

Expertise Versichertenkarte nach Art. 42a KVG

Teil 1 / 2

Version	Autor	Datum	Kommentar
2.9b	R. Eberhard S. Staible	7. Juni 2011	

Inhaltsverzeichnis

1	Management Summary.....	4
1.1	Beurteilung der technischen Sachlage.....	4
1.2	Beurteilung der organisatorischen Sachlage.....	5
1.3	Empfehlungen für das weitere Vorgehen.....	6
2	Referenzen.....	7
2.1	Über Keyon.....	7
2.2	Abkürzungen.....	7
2.3	Definitionen.....	8
2.4	Referenzen auf weitere Dokumente.....	8
2.5	Übersicht über die Dokumentenstruktur.....	10
3	Auftrag.....	11
3.1	Anstoss für den Auftrag.....	11
3.2	Ziele.....	12
3.3	Aufgaben.....	12
4	Grundlagen.....	13
4.1	eCH-0064.....	13
4.1.1	Middleware.....	14
4.2	Public Key Infrastructure.....	15
5	Analyse eCH-0064 – Detailspezifikationen.....	17
5.1	Einleitung.....	17
5.2	Übersicht über die Ergebnisse.....	17
5.2.1	Detailspezifikation der Schweizerischen Post.....	19
5.2.2	Detailspezifikation der SASIS.....	20
5.3	eCH-0064: Kapitel 3.4 Dateisystem.....	21
5.3.1	EF.ARR.....	21
5.3.2	EF.ATR.....	21
5.3.3	EF.DIR.....	23
5.3.4	EF.ICCSN.....	24
5.3.5	EF.ID.....	26
5.3.6	EF.AD.....	28
5.3.7	EF.LOG.....	29
5.3.8	EF.CVC.PDC.....	29
5.3.9	EF.PK.CA_ORG_PDC.....	30
5.3.10	EF.CVC.CA_ORG_PDC.....	31
5.3.11	EF.CVC.CA_ORG_HPC.....	32
5.3.12	Verwaltung der Notfalldaten.....	33
5.4	eCH-0064: Kapitel 3.6.4.1, Card to Card-Authentication and Authorization.....	34
5.4.1	Card to Card Authentication and Authorization.....	34
5.4.2	Terminalauthentisierung.....	37
5.4.3	CA-Hierarchie.....	38
5.5	eCH-0064: Kapitel 3.6.4.1, PIN-Management.....	40
5.6	eCH-0064: Kapitel 5, Kantonale Modellversuche nach Artikel 16 [VVK].....	42
5.7	eCH-0064: Kapitel 6, Definition Zertifikate.....	45
5.7.1	CV-Zertifikatsformat.....	45
5.7.2	Signaturformat im Card to Card-Authentisierungsverfahren.....	49
6	HPC.....	50
6.1	Technische Beschreibung der FMH-HPC.....	50
7	Ursachen der Abweichungen.....	52

7.1	Kompetenzen, Verantwortung und Absprachen	52
7.1.1	Aus Sicht EDI resp. BAG	52
7.1.2	Aus Sicht der Versicherer	52
7.2	Technische Verfahren und Standards	53
8	Empfehlungen für das weitere Vorgehen	54
8.1	Allgemeine Empfehlungen	54
8.1.1	TAV VVK-EDI	54
8.1.2	Testdaten und Referenzimplementation	54
8.1.3	Public Key Infrastruktur	55
8.2	Weiteres Vorgehen	56
8.2.1	Weiterentwicklung des bestehenden Systems	56
8.2.2	Spezifikation des zukünftigen Systems	58
9	Anhang 1: Hex-Dump der Versicherungskarten	61
9.1	VK-Post	61
9.1.1	Hex-Dump	62
9.2	VK-SASIS	64
9.2.1	Hex-Dump	64
10	Anhang 2: Analyse Dateisystem	69
10.1	Definitionen	69
10.2	Dateisystem [eCH64]	69
10.2.1	EF.DIR	69
10.2.2	EF.ICCSN	71
10.2.3	EF.ID	72
10.2.4	EF.AD	73
10.2.5	EF.LOG	75
10.2.6	EF.BGTD	76
10.2.7	EF.IMMD	77
10.2.8	EF.TPLD	78
10.2.9	EF.KHUF	79
10.2.10	EF.ZUSE	80
10.2.11	EF.MEDI	81
10.2.12	EF.ALLG	82
10.2.13	EF.ADDR	83
10.2.14	EF.VERF	84
10.2.15	EF.ATR	85
10.2.16	EF.PIN	86
10.2.17	EF.StatusPIN	87
10.2.18	EF.S _B	88
10.2.19	EF.ARR ₁	89
10.2.20	EF.ARR ₂	90
10.2.21	EF.CVC.PDC	91
10.2.22	EF.PK.CA_ORG_PDC	92
10.2.23	EF.CVC.CA_ORG_PDC	93
10.2.24	EF.CVC.CA_ORG_HPC	94
10.2.25	EF.AUT	95
10.3	Dateien ausserhalb des Standards	96
10.3.1	EF.Version	96
10.3.2	iEF.C2CSTATE	97
10.3.3	EF.GPKeys	98
10.3.4	Personal Unlocking Key	99
10.3.5	SK.Admin	99

1 Management Summary

Die Versichertenkarte identifiziert Leistungsbezüger (resp. versicherte Personen) und beinhaltet die wichtigsten administrativen Daten. Zukünftig besteht für die Versicherten die Möglichkeit persönliche und medizinische Daten auf der Versichertenkarte speichern zu lassen. Die Verwaltung der Daten auf der Versichertenkarte setzt eine erfolgreiche Authentisierung des Fachpersonals über die HPC voraus, welche die Leistungserbringer (Ärzte, Apotheker, etc.) identifiziert. Bis zum heutigen Zeitpunkt sind nur administrative Daten auf der Versichertenkarte gespeichert. Aufgrund fehlender Software und HPC Karten, die mit CV-Zertifikaten personalisiert sind, werden noch keine medizinischen Daten auf die Versichertenkarte gespeichert.

Die Expertise untersucht die Abweichungen zwischen der [eCH64] Spezifikation und den beiden Detailspezifikationen [SPEC-POST] und [SPEC-SASIS]. Sie unterstützt das BAG bei der Beurteilung der Abweichungen der beiden Detailspezifikationen von der [VVK-EDI] respektive [eCH64] und gibt Empfehlungen für ein mögliches weiteres Vorgehen ab.

Die Expertise hat ergeben, dass die Versichertenkarten der Post und der SASIS unter Verwendung einer Middleware einwandfrei funktionieren. Beide Umsetzungsvarianten weisen Abweichungen von den rechtlichen Vorgaben auf, die zum Teil auf Fehler und Interpretationsspielraum in den zugrundeliegenden Standards beruhen.

1.1 Beurteilung der technischen Sachlage

[eCH64] spezifiziert das System der Schweizerischen Versichertenkarte und bezieht sich, analog zu [VVK-EDI], auf unterschiedliche ISO Normen und RFC Standards. Falls spezifische Sachverhalten in [eCH64] nicht näher erläutert sind, gelten die jeweils übergeordneten ISO Normen und RFC Standards.

In [eCH64] Kapitel 2.2 ist festgehalten, dass es sich hierbei um keinen Implementationsstandard, sondern ein Konzeptions-, Struktur- und Verfahrensstandard handelt. Demnach sind weitere Detailspezifikationen, wie sie in [SPEC-POST] und [SPEC-SASIS] erarbeitet wurden, für die Implementierung einer Versichertenkarte notwendig.

