SMC und im zweiten Schritt authentisiert sich die HPC bzw. SMC gegenüber der eGK.

Die Zufallszahlen RND.HPC bzw. RND.eGK sind unmittelbar vor der Ausführung des Kommandos EXTERNAL AUTHENTICATE von der entsprechenden Karte anzufordern, d.h. zwischen dem Kommando GET CHALLENGE und dem Kommando EXTERNAL AUTHENTICATE darf kein anderes Kommando ausgeführt werden.

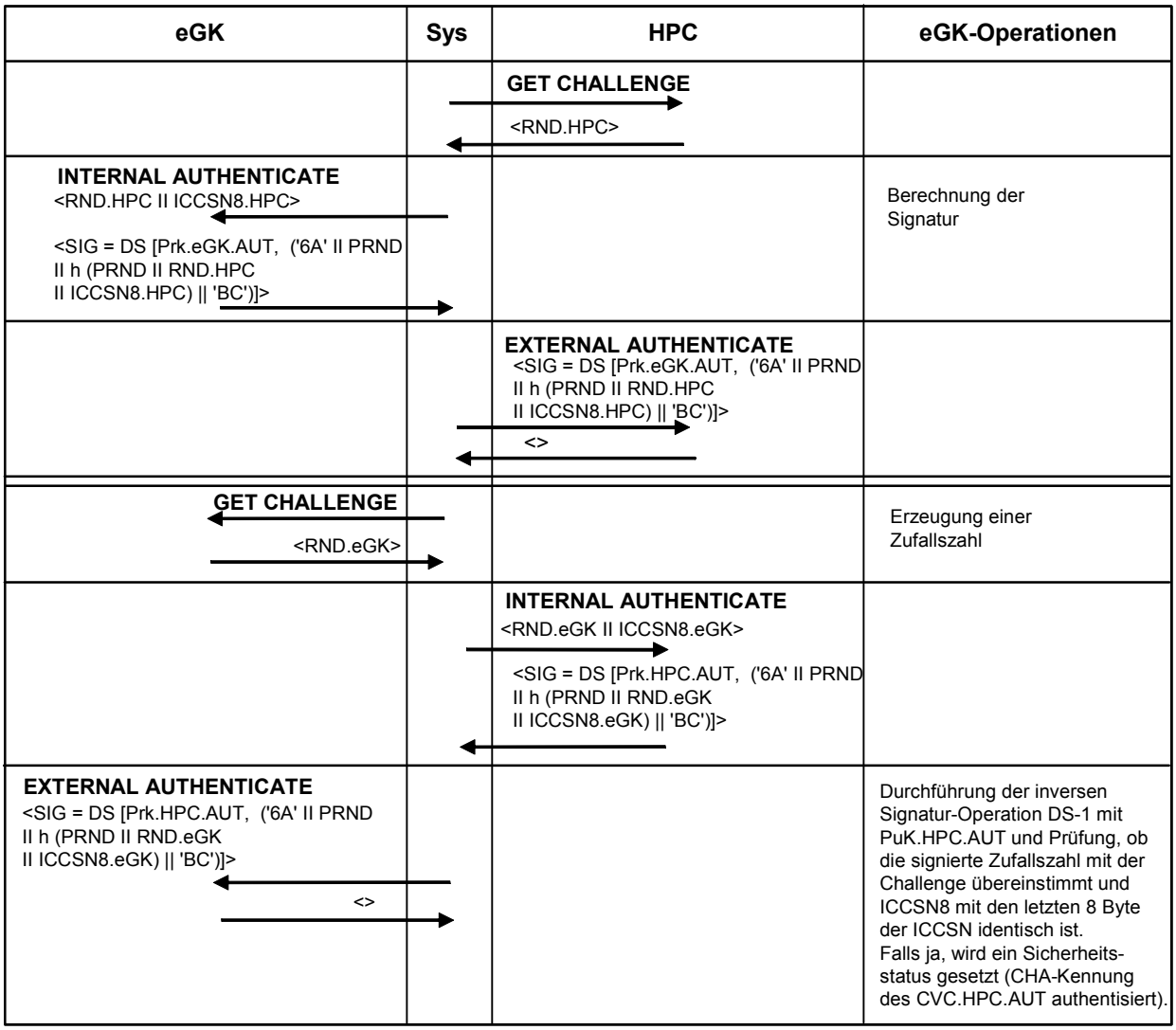

Abbildung E. 1 - Asymmetrisches Authentisierungsverfahren ohne SM-Schlüsselvereinbarung