Die [SPEC-POST] und die [SPEC-SASIS] wurden weitgehend autonom, mit unzureichender übergeordneter Koordination, erarbeitet¹. Da verschiedene Punkte aus [eCH64] und den referenzierten ISO Normen und RFC Standards interpretierbar sind, weichen die [SPEC-POST] und die [SPEC-SASIS] untereinander und von [eCH64] ab. Weiter sind einzelne Vorgaben aus [eCH64], [VVK-EDI] sowie einer referenzierten ISO Norm² fehlerhaft, was sich aufgrund unzureichender Koordination zwischen den

¹ Aufgrund verschiedener Sachverhalte muss davon ausgegangen werden, dass sich die Schweizerische Post sowie die SASIS partiell abgesprochen haben.

² ISO 21549-5

Versicherern negativ auf die Übereinstimmung der [SPEC-POST] und [SPEC-SASIS] auswirkt.

Obwohl beide Spezifikationen Abweichungen gegenüber [eCH64] aufweisen, ist die [SPEC-POST] näher an [eCH64] als die [SPEC-SASIS]. Eine Übersicht über die Ergebnisse ist in Kapitel 5.2 / Seite 17 aufgeführt.

- Die [SPEC-POST] hält sich grundsätzlich an die Vorgaben von [eCH64], weicht aber, aufgrund unzureichender Koordination, in verschiedenen Punkten von [eCH64] ab. Die Abweichungen ergeben sich primär aus Interpretationen der Standards sowie Fehlern in der Umsetzung.
- Die [SPEC-SASIS] hält sich mehrheitlich an die Vorgaben von [eCH64], weicht aber, aufgrund unzureichender Koordination, in verschiedenen Punkten von [eCH64] ab.

Die Abweichungen ergeben sich primär aus der Verwendung eines von [eCH64] abweichenden Zertifikatsformats und von [eCH64] abweichenden Schlüssellängen oder Hash-Algorithmen, was einen entsprechenden Einfluss auf verschiedene Prozesse und Komponenten im System der Versichertenkarte hat. Zudem ergeben sich Abweichungen aus Interpretationen der Standards und Fehlern in der Umsetzung.

Die praktischen Analysen der Versichertenkarte im Zusammenhang mit einer HPC haben gezeigt, dass die Versichertenkarten gemäss den jeweiligen Spezifikationen umgesetzt wurden und unter Verwendung einer geeigneten Middleware interoperabel zueinander sind. Einzelheiten über die praktischen Analysen sind in [Expertise VK - Tech] aufgeführt. Die Detailspezifikationen beider Kartenherausgeber sind in mehreren, Punkten so zu ergänzen, dass sie als Grundlage für die Entwicklung einer Middleware verwendet werden können. Die nach Art. 2 Abs. 2 VVK geforderte Kompatibilität der Versichertenkarten im Sinne des Gesetzgebers ist aus Sicht der Autoren dieser Expertise jedoch nicht gegeben.

1.2 Beurteilung der organisatorischen Sachlage

Die Verantwortlichkeiten im Zusammenhang mit der Spezifikation und Umsetzung der Versichertenkarte wurden unzureichend koordiniert. Gemäss [VVK] und [VVK-Erläuterung]

- sorgen die Versicherer dafür, dass die von ihnen herausgegebenen Versichertenkarten untereinander kompatibel sind
- ist das EDI verantwortlich für die Festlegung der hierfür notwendigen technischen Einzelheiten, inklusive der Normen und Standards. Das EDI stützt sich hierbei auf Empfehlung einer Fachgruppe des Vereins eCH

Die Koordination und Zusammenarbeit zwischen den Versicherern untereinander sowie zwischen den Versicherern und dem EDI war unzureichend und führte zu den unterschiedlichen Spezifikationen und Umsetzungen der Versichertenkarte. Einzelheiten hierzu sind in Kapitel 7 / Seite 52 aufgeführt.

1.3 Empfehlungen für das weitere Vorgehen

Die Ziele des Systems der Versichertenkarte sind in [VVK-Erläuterung] festgehalten und können mit den aktuell verfügbaren Versichertenkarten der Post und SASIS unter Verwendung einer geeigneten Middleware umgesetzt werden. Um die Nachhaltigkeit der Lösung sicherzustellen, sollten jedoch die Grundlagen im Zusammenhang mit den technischen Spezifikationen neu geregelt werden.

Die Aktivitäten für das weitere Vorgehen können so aufgeteilt werden, dass diese die Weiterentwicklung des bestehenden Systems sowie die Spezifikation des zukünftigen Systems gewährleisten. Die Arbeiten an den beiden Aktivitäten sollten parallel durchgeführt werden. Einzelheiten hierzu sind in Kapitel 8 / Seite 54 aufgeführt.

2 Referenzen

2.1 Über Keyon

Keyon ist ein führender Anbieter von Lösungen und Dienstleistungen in den Bereichen IT-Sicherheit und Software Engineering, Public Key Infrastrukturen sowie rechtsgültiger Langzeitarchivierung. Das Unternehmen verfügt über erstklassige Referenzen und hat eine Vielzahl strategischer Projekte für den Bund und seine Kunden aus den Bereichen Finanz, Versicherung, Handel, Industrie und Telekommunikation umgesetzt.

2.2 Abkürzungen

Bezeichnung	Beschreibung
AID	Application Identifier
AM	Access Mode
APDU	Application Protocol Data Unit nach ISO 7816
API	Application Programming Interface
CA	Certification Authority
CLA	Class-Byte of a command APDU
CVC	Card Verifiable Certificate
DO	Data Object
EDI	Eidgenössische Departement des Inneren
FID	File ID
HPC	Health Professional Card
INS	Instruction-Byte of a command APDU
LC	Length Command
LE	Length Expected
MF	Master File
P1	Parameter P1 of a command APDU
P2	Parameter P2 of a command APDU
PKI	Public Key Infrastruktur
SASIS	SASIS AG, beauftragte der santésuisse.
SC	Security Condition
SFID	Short File ID (implizite, interne Selektion)
VK	Versichertenkarte

2.3 Definitionen

Bezeichnung	Beschreibung
VK-Post	Versichertenkarte umgesetzt nach den Spezifikationen der Schweizerischen Post
VK-SASIS	Versichertenkarte umgesetzt nach den Spezifikationen der SASIS

2.4 Referenzen auf weitere Dokumente

Referenz	Beschreibung
[eCH106]	eCH-0106 - Spezifikation für das System Versichertenkarte Offline Card-to-Card Authentication and Authorization, V1.0 vom 22. März 2010
[eCH64]	eCH-0064 - Spezifikationen für das System Versichertenkarte, V1.0 vom 4. Februar 2008
[EIDI-V]	Verordnung des EFD über elektronische Daten und Informationen, SR 641.201.511, Stand am 1. Januar 2010
[Expertise VK - Tech]	Expertise Versichertenkarte nach Art. 42a KVG, Teil 2 / 2, Praktische Analyse, Version 1.3
[FMH-HPC]	Technische Beschreibung, Version 1.0 vom 01.09.2009
[ISO 21549-5]	E DIN EN ISO 21549-5:2006-09 Draft Health informatics — Patient healthcard data — Part 5: Identification data (ISO/DIS 21549-5:2006)
[ISO 21549-6]	ISO/DIS 21549-6 Health informatics — Patient healthcard data — Part 6: Administrative data
[ISO 7816-4]	ISO/IEC 7816-4:2005(E) Identification cards - Integrated circuit cards - Part 4: Organization, security and commands for interchange
[ISO 8825-1]	ISO/IEC 8825-1:2008 Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)
[ISO 9796-2]	ISO/IEC 9796-2:2002 Information technology — Security techniques — Digital signature schemes giving message recovery Part 2: Integer factorization based mechanisms

[ISO 9798-3]	ISO/IEC 9798-3 Information technology – Security techniques – Entity authentication Part 3: Mechanisms using digital signature techniques
[SPEC-POST]	Implementierungsanleitung für die Versichertenkarte nach eCH-0064, V1.0.0
[SPEC-SASIS]	SASIS - Versichertenkarte, Detailspezifikation, Version 1.4ech, Ausgabe: 09.08.2010
[TAV EIDI-V]	Technische und administrative Vorschriften für Zertifizierungsdienste im Bereich der EIDI-V im Zusammenhang mit der Ausstellung von Zertifikaten basierend auf fortgeschrittenen Signaturen, SR 641.201.511.1 / Anhang, Inkrafttreten 1. Januar 2010 <small>Referenziert in Art. 2 Abs. 4 [EIDI-V]</small>
[TAV ZertES]	Technische und administrative Vorschriften über Zertifizierungsdienste im Bereich der elektronischen Signatur, SR 943.032.1 / Anhang, Inkrafttreten: 1. Dezember 2006 <small>Referenziert in Art. 3 Abs. 2 und Art. 13 [VZertES]</small>
[VVK]	Verordnung über die Versichertenkarte für die obligatorische Krankenpflegeversicherung vom 14. Februar 2007 (Stand am 1. Januar 2009), SR 832.105
[VVK-EDI]	Verordnung des EDI über die technischen und grafischen Anforderungen an die Versichertenkarte für die obligatorische Krankenpflegeversicherung, vom 20. März 2008 (Stand am 1. April 2008), SR 832.105.1
[VVK-EDI-Änderung]	Schreiben vom 9. August 2010 des Eidgenössisches Departement des Innern EDI mit dem Titel: „Erläuterungen zu den Änderungen der Verordnung des EDI über die technischen und grafischen Anforderungen an die Versichertenkarte für die obligatorische Krankenpflegeversicherung (VVK-EDI), welche voraussichtlich am 1. November 2010 in Kraft treten wird“
[VVK-EDI-Erläuterung]	Erläuterung zur VVK-EDI vom 20. März 2008
[VVK-Erläuterung]	Erläuterung zur VVK vom 14. Februar 2007
[VZertES]	Verordnung über die elektronische Signatur, SR 943.032, Stand 1. Januar 2005

2.5 Übersicht über die Dokumentenstruktur



Expertise Teil 1/2 Version 2.9b	Primäres Dokument <ul style="list-style-type: none"> ▪ Management Summary ▪ Analyse der Spezifikation ▪ Technische Analyse des Sachverhalts, welche keine Card to Card Authentisierung voraussetzten ▪ Rechtliche und organisatorische Analyse des Sachverhalts ▪ Empfehlungen
Expertise Teil 2/2 Version 1.4	Technische Analyse des Sachverhalts, welche eine Card to Card Authentisierung voraussetzten
APDU Protokoll	APDU Protokoll als Grundlage für die Expertise Teil 2/2

3 Auftrag

3.1 Anstoss für den Auftrag

Die technischen Anforderungen an die Versichertenkarte sind in der Verordnung des EDI über die technischen und grafischen Anforderungen an die Versichertenkarte für die obligatorische Krankenpflegeversicherung (VVK-EDI) und im Standard „eCH-0064 – Spezifikationen für das System Versichertenkarte“ festgehalten. Dieser ist aber kein Implementierungsstandard und kann nur durch eine zusätzlich erarbeitete Detailspezifikation nach Kap. 3.6.4.5 (Anforderungen an die Detailspezifikation) des Standards umgesetzt werden.

Die Verordnung über die Versichertenkarte für die obligatorische Krankenpflegeversicherung (VVK) überträgt den Versicherern die Aufgabe, untereinander kompatible Karten herzustellen und damit auch die Erarbeitung der Detailspezifikation. Zudem verlangt die VVK auch, dass die Bearbeitung der medizinischen Daten auf der Versichertenkarte (VK) nur mit Hilfe elektronischer Ausweise der medizinischen Leistungserbringer (HPC) möglich sein darf. Die Detailspezifikation und die Spezifikationen der HPC müssen daher aufeinander abgestimmt sein.

Die Versichertenkarten werden von zwei verschiedenen Kartenherstellern, der SASIS AG und der Schweizerischen Post hergestellt. Beide Kartenhersteller haben die Vorgaben im Standard eCH-0064 und die ausführenden technischen Detailspezifikationen unterschiedlich umgesetzt. Die Kompatibilität zwischen den Versichertenkarten und damit auch die Interoperabilität zwischen den Versichertenkarten und der Health Professional Card muss angesichts dieser Ausgangslage in einer ersten Phase durch geeignete Software (Middleware) sichergestellt werden. In einer zweiten Phase soll eine Konvergenzlösung gefunden werden. Dafür braucht es insbesondere Kenntnis der Unterschiede der beiden Detailspezifikationen untereinander und Unterschiede der Detailspezifikationen zu den rechtlichen Vorgaben der Versichertenkarte.

Es bleibt festzuhalten, dass die Versichertenkarte kein privates Produkt sondern ein öffentliches Gut ist. Die Kartenherausgeber müssen somit sämtliche Spezifikationen offenlegen, soweit diese für die Kompatibilität der Versichertenkarten eine Rolle spielen.

3.2 Ziele

Das BAG verfolgt folgende Ziele mit dieser Prüfung:

- Die Funktionstüchtigkeit der beiden Versichertenkarten für alle rechtlich geforderten Anwendungen (administrative Daten, medizinische Daten und kantonale Modellversuche) ist klar ersichtlich.
- Differenzen der beiden Detailspezifikationen und effektiv produzierten Versichertenkarten sind klar ersichtlich.
- Abweichungen der beiden Detailspezifikationen und effektiv produzierten Versichertenkarten von der Verordnung des EDI über die technischen und grafischen Anforderungen an die VVK-EDI und dem Standard eCH-0064 sind klar ersichtlich.
- Der Bericht unterstützt das BAG in einer möglichen Überarbeitung der VVK-EDI und des Standards eCH-0064 und bei der Schaffung einer zukünftigen Konvergenzlösung.

3.3 Aufgaben

Um die Zielsetzungen zu erreichen, sollen folgende Arbeiten ausgeführt werden:

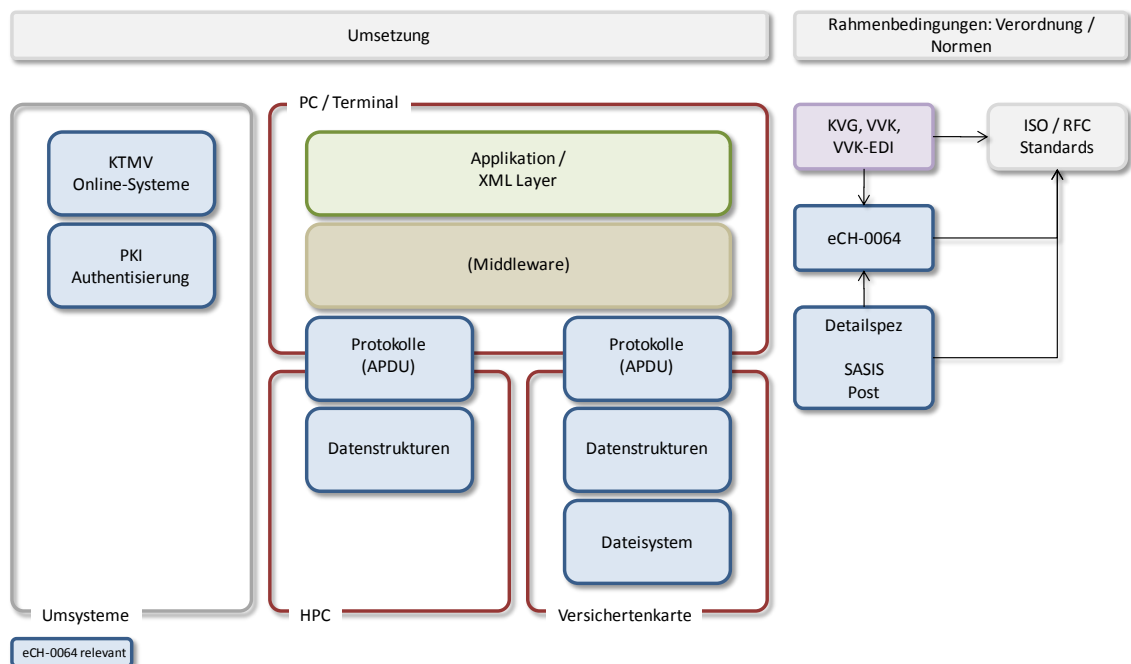
- Die Detailspezifikationen von SASIS und Post sind auf Vollständigkeit zu prüfen. Fehlende Angaben sind von den Kartenherausgebern allenfalls einzufordern. Die Detailspezifikationen müssen sämtliche Angaben enthalten, um das Lesen der administrativen Daten und die Bearbeitung der medizinischen Daten zu ermöglichen. Zudem ist das Angebot der beiden Versichertenkarten für kantonale Modellversuche wie Schlüssel, Zertifikate (lesen/schreiben bzw. nachladen), PIN's und Daten darzustellen.
- Die Funktionsfähigkeit des Systems mit effektiv produzierten Karten (FMH, SASIS und Post) ist zu beurteilen und die Ergebnisse sind festzuhalten.
- Die Differenzen der beiden Detailspezifikationen und effektiv produzierten Karten untereinander sind festzuhalten.
- Die Abweichungen der Detailspezifikationen und effektiv produzierten Karten von der VVK-EDI und eCH-0064 sind festzuhalten
- Nach Möglichkeit sind Anpassungen von VVK-EDI und eCH-0064 zu empfehlen.