Anmerkung: Der Octet String '6A … BC' hat stets dieselbe Länge in Bytes wie der Modulus des RSA-Schlüssels.

E.3 Asymmetrisches Authentisierungsverfahren mit SM-Schlüsselvereinbarung

Abb. E.2 zeigt das asymmetrische Authentisierungsverfahren mit SM-Schlüsselvereinbarung zwischen eGK und SMC. Zuvor ist in der eGK der Schlüssel PrK.eGK.AUT und der Algorithmus mit der Referenz '1F' zu selektieren. Als Hash-Algorithmus h wird SHA-1, als Signier-Algorithmus RSA mit Schlüssellänge von 1024 Bit verwandt. Außerdem ist mittels CVC der PuK.SMC.AUT zu importieren.

Die in Abb. E.2 dargestellte Kommandosequenz darf durch kein anderes Kommando unterbrochen werden, da sonst das Authentisierungsverfahren nicht erfolgreich beendet werden kann.

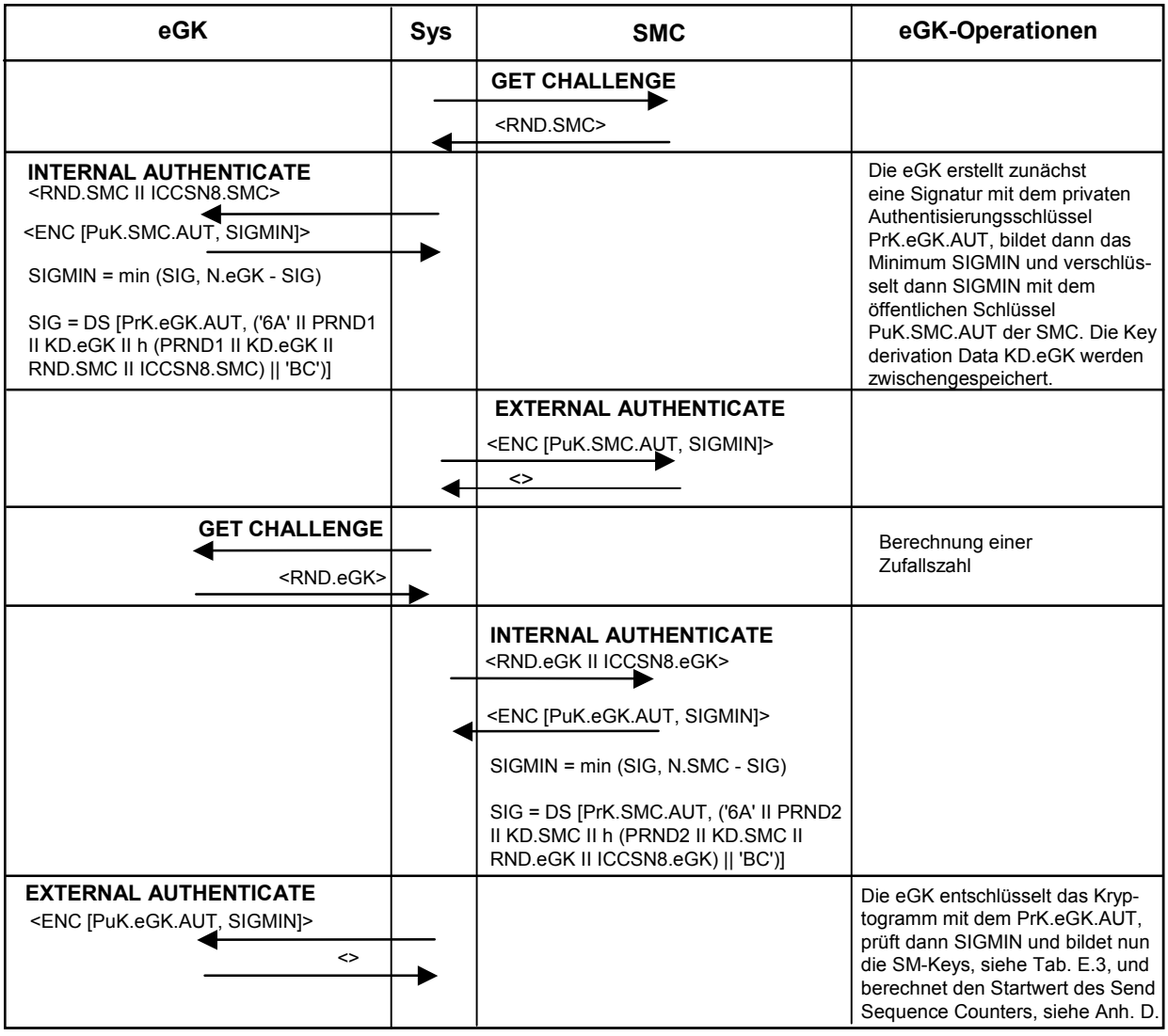

Abbildung E. 2 - Asymmetrisches Authentisierungsverfahren mit SM-Schlüsselvereinbarung

Anmerkung: Der Octet String '6A … BC' hat stets dieselbe Länge in Bytes wie der Modulus des RSA-Schlüssels.

Die SM-Schlüssel werden berechnet, wie in Abb. E.3 dargestellt.

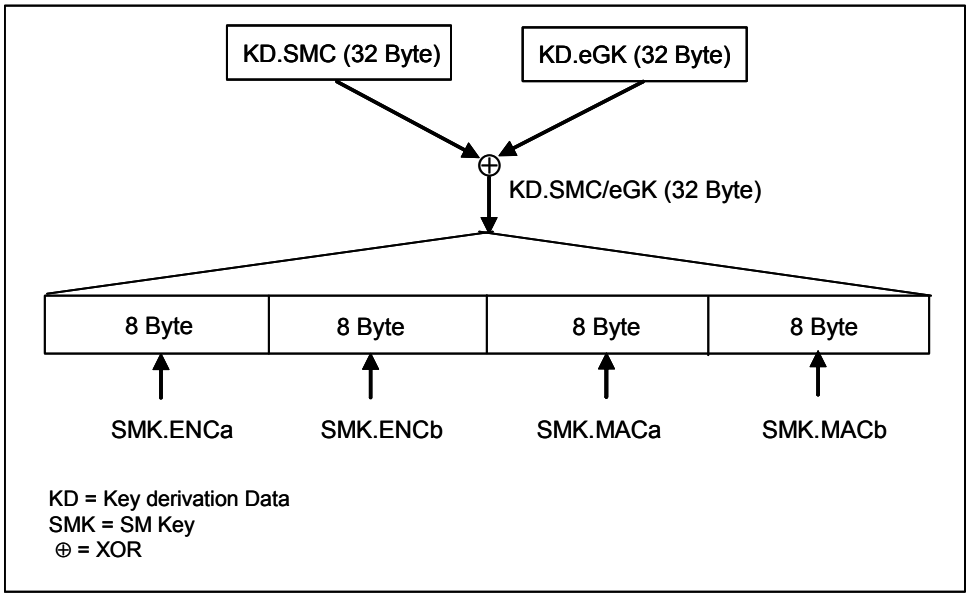

Abbildung E. 3 – SM-Key-Vereinbarung nach [DIN66291-4] und [prEN14890-1] bei asymmetrischen Verfahren

Die Kommando APDU, welche nach dem EXTERNAL AUTHENTICATE Kommando als nächste zur eGK geschickt wird, MUSS bereits die ausgehandelten Session Key nutzen, andernfalls DÜRFEN die Session Keys NICHT mehr verwendbar sein und der Sicherheitszustand aus dem Authentisierungsverfahren DARF NICHT mehr verwendbar sein.