4 Grundlagen

Um die Abweichungen der Spezifikationen und deren Auswirkungen auf Umsysteme besser verstehen zu können, werden in diesem Kapitel kurz die wichtigsten Grundlagen des Systems der Schweizerischen Versichertenkarte auf technischer Ebene beschrieben.

4.1 eCH-0064

[eCH64] spezifiziert das System der Schweizerischen Versichertenkarte unter Berücksichtigung international gültiger Normen auf technischer Ebene. Folgend ist eine Übersicht über die Rahmenbedingungen gegeben und wie diese in Bezug zur technischen Umsetzung der Systems Versichertenkarte stehen.



[eCH64] spezifiziert das System der Schweizerischen Versichertenkarte und bezieht sich, analog zu [VVK-EDI], auf unterschiedliche ISO Normen und RFC Standards. Falls spezifische Sachverhalte in [eCH64] nicht näher erläutert sind, gelten die jeweils übergeordneten ISO Normen und RFC Standards.

In [eCH64] Kapitel 2.2 ist festgehalten, dass es sich hierbei um keinen Implementationsstandard, sondern ein Konzeptions-, Struktur- und Verfahrensstandard handelt. Weiter ist in [eCH64] Kapitel 2.1 festgehalten, dass nur ein Minimalset an technologischen Vorgaben definiert würde, um möglichst vielen Anbietern am Markt eine Umsetzung der Vorgaben zu ermöglichen.

Da verschiedene Punkte aus [eCH64] und den referenzierten ISO Normen und RFC Standards interpretierbar sind, müssen diese für die Implementierung einer Versichertenkarte detaillierter spezifiziert werden. Analoges gilt für allfällige Anpassungen im Zusammenhang mit Geschäftsprozessen oder gesetzlichen Vorgaben. Zudem müssen in den Detailspezifikationen einzelne fehlerhafte Vorgaben aus [eCH64] korrigiert werden.

[SPEC-POST] und [SPEC-SASIS] sind entsprechende Detailspezifikationen, welche eine technische Implementierung einer Versichertenkarte ermöglichen sollen. Sie präzisieren die Vorgaben aus [eCH64] und den referenzierten ISO Normen und RFC Standards und sind verbindlich für die Implementierung der Versichertenkarte. Die Detailspezifikationen beziehen sich auf Datenstrukturen, Dateisysteme, Protokolle und Authentisierungsverfahren (jeweils blau hinterlegt). Nicht berücksichtigt in den Detailspezifikationen sind sämtliche Prozesse im Zusammenhang mit der Ausstellung und Verwaltung der Versichertenkarte sowie der fachspezifische Einsatz der Versichertenkarte in einem Geschäftsprozess.

Hinweis	Die Interpretation von [eCH64] und den referenzierten ISO Normen und RFC Standards setzt ein umfassendes, tiefgreifendes technisches Fachwissen voraus. Verschiedene Sachverhalten, Datenstrukturen und Protokollbeschreibungen werden auf Ebene Bits und Bytes diskutiert und jeweils mit Sachverhalten, Datenstrukturen oder Protokollbeschreibungen aus anderen Standards verknüpft. Die in [eCH64] referenzierten ISO Normen und RFC Standards sind interpretierbar und [eCH64] verlangt daher zusätzliche Detailspezifikationen. Ihre Erarbeitung muss aber übergeordnet koordiniert werden.
---------	--

4.1.1 Middleware

Als Middleware werden anwendungsneutrale Schnittstellen bezeichnet, die zwischen unterschiedlichen Komponenten vermitteln, so dass die Komplexität dieser Komponenten verborgen wird.

Im Umfeld der Versichertenkarte war vorgesehen, die Applikationen über die kartenspezifische Schnittstelle gemäss [eCH64] anzubinden. Eine übergeordnete, kartunenabhängige Schnittstelle in Form eines API resp. einer Middleware war in [eCH64] nur empfohlen³, jedoch nicht zwingend.

Hinweis	Die Erarbeitung einer allfälligen API müsste zwingend übergeordnet koordiniert werden. Sie muss aus Sicht der Applikation unabhängig von den Eigenschaften der spezifischen Karten sein.
---------	--

³ [eCH64], Kapitel 3.6.4.5

4.2 Public Key Infrastructure

Jeder Inhaber einer Versichertenkarte und jeder Inhaber einer HPC hat eine digitale Identität in Form eines oder mehrerer digitaler Zertifikate⁴. Die Zertifikate werden unter Berücksichtigung definierter Registrier-, Ausgabe- und Verwaltungsprozessen von CAs ausgegeben.

Grundsätzlich wird in [eCH64] zwischen den folgenden Zertifikaten unterschieden:

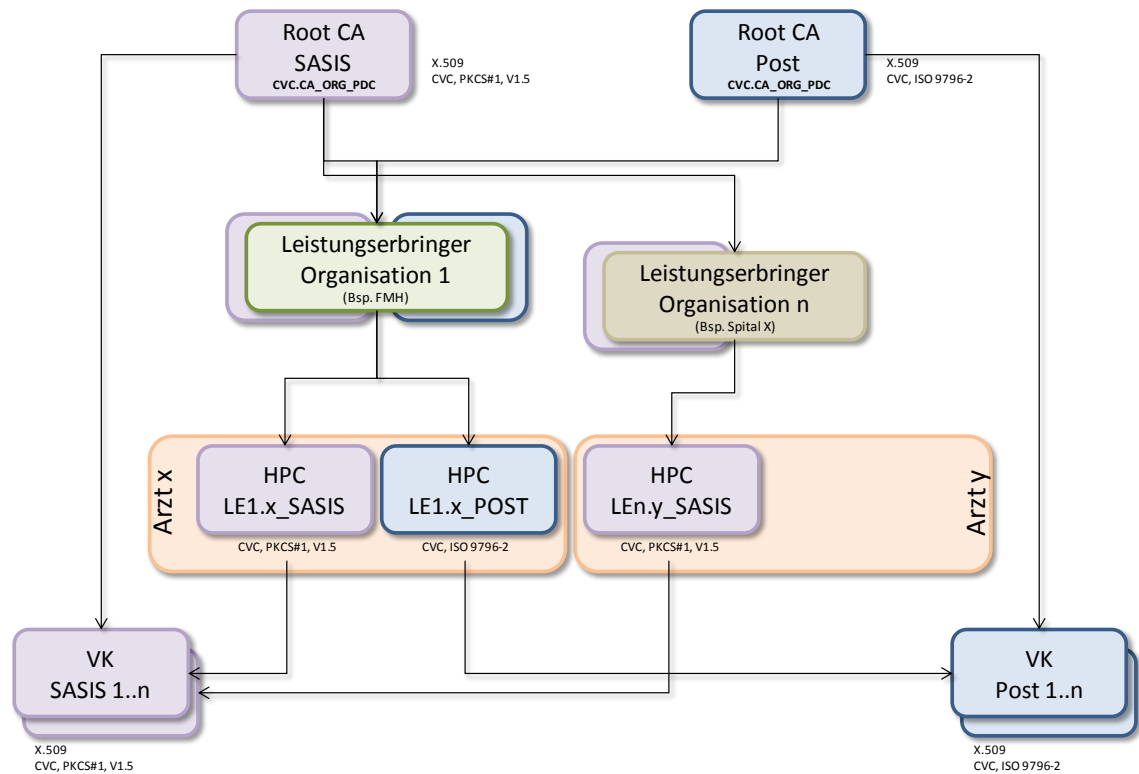
Typ	Beschreibung
Root Zertifikat	Stammzertifikat einer CA. Das Root Zertifikat ist die Basis für die Validierung digitaler Identitäten. Es kann in Form einer X.509- oder CV Struktur abgebildet werden ⁵ .
CVC	Digitale Identität in Form eines <i>Card Verifiable Certificates</i> . CV Zertifikate werden an Leistungsbezüger (Versichertenkarte) und Leistungserbringer (HPC) im Zusammenhang mit der <i>Card to Card Authentication</i> eingesetzt.
X.509	Digitale Identität in Form eines X.509 Zertifikats. X.509 Zertifikate werden an Leistungsbezüger (Versichertenkarte) und Leistungserbringer (HPC) im Zusammenhang mit der Authentisierung an Online-Systemen ⁶ eingesetzt.

[eCH64] definiert folgende CA Hierarchien und Abhängigkeiten im Zusammenhang mit der *Card to Card Authentication*:

⁴ Vergleichbar mit der SuisseID

⁵ Es kann davon ausgegangen werden, dass für CVC und X.509 Zertifikate jeweils eigenen Root CA Instanzen etabliert wurden.

⁶ Kantonale Modellversuche



Im Zusammenhang mit der *Card to Card Authentication* ergeben sich somit die folgenden Sachverhalte:

Komponente	Beschreibung		
VK	Das spezifische Root Zertifikat wird in die Versichertenkarte des jeweiligen Anbieters integriert. Es bildet die Grundlage für die Validierung des HPC Zertifikats im Zusammenhang mit der <i>Card to Card Authentication</i> .		
HPC	Die Root CA der jeweiligen Anbieter stellen Zertifikate für Leistungserbringer-Organisationen ⁷ aus. Diese wiederum stellen Zertifikate an spezifische Leistungserbringer ⁸ aus. <div style="border: 1px solid red; padding: 5px; margin-top: 5px;"> <table border="1"> <tr> <td style="background-color: #f8d7da;">Hinweis</td> <td>Aufgrund der oben abgebildeten CA Hierarchie ist ersichtlich, dass die Anzahl HPC Zertifikate, die für einen spezifischen Leistungserbringer ausgegeben werden müssen, identisch ist mit der Anzahl existierender Root CA (unabhängig von den jeweils eingesetzten Algorithmen und Schlüssellängen).</td> </tr> </table> </div>	Hinweis	Aufgrund der oben abgebildeten CA Hierarchie ist ersichtlich, dass die Anzahl HPC Zertifikate, die für einen spezifischen Leistungserbringer ausgegeben werden müssen, identisch ist mit der Anzahl existierender Root CA (unabhängig von den jeweils eingesetzten Algorithmen und Schlüssellängen).
Hinweis	Aufgrund der oben abgebildeten CA Hierarchie ist ersichtlich, dass die Anzahl HPC Zertifikate, die für einen spezifischen Leistungserbringer ausgegeben werden müssen, identisch ist mit der Anzahl existierender Root CA (unabhängig von den jeweils eingesetzten Algorithmen und Schlüssellängen).		

⁷ Beispiel: FMH, Spital X, etc.

⁸ Beispiel: Spezifischer Arzt oder Apotheker

5 Analyse eCH-0064 – Detailspezifikationen

5.1 Einleitung

In diesem Kapitel sind Abweichungen der [SPEC-POST] und der [SPEC-SASIS] gegenüber [eCH64] beschrieben. Untersucht wurden alle relevanten Punkte, welche direkt Einfluss auf die Implementierung und Nutzung der Versichertenkarte haben.

5.2 Übersicht über die Ergebnisse

In diesem Kapitel werden die Ergebnisse der Untersuchungen konsolidiert und grob bewertet. Aufgrund des Umfangs und der Komplexität des Systems und den beschränkten Ressourcen muss davon ausgegangen werden, dass nicht alle Abweichungen der [SPEC-POST] und der [SPEC-SASIS] gegenüber [eCH64] aufgedeckt werden konnten. Es wurden aber genügend Abweichungen gefunden, um eine allgemeingültige Aussage über die Interoperabilität der Versichertenkarten machen zu können.

Die Abweichungen der [SPEC-POST] und der [SPEC-SASIS] gegenüber [eCH64] haben die folgenden Ursachen:

- a) In [eCH64] ist kein Implementationsstandard, sondern ein Konzeptions-, Struktur- und Verfahrensstandard.
- b) Die [SPEC-POST] und die [SPEC-SASIS] wurden weitgehend autonom, mit unzureichender übergeordneter Koordination, erarbeitet.
- c) Verschiedene Punkte aus [eCH64] und den referenzierten ISO Normen und RFC Standards sind interpretierbar.
- d) Einzelne Vorgaben aus [eCH64], [VVK-EDI] sowie einer referenzierten ISO Norm⁹ sind fehlerhaft. Aufgrund der unzureichenden Koordinierung der Parteien führten die jeweiligen Fehler zu Abweichungen in den Spezifikationen oder in der Umsetzungen.

⁹ ISO 21549-5

Folgende Tabelle gibt eine Übersicht über die Erkenntnisse und gefundenen Abweichungen. Die einzelnen Spalten sind wie folgt zu lesen:

Fehler Spec	Fehler in den zugrundeliegenden Spezifikationen, die einen direkten Einfluss auf I/FI oder H/A/U haben. Aufgrund der unzureichenden Koordinierung der Parteien führten die Fehler in den zugrundeliegenden Spezifikationen zu Abweichungen in den jeweiligen Spezifikationen oder Umsetzungen.
I/FI	Abweichung der Spezifikation resp. der Umsetzung aufgrund einer Interpretation resp. einer fehlerhaften Implementation gegenüber [eCH64].
H/A/U	Abweichung der Spezifikation resp. der Umsetzung aufgrund dem Hinzufügen, Ändern oder Unterlassen von Elementen gegenüber [eCH64].