E.4 Symmetrisches Authentisierungsverfahren mit SM-Schlüsselvereinbarung

Abb. E.4 zeigt das symmetrisches Authentisierungsverfahren mit SM-Schlüsselvereinbarung zwischen eGK und einem Server (z.B. VSDD oder CAMS). Das Verfahren ist identisch mit dem in Kapitel 7 von [prEN14890-1] beschrienen Verfahren. Der Server besitzt einen MasterKey MK, von dem alle individuellen SK.eGK mittels ICCSN der eGK abgeleitet werden (siehe Anmerkung).

Vor der Authentisierungsprozedur ist in der eGK das "Schlüsselpaar" SK.x. zu selektieren (SK.x verweist auf SK.x.ENC und SK.x.MAC). Die Angabe einer Algorithmus-Referenz ist nicht erforderlich (Algorithmus-Referenz implizit). Als Algorithmus wird 3DES verwendet.

Anmerkung:
Statt eines einzigen Schlüssels (MK) können auf dem Server auch n GruppenKeys GK (ein GK deckt einen bestimmten ICCSN-Bereich der eGKs ab) oder m IndividualKeys IK vorhanden sein (m Zahl der ausgegebenen eGKs, d.h. es erfolgt auf dem Server keine Schlüsselableitung, sondern nur die Schlüsselselektion mit der ICCSN der eGK).