Pos	Analyse	[SPEC-POST]			Kommentar	[SPEC-SASIS]			Kommentar
		Fehler Spec	I/FI	H/A/U		Fehler Spec	I/FI	H/A/U	
1	A1								
2	A2								
3	A3		X		Fehlender PKCS#15 Pfad in EF.DIR	x ¹	X		Struktur "transparent" anstelle "linear variable"
4	A4	x ¹	X		BCD Kodierung von "Z" nicht definiert	x ¹		X	Kodierung nach ISO 8601:2004
5	A5	x ²	X		ASN.1 Kodierung fehlerhaft umgesetzt	x ²	X		ASN.1 Kodierung fehlerhaft umgesetzt
6	A6	x ³	X		Fehlerhafte Umsetzung der optionalen Felder				Keine optionalen Felder definiert, daher kein I/F
7	A7							X	Unterlassen von EF.LOG
8	A8				Siehe A18				Siehe A18
9	A9	x ⁴	X		EF.PK.CA_ORG_PDC als "Working" definiert		X		EF.PK.CA_ORG_PDC als "Working" definiert
10	A10								Siehe A15
11	A11			X	Unterlassen von EF.CVC.CA_ORG_HPC			X	Unterlassen von EF.CVC.CA_ORG_HPC
12	A12	x ⁵	X		Speicherkapazität der Karte	x ⁵	X		Speicherkapazität der Karte
13	A13		X		Kartenspezifische Eigenschaften		X		Kartenspezifische Eigenschaften
14	A14								
15	A15							X	Von [eCH64] abweichende CA Hierarchie
16	A16			X	Interpretation [VVK-EDI] / [eCH64]			X	Von [eCH64] abweichendes Zertifikatsformat
17	A17		X	X	PKCS#15, PIN2, etc.		X	X	PKCS#15, PIN2, etc.
18	A18	x ¹	X		Von [eCH64] abweichendes Zertifikatsformat	x ¹		X	Von [eCH64] abweichendes Zertifikatsformat
19								X	Von [eCH64] abweichendes Signatur-Padding
20								X	Von [eCH64] abweichender Hash-Algorithmus
21								X	Von [eCH64] abweichende Schlüssellänge
22	A19		X		Fehlerhafte Implementation		X		Abhängig von A18
	Total	6	10	3		5	7	9	

Legende Fehlerhafte Spezifikation

- ¹ Fehler in eCH64
- ² Fehler in ISO 21549-5
- ³ Fehler in VVK-EDI
- ⁴ Die Vorgaben von eCH64 sind nicht praxistauglich
- ⁵ Unklare Vorgaben in eCH64 / VVK-EDI bez. Umgang mit reduzierten der Datenkatalogen

Es ist schwierig, die einzelnen Abweichungen in Bezug auf den Einfluss auf das Gesamtsystem zu gewichten. Entscheidend ist, ob eine Abweichung über eine Middleware korrigiert werden kann oder nicht. Die praktischen Analysen haben gezeigt, dass die aktuellen Versichertenkarten unter Verwendung einer geeigneten Middleware interoperable zueinander sind [Expertise VK - Tech].

- Offensichtlich grosse Abweichungen von [eCH64] wie beispielsweise das Zertifikatsformat¹⁰ können unter Berücksichtigung der aktuellen Situation im

¹⁰ A18, [VK-SASIS]

Zusammenhang mit der Zertifikatshierarchie über eine geeignete Middleware zueinander interoperabel gemacht werden.

- Offensichtlich geringfügige Abweichungen von [eCH64] wie beispielsweise die fehlerhafte Kodierung von DF.NOT¹¹ im Zertifikat könnten aufgrund von Prozessen, welche ausschliesslich auf der Versichertenkarte ausgeführt werden, nicht über eine geeignete Middleware zueinander interoperabel gemacht werden¹².

Unabhängig vom Grad der Abweichungen muss eine Applikation die VK-POST und VK-SASIS unterschiedlich gemäss [SPEC-POST] resp. [SPEC-SASIS] ansteuern. Die Interoperabilität der unterschiedlichen Versichertenkarten muss über eine geeignete Middleware sichergestellt werden¹³.

5.2.1 Detailspezifikation der Schweizerischen Post

Die [SPEC-POST] hält sich grundsätzlich an die Vorgaben von [eCH64]. Die technischen Einzelheiten werden verständlich beschrieben. Teilweise wären ausführlichere Beschreibungen oder Kommentare angebracht.

- Die H/A/U spezifischen Abweichungen A11 und A17 fallen wenig ins Gewicht, da sie mit der [SPEC-SASIS] abgeglichen wurden und keinen negativen Einfluss auf das Gesamtsystem haben.
- Die H/A/U spezifische Abweichung A16 hat einen Einfluss auf verschiedene Prozesse im Zusammenhang mit dem PIN-Schutz der medizinischen Daten.
- Die I/FI spezifischen Abweichungen zu [eCH64] resp. zur [SPEC-SASIS] können über eine Middleware harmonisiert werden.
- Die I/FI spezifische Abweichung A18 hätte bei Identischer Umsetzung der VK-Post und VK-Sasis die Verwendung eines einheitlichen Leistungserbringerzertifikats gemäss [eCH64] verunmöglicht. Siehe Fussnote 12.

¹¹ A18, [VK-Post], siehe Hinweis

¹² Die Kodierung von DF.NOT bei der VK-Post hat in der aktuellen Situation im Zusammenhang mit der Zertifikatshierarchie keinen negativen Einfluss. Falls die VK-Post und VK-SASIS, mit Ausnahme der Kodierung von DF.NOT, identisch umgesetzt worden wären, hätte dies, analog der aktuellen Zertifikatsänderung bei der VK-SASIS, die Verwendung eines einheitlichen Leistungserbringerzertifikats verunmöglicht.

¹³ Siehe Hinweis in Kapitel 6

5.2.2 Detailspezifikation der SASIS

Die [SPEC-SASIS] hält sich weitgehend an die Vorgaben von [eCH64]. Die technischen Einzelheiten werden gut verständlich und ausführlich beschrieben.

- Die H/A/U spezifischen Abweichungen A11 und A17 fallen wenig ins Gewicht, da sie mit der [SPEC-POST] abgeglichen wurden und keinen negativen Einfluss auf das Gesamtsystem haben.
- Die H/A/U spezifische Abweichung A7 hat keinen negativen Einfluss auf das Gesamtsystem¹⁴.
- Die H/A/U spezifische Abweichung A15 hat einen Einfluss auf die Hierarchie der Public Key Infrastruktur im Zusammenhang mit Terminalauthentisierung auf Seiten der Applikation (HPC) und muss dort entsprechend gehandhabt werden. Die Abweichung hat keinen Einfluss auf die *Card to Card Authentication*.
- Die H/A/U spezifische Abweichung A18 hat einen Einfluss auf die Public Key Infrastruktur sowie auf verschiedene Prozesse im Zusammenhang mit der *Card to Card Authentication* der VK-SASIS. Die Abweichung hat keinen Einfluss auf die *Card to Card Authentication* der VK-POST, da die VK-Post mit eigenen Zertifikaten arbeitet.

Auffallend ist, dass die [SPEC-SASIS] sich in Kapitel 6 (Card to Card Authentifizierung und Autorisierung) im Abschnitt 6.1.1.1 (Normative Grundlagen) nicht auf [eCH64], sondern auf die technischen Richtlinien des BSI im Zusammenhang mit Extended Access Control (EAC) beziehen. Die CV Zertifikate wurden entsprechend der zuvor genannten Norm ausgegeben und halten sich nicht an die Vorgaben von [eCH64].

- Die I/FI spezifischen Abweichungen zu [eCH64] resp. zur [SPEC- POST] können über eine Middleware harmonisiert werden.

¹⁴ Siehe Hinweis in Kapitel 5.3.7

5.3 eCH-0064: Kapitel 3.4 Dateisystem

Im Folgenden sind die wesentlichen Unterschiede in Bezug auf das Dateisystem festgehalten. Es wurden nur die Dateien beschrieben, deren unterschiedliche Spezifikationen einen Einfluss auf das System der Versichertenkarte haben.

5.3.1 EF.ARR

A1.	EF.ARR
[eCH64]	Beschreibung
Kapitel 3.4.2	Im karteninternen File EF.ARR werden die Zugriffsregeln auf Ebene MF geregelt.
[SPEC-POST]	Beschreibung
	EF.ARR ist in der [SPEC-POST] nicht beschrieben. Die Zugriffsregeln werden kartenspezifisch intern abgebildet.
[SPEC-SASIS]	Beschreibung
	EF.ARR ist in der [SPEC-SASIS] nicht beschrieben. Die Zugriffsregeln werden kartenspezifisch intern abgebildet.

5.3.2 EF.ATR

A2.	EF.ATR
[eCH64]	Beschreibung
Kapitel 3.4.2	<p>Das File EF.ATR enthält Datenobjekte zur Identifizierung der Karte und ein Datenobjekt zur Anzeige der Grösse der Ein-/Ausgabe-Puffers. Die maximalen Längen der einzelnen APDU Kommandos unterscheiden sich bei den beiden Spezifikationen, sind aber Konform mit [eCH64].</p> <p>Bez. der problematischen Handhabung der Filestruktur wird auf Kapitel 5.3.3 verwiesen.</p>
[SPEC-POST]	Beschreibung
Kapitel 2.5	<p>Die Längen der einzelnen APDU Kommandos sind gemäss [SPEC-POST] mit 0x04AF (1'199 Bytes) angegeben. Die VK-POST unterstützt Extended Lc and Le Felder.</p> <p>Hex-Dump</p> <pre> EF.ATR 0x2f, 0x01 SELECT response: ResponseAPDU: 2 bytes, SW=9000 READ BINARY response: ResponseAPDU: 36 bytes, SW=9000 00000000 E0 10 02 02 04 AF 02 02 04 AF 02 02 04 AF 02 02 00000010 04 AF 66 0E 46 0C 04 44 45 47 2B 44 1B B4 02 00 ..f.F..DEG+D.... 00000020 0A 02 .. </pre>

Folgend ist die Auswertung des ATR aufgeführt.

TS = 0x3B	Direct Convention
T0 = 0xDB	Y(1): b1101, K: 11 (historical bytes)
TA(1) = 0x96	Fi=512, Di=32, 16 cycles/ETU (250000 bits/s at 4.00 MHz, 312500 bits/s for fMax=5 MHz)
TC(1) = 0xFF	Extra guard time: 255 (special value)
TD(1) = 0x81	Y(i+1) = b1000, Protocol T=1
----	----
TD(2) = 0x31	Y(i+1) = b0011, Protocol T=1
----	----
TA(3) = 0xFE	IFSC: 254
TB(3) = 0x45	Block Waiting Integer: 4 - Character Waiting Integer: 5
----	----
Historical bytes	80 67 04 1B B4 2A 00 0A 02 81 05
Category indicator byte: 0x80	(compact TLV data object) Tag: 6, Len: 7 (pre-issuing data) Data: 04 1B B4 2A 00 0A 02 Tag: 8, Len: 1 (status indicator) LCS (life card cycle): 5
TCK = 0x53	(correct checksum)

[SPEC-SASIS]

Beschreibung

Kapitel 4.3

Die **Längen** der einzelnen APDU Kommandos sind gemäss [SPEC-SASIS] mit 0x0400 (1'024 Bytes) angegeben. Die VK-SASIS unterstützt Extended Lc and Le Felder.

Hex-Dump

```
EF.ATR 0x2f, 0x01
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 45 bytes, SW=9000
00000000 E0 10 02 02 04 00 02 02 04 00 02 02 04 00 02 02 .....
00000010 04 00 66 17 46 15 02 49 6E 74 65 72 31 20 20 20 ..f.F..Interl
00000020 4D 54 43 4F 53 20 70 20 32 2E 31                      MTCOS p 2.1..
```

Folgend ist die Auswertung des ATR aufgeführt.

TS = 0x3B	Direct Convention
T0 = 0x9F	Y(1): b1001, K: 15 (historical bytes)
TA(1) = 0x13	Fi=372, Di=4, 93 cycles/ETU (43010 bits/s at 4.00 MHz, 53763 bits/s for fMax=5 MHz)
TD(1) = 0x81	Y(i+1) = b1000, Protocol T=1
----	----
TD(2) = 0xB1	Y(i+1) = b1011, Protocol T=1
----	----
TA(3) = 0x80	IFSC: 128
TB(3) = 0x37	Block Waiting Integer: 3 - Character Waiting Integer: 7
TD(3) = 0x1F	Y(i+1) = b0001, Protocol T=15
----	----
TA(4) = 0x03	Clock stop: not supported - Class accepted by the card: (3G) A 5V B 3V
----	----
Historical bytes	80 31 F8 69 4D 54 43 4F 53 70 02 01 02 81 07
Category indicator byte: 0x80	(compact TLV data object) Tag: 3, Len: 1 (card service data byte) Card service data byte: 248 - Application selection: by full DF name - Application selection: by partial DF name - BER-TLV data objects available in EF.DIR - BER-TLV data objects available in EF.ATR - EF.DIR and EF.ATR access services: by READ BINARY command - Card with MF Tag: 6, Len: 9 (pre-issuing data) Data: 4D 54 43 4F 53 70 02 01 02 Tag: 8, Len: 1 (status indicator) LCS (life card cycle): 7
TCK = 0x86	(correct checksum)

5.3.3 EF.DIR

A3.	EF.DIR
[eCH64]	Beschreibung
Kapitel 3.4.2	<p>Das File EF.DIR ist ein Verzeichnis für die in der Versichertenkarte vorhandenen Anwendungen.</p> <p>[eCH64] definiert fälschlicherweise die Struktur von EF.ATR als transparent und die Struktur von EF.DIR als linear variabel. Gemäss [ISO 7816-4] Kapitel 8.1.1.2.3 ist definiert, dass die Struktur der beiden Files EF.ATR und DIR entweder linear variabel oder transparent sein müssen.</p> <p>Die VK-Post und die VK-SASIS spezifizieren und implementieren den Typ der Filestruktur sowie den Dateninhalt unterschiedlich. Entsprechend ist der Zugriff auf die Files sowie die Auswertung der Daten unterschiedlich. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>
[SPEC-POST]	Beschreibung
Kapitel 2.4	<p>a) Die [SPEC-POST] spezifiziert die Struktur der beiden Files EF.ATR und EF.DIR gemäss [eCH64]¹⁵. [eCH64] folgt jedoch nicht der Vorgaben von [ISO 7816-4]. EF.DIR ist entsprechend als linear variable Struktur umgesetzt.</p> <p>b) EF.DIR beinhaltet lediglich die AID von DF.NOT resp. dem DF.PKCS#15 File, nicht aber den Pfad (FID) zu den entsprechenden DFs. Im Zusammenhang mit PKCS#15¹⁶ ist der Application Identifier (Tag 0x4F) und der Pfad (Tag 0x51) zwingend.</p> <p>Die entsprechenden FIDs müssen aus der [SPEC-POST] bezogen werden und statisch in der Middleware gespeichert werden.</p> <p>c) Record 3 ist in der [SPEC-POST] nicht definiert.</p> <p>Hex-Dump</p> <pre> EF.DIR 0x2f, 0x00 SELECT response: ResponseAPDU: 2 bytes, SW=9000 READ RECORD 1 response: ResponseAPDU: 12 bytes, SW=6282 00000000 61 08 4F 06 D7 56 83 21 05 00 a.o..V!.. READ RECORD 2 response: ResponseAPDU: 18 bytes, SW=6282 00000000 61 0E 4F 0C A0 00 00 00 63 50 4B 43 53 2D 31 35 a.o.....cPKCS-15 READ RECORD 3 response: ResponseAPDU: 17 bytes, SW=6282 00000000 61 0D 4F 0B F7 56 50 6F 73 74 4F 54 50 00 00 a.o..VPostOTP.. </pre>

¹⁵ EF.ATR: Transparent, EF.DIR: Linear variable

¹⁶ Kapitel 5.4.1.