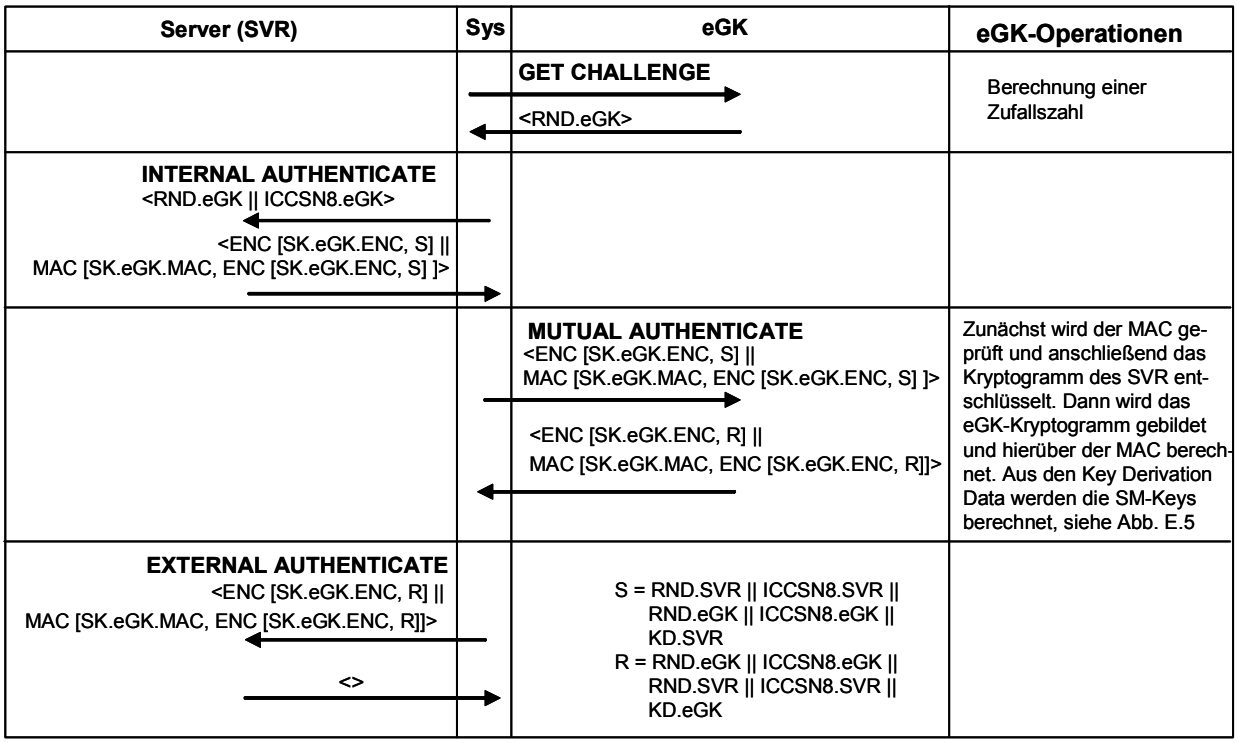

Abbildung E. 4 - Symmetrisches Authentisierungsverfahren mit SM-Schlüsselvereinbarung

Bei der Berechnung von ENC und MAC sind folgende Regeln zu beachten:

- Für die Berechnung des ENC-Teils im Kommando/Antwort-Feld hat IV für CBC-Verschlüsselung den Wert Null. Es wird kein Padding verwendet.
- Für die Berechnung des MAC-Teils im Kommando/Antwort-Feld wird der Ausgangs-Prüf-Block. Yo gleich Null gesetzt. Es wird Padding nach [ISO7816-4] '80…' verwendet.

Die SM-Schlüssel werden berechnet, wie in Abb. E.5 dargestellt.

Die Kommando APDU, welche nach dem MUTUAL AUTHENTICATE Kommando als nächste zur eGK geschickt wird, MUSS bereits die ausgehandelten Session Key nutzen, andernfalls DÜRFEN die Session Keys NICHT mehr verwendbar sein und der Sicherheitszustand aus dem Authentisierungsverfahren DARF NICHT mehr verwendbar sein.

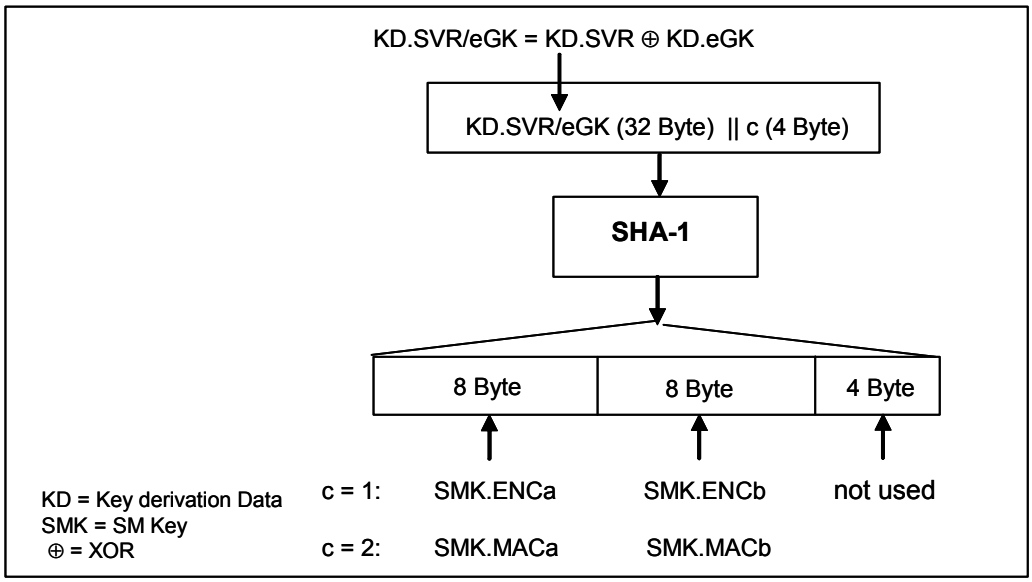

Abbildung E. 5– SM-Key-Ableitung nach ANSI X9.63 bei symmetrischen Verfahren

E.5 Challenge/Response-Verfahren mit symmetrischem Schlüssel

Bei diesem Verfahren wird eine Challenge (8 Byte, kein Padding) mit einem 3DES-Schlüssel (ECB-Modus) chiffriert, siehe Abb. E.6. Die Angabe einer Algorithmus-Referenz ist nicht erforderlich (Algorithmus-Referenz implizit).

eGK	Sys	Server	eGK-Operationen
GET CHALLENGE <RND.eGK>			Berechnung einer Zufallszahl
		INTERNAL AUTHENTICATE <RND.eGK> <ENC [SK.eGK.ENC, RND.eGK]>	
EXTERNAL AUTHENTICATE <ENC [SK.eGK.AUT, RND.eGK]> <>			Die eGK entschlüsselt das Kryptogramm mit dem SK.eGK.AUT, prüft dann, ob das Ergebnis mit zwischengespeicherten Zufallszahl übereinstimmt. Falls ja, wird ein Sicherheitsstatus gesetzt (Entity mit SK.eGK.AUT authentisiert).

Abbildung E. 6- Challenge/Response-Verfahren mit symmetrischem Schlüssel

E.6 Client/Server- Authentisierungsverfahren mit X.509-Zertifkat

Beim Client/Server-Authentisierungsverfahren wird von der eGK eine Signatur mit dem Private Key des Karteninhabers (Cardholders) PrK.CH.AUT berechnet. Hierbei wird der eGK die Digestinfo (oder ein funktional der Digestinfo entsprechender String) übergeben. Die Karte wendet dann das Paddingverfahren nach [PKCS#1] EMSA-PKCS1-v1_5 an, siehe Tabelle E.1. Die Angabe einer Algorithmus-Referenz ist nicht erforderlich (Algorithmus-Referenz implizit).

Tabelle E. 2 - Digital Signature Input für Client/Server-Authentisierung

DSI	Input-Elemente
'00' || '01' || PS || '00' || T	PS = Padding String aus Octets mit 'FF' T = Daten im Datenfeld des Kommandos INTERNAL AUTHENTICATE (Die Länge des Signatur-Inputs T darf gemäß [DIN66291-4] höchstens 40% der Moduluslänge des Signatur-Schlüssels betragen). Der formatierte Octet-String muss aus k Octets bestehen, wobei k die Länge des Modulus des privaten Authentisierungsschlüssels in Octets bezeichnet.

Anhang F

F1 - Abkürzungen

Kürzel	Erläuterung
AID	Application Identifier
AlgRef.	Algorithmus Referenz
AM	Zugriffsmodus (Access Mode)
AMS	Anwendungs-Management-System (Application Management System) sh. auch CAMS
ARR	Access Rule Reference
ASN.1	Abstract Syntax Notation One
AT	Authentication Template
ATR	Answer-to-Reset
AUT	Authentisierung (Authentication)
AVS	Apothekenverwaltungssystem
B	Byte
BA	Berufsausweis für Mitarbeiter im Gesundheitswesen
BCD	Binär kodierte Dezimalzahl (Binary Coded Decimal)
BER	Basic Encoding Rules
C	Zertifikat (Certificate)
CA	Zertifizierungsinstanz (Certification Authority)
CAMS	Karten-Anwendungs-Managementsystem (Card Application Management System) sh.auch AMS
CMS	Karten-Management-System (Card Management System)
CAR	Referenz der Zertifizierungsinstanz (Certification Authority Reference)
CBC	Cipher Block Chaining
CC	kryptografische Prüfsumme (Cryptographic Checksum)
CDB	Check Digit Byte
CG	Kryptogramm (Cryptogram)
CH	Karteninhaber (Cardholder)
CHA	Berechtigung des Karteninhabers (Certificate Holder Authorization)
CHR	Referenz des Karteninhabers (Certificate Holder Reference)

Kürzel	Erläuterung
CIA	kryptografische Informationsanwendung (Cryptographic Information Application)
CIO	kryptografische Informationsobjekte (Cryptographic Information Objects)
CLA	Class-Byte eines Befehls
COS	Kartenbetriebssystem (Card Operating System)
CPI	Kennung des Zertifikatsprofils (Certificate Profile Identifier)
CRT	Control Reference Template
CS	CertSign (CertificateSigning)
CSP	Zertifizierungsdiensteanbieter (Certificate Service Provider)
CT	Confidentiality Template
CV	Card Verifiable (Zertifikat)
CWA	Cen Workshop Agreement
C2C	Card-to-Card
C2S	Card-to-Server
D	Verzeichnis (Directory)
DE	Datenelement
DER	Distinguished Encoding Rules
DES	Data Encryption Standard
DF	Dedicated File
DI	Baud rate adjustment factor
DIR	Verzeichnis (Directory)
DM	Display Message
DO	Datenobjekt
DSI	Digital Signature Input
DST	Digital Signature Template
EAL	Evaluation Assurance Level
ECB	Electronic Code Book
EF	Elementary File
EFID	Short EF Identifier
eGK	elektronische Gesundheitskarte
eHC	electronic Health Card
EHIC	European Health Insurance Card
ENC	Verschlüsselung (Encryption)

Kürzel	Erläuterung
ENC( )	verschlüsselte Daten (Encrypted data)
EOF	Dateiende (End-of-File)
ES	Elektronische Signatur
FCP	File Control Parameter
FI	Clock rate conversion factor
FID	Dateikennung (File Identifier)
FM	Dateimanagement (File Management)
GP	Global Platform
HB	Historical Bytes
HBA	Heilberufsausweis
HCA	Gesundheitsanwendung (Health Care Application)
HI	Krankenversicherung (Health Insurance)
HID	Versichertendaten (Health Insurance Data)
HP	Heilberufler (Health Professional)
HPC	Heilberufsausweis (Health Professional Card)
ICC	Integrated Circuit Card
ICCSN	ICC Serial Number
ICM	IC-Herstellerkennung (IC Manufacturer)
ID	Identifier
IFD	Interface Device
IFSC	Information Field Size Card
IFSD	Information Field Size Device
IIN	Kennung des Kartenanbieters (Issuer Identification Number)
IV	Initial Value
KD	Key derivation Data
Key Ref.	Schlüssel-Referenz (Key Reference)
LSB	Least Significant Byte(s)
MAC	Message Authentication Code
MF	Master File
MII	Major Industry Identifier
MSB	Most Significant Byte
MSE	Manage Security Environment
OID	Objektkennung (Object Identifier)
PI	Padding Indicator

Kürzel	Erläuterung
PIN	Personenkennung (Personal Identification Number)
PIX	Proprietary Appl. Prov. Extension
PK	Öffentlicher Schlüssel (Public Key)
PuK	Öffentlicher Schlüssel (Public Key)
PKI	Public Key Infrastructure
PP	Schutzprofil (Protection Profile)
PPS	Protocol Parameter Selection
PrK	Privater Schlüssel (Private Key)
PRND	Padding Random Number
PSO	Perform Security Operation
PVS	Praxisverwaltungssystem
Q	Qualifiziert
QES	Qualifizierte Elektronische Signatur
R	Rollenkennung
RC	Retry Counter
RCA	Wurzelinstanz (Root CA)
RD	Referenzdaten (Reference Data)
RF	Radio Frequency
RFC	Request for Comment
RID	Registered Application Provider Id.
RND	Zufallszahl (Random Number)
RSA	Algorithmus von Rivest, Shamir, Adleman
SVR	Server
SC	Sicherheitsbedingung (Security Condition)
SE	Sicherheitsumgebung (Security Environment)
SFID	Short EF Identifier
SIG	Signatur (Signature)
SK	geheimer Schlüssel (Secret Key)
SM	Secure Messaging
SMC	Sicherheitsmodulkarte (Security Module Card)
SMK	SM-Schlüssel (SM key)
SN	Serien-Nummer (Serial Number)
SSC	Send Sequence Counter
SSCD	Sichere Signaturerstellungseinheit (Secure Signature Creation Devi-

Kürzel	Erläuterung
	ce)
SSL	Security Sockets Layer
SST	Self Service Terminal (eKiosk)
SW	Status Word
TDES	Siehe 3DES
TLV	Tag Length Value
TC	sicherer Kanal (Trusted Channel)
UID	Benutzerkennung (User Identification)
UQ	Usage Qualifier
UTF8	8-bit Unicode Transformation Format
ZDA	Zertifizierungs-Dienste-Anbieter
3DES	Triple-DES

F2 - Glossar

Das Projektglossar wird als eigenständiges Dokument zur Verfügung gestellt.

F3 - Abbildungsverzeichnis

F4 - Tabellenverzeichnis

Anhang A

Anhang B

Anhang C

Anhang D

F5 - Referenzierte Dokumente

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
[ALGCAT]	Geeignete Algorithmen zur Erfüllung der Anforderungen nach §17 Abs. 1 bis 3 SigG vom 22. Mai 2001 in Verbindung mit Anlage 1 Abschnitt I Nr. 2 SigV vom 22. November 2001, Algorithmenkatalog 2005 vom 02.01.2005, 30. März 2005, Bundesanzeiger Nr. 59, S. 4695-4696 , siehe www.bundesnetzagentur.de
[DIN66291-1]	DIN V66291-1 (2000): Chipkarten mit Digitaler Signatur-Anwendung/Funktion nach SigG/SigV Teil 1: Anwendungsschnittstelle
[DIN66291-4]	DIN V66291-4: 2002 Chipkarten mit Digitaler Signatur-Anwendung/Funktion nach SigG/SigV Teil 4: Grundlegende Sicherheitsdienste
[ECC-2]	CEN TC224 (Mai 2005): European Citizen Card, Part 2 – Logical Data Structures and Security Services
[gemSpec_eGK_P2]	gematik (24.08.2007): Einführung der Gesundheitskarte – Die Spezifikation elektronische Gesundheitskarte ; Teil 2 – Anwendungen und anwendungsspezifische Strukturen Version 1.5.0, www.gematik.de
[HBA]	Bundesärztekammer et al. (28.05.2006): Heilberufs-Ausweis und Security Module Card, Teil 1 bis 3 V2.1, http://www.baek.de/page.asp?his=1.134.3421.4132
[ISO3166-1]	ISO/IEC 3166-1: 1997 Codes for the representations of names of countries – Part 1: Country codes
[ISO7816-2]	ISO/IEC 7816-2: 2007 (2nd edition) Identification cards — Integrated circuit cards — Part 2: Cards with contacts: Dimensions and location of the contacts
[ISO7816-3]	ISO/IEC 7816-3: 2006 (2nd edition) Identification cards - Integrated circuit cards with contacts - Part 3: Electrical interface and transmission protocols
[ISO7816-4]	ISO/IEC 7816-4: 2004 (2nd edition) Identification cards - Integrated circuit cards - Part 4: Organization, security and commands for interchange
[ISO7816-8]	ISO/IEC 7816-8: 2004 (2nd edition) Identification cards - Integrated circuit cards - Part 8: Commands for security operations
[ISO7816-9]	ISO/IEC 7816-9: 2004 (2nd edition) Identification cards - Integrated circuit cards - Part 9: Commands for card management

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
[ISO7816-13]	ISO/IEC 7816-13: 1st edition 2007 Identification cards - Integrated circuit cards - Part 13: Commands for application management in multi-application environment
[ISO9564-1]	ISO 9564-1 Banking – Personal Identification Number management and Security, Part 1: PIN protection principles and techniques, 1999
[ISO9796-2]	ISO 9796-2: 2002 Information technology – Security techniques – Digital signature schemes giving message recovery – Part 2: Mechanisms using a hash function Update
[ISO11568-2]	ISO/IEC 11568-2: Banking -- Key management (retail) -- Part 2: Symmetric ciphers, their key management and life cycle
[PKCS#1]	PKCS #1 RSA Cryptography Standard V2.1: June 14, 2002
[prEN14890-1]	prEN 14890-1 (Draft: February 2007) Application Interface for smart cards used as secure signature Ccreation Devices - Part 1: Basic services