[SPEC-SASIS]	Beschreibung
Kapitel 4.2	<p>a) Die [SPEC-SASIS] spezifiziert die Struktur der beiden Files EF.ATR und EF.DIR als transparent und folgt somit [ISO 7816-4], jedoch nicht [eCH64].</p> <p>b) Analog zur [SPEC-POST] beinhaltet EF.DIR die AIDs von DF.NOT und DF.PKCS#15. Zudem enthält das EF.DIR der [SPEC-SASIS] je ein Applikations Label sowie die Referenz (FID) zum entsprechenden DF. Eine Middleware kann auf der Basis der beiden letztgenannten Parameter dynamisch auf DF.NOT resp. DF.PKCS#15 zugreifen.</p> <p>Hex-Dump</p> <pre> EF.DIR, 0x2f, 0x00 SELECT response: ResponseAPDU: 2 bytes, SW=9000 READ BINARY response: ResponseAPDU: 86 bytes, SW=6282 00000000 61 19 4F 06 D7 56 83 21 05 00 50 09 45 6D 65 72 a.O..V.!..P.Eme 00000010 67 65 6E 63 79 51 04 3F 00 DF 01 61 1D 4F 0C A0 gencyQ?...a.O.. 00000020 00 00 00 63 50 4B 43 53 2D 31 35 50 07 50 4B 43 ...cPKCS-15P.PKC 00000030 53 2D 31 35 51 04 3F 00 DF 02 61 18 4F 0A F7 56 S-15Q?...a.O..V 00000040 83 21 05 4B 74 4D 56 00 50 04 4B 74 4D 56 51 04 .!.KtMV.P.KtMVQ. 00000050 3F 00 DF 03 ?... </pre>

5.3.4 EF.ICCSN

A4.	EF.ICCSN
[eCH64]	Beschreibung
Kapitel 3.4.2	<p>Das File EF.ICCSN entspricht der Kennnummer der Versichertenkarte.</p> <p>Record 1 und 2 sind in [eCH64] so spezifiziert, dass sie umgesetzt werden können. Problematisch ist Record 3. Hier legt [eCH64] fälschlicherweise fest, dass die generalTime in Record 3 als BCD kodiert werden muss¹⁷.</p> <p>Die VK-Post und die VK-SASIS spezifizieren und implementieren unterschiedliche Dateninhalte. Entsprechend ist die Auswertung der Daten unterschiedlich. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>

¹⁷ Der Buchstabe „Z“ kann in BCD grundsätzlich nicht kodiert werden.

[SPEC-POST]	Beschreibung
Kapitel 2.6	<p>a) Die GeneralizedTime ist in 8 Bytes kodiert und folgt den Vorgaben aus [eCH64]. Der Buchstabe „Z“ wurde als ASCII Wert den Datumsangaben angefügt.</p> <p>b) Die Struktur von EF.ICCSN wurde in [eCH64] als linear variable spezifiziert. Die [SPEC-POST] spezifiziert entgegen [eCH64] EF.ICCSN als linear fixed. Dies hat zur Folge, dass die Middleware die Füllbytes gemäss [SPEC-POST] entfernen muss.</p> <p>Hex-Dump</p> <pre> EF.ICCSN 0x2f, 0x05 SELECT response: ResponseAPDU: 2 bytes, SW=9000 READ RECORD 1 response: ResponseAPDU: 14 bytes, SW=6282 00000000 5A 0A 80 75 69 99 99 99 99 99 99 94 Z..ui..... READ RECORD 2 response: ResponseAPDU: 14 bytes, SW=6282 00000000 69 99 99 99 99 99 99 94 00 00 00 00 i..... READ RECORD 3 response: ResponseAPDU: 14 bytes, SW=6282 00000000 20 10 01 12 10 13 42 5A 00 00 00 00 BZ.... </pre>
[SPEC-SASIS]	Beschreibung
Kapitel 4.4	<p>Die GeneralizedTime ist nach ISO 8601:2004 kodiert und folgt nicht den Vorgaben aus [eCH64].</p> <p>Hex-Dump</p> <pre> EF.ICCSN 0x2f, 0x05 SELECT response: ResponseAPDU: 2 bytes, SW=9000 READ RECORD 1 response: ResponseAPDU: 14 bytes, SW=6282 00000000 5A 0A 80 75 60 12 34 00 00 00 20 85 Z..u`.4.... READ RECORD 2 response: ResponseAPDU: 10 bytes, SW=6282 00000000 39 39 39 39 39 30 30 31 99999001 READ RECORD 3 response: ResponseAPDU: 15 bytes, SW=6282 00000000 32 30 30 39 30 31 30 31 30 30 30 30 5A 200901010000z </pre>

5.3.5 EF.ID

A5.	EF.ID
[eCH64]	Beschreibung
Kapitel 3.4.2	<p>Das File EF.ID enthält die Identifikationsdaten des Karteninhabers nach Anhang 1 der [VVK-EDI].</p> <p>[VVK-EDI] verweist bezüglich Kodierung der Identifikationsdaten auf [ISO 21549-5:2006] und auf [ISO 21549-6:2006]. Problematisch hierbei ist, dass die ASN.1 Kodierung in [ISO 21549-5:2006], Tabelle 1, falsch spezifiziert wurde. Ein generischer ASN.1 Parser wäre nicht in der Lage, die entsprechenden Identifikationsdaten ohne zusätzliche Logik auslesen zu können.</p> <p>Korrekte Kodierung der Identifikationsdaten</p> <p>ASN.1 Struktur</p> <pre> 0000 65 4C: [APPLICATION 5] { 0002 A0 27: [0] { 0004 A1 A: [1] { 0006 81 8: [1] 'Eberhard' : : 0010 A2 19: [2] { 0012 30 17: SEQUENCE { 0014 81 4: [1] 'Rene' 001A 81 5: [1] 'Guido' 0021 81 8: [1] 'keyon AG'¹⁸ : : : 002B 82 8: [2] '19900101' 0035 83 14: [3] 'cardHolderIdentifier' 004B 84 1: [4] : : 01 : : } } </pre> <p>Kodierung der Identifikationsdaten nach [ISO 21549-5:2006]</p> <p>ASN.1 Struktur</p> <pre> 0000 65 4C: [APPLICATION 5] { 0002 80 27: [0] : A1 0A 81 08 45 62 65 72 68 61 72 64 A2 19 30 17 : 81 04 52 65 6E 65 81 05 47 75 69 64 6F 81 08 6B : 65 79 6F 6E 20 41 47 002B 82 8: [2] '19900101' 0035 83 14: [3] 'cardHolderIdentifier' 004B 84 1: [4] : : 01 : : } </pre> <p>Aus obiger Struktur ist ersichtlich, dass die einzelnen Identifikationsdaten (Name und Vornamen) nicht ohne zusätzliche Logik ausgelesen werden können. Die Spezifikation nach [ISO 21549-5:2006] ist aus Sicht der Autoren dieser Expertise fehlerhaft.</p> <p>Die VK-Post und die VK-SASIS spezifizieren und implementieren unterschiedliche Dateninhalte. Beide halten sich bez. Kodierung der Daten nicht an [ISO 21549-5:2006]. Entsprechend ist die Auswertung der Daten unterschiedlich. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>

¹⁸ keyon AG wurde als zusätzlicher Vorname der Struktur hinzugefügt.

[SPEC-POST]	Beschreibung
Kapitel 2.4	<p>Die [SPEC-POST] setzt die Vorgaben von [eCH64] um und orientiert sich an [ISO 21549-5:2006], hat jedoch die Kodierung der Daten falsch implementiert. Ein generischer ASN.1 Parser ist ohne zusätzliche Logik nicht in der Lage, die Daten korrekt auszulesen.</p> <p>Hex-Dump</p> <pre>EF.ID 0x2f, 0x06 SELECT response: ResponseAPDU: 2 bytes, SW=9000 READ BINARY response: ResponseAPDU: 62 bytes, SW=6282 00000000 65 3A 80 1C 30 1A 30 0C 0C 0A 48 61 6E 73 2D 50 e:..0.0...Hans-P 00000010 65 74 65 72 30 0A 30 08 0C 06 4D 75 73 74 65 72 eter0.0...Muster 00000020 82 08 31 39 36 35 30 37 32 32 83 0D 37 35 36 31 ..19650722..7561 00000030 32 33 34 35 36 37 38 39 30 84 01 01 234567890...</pre> <p>ASN.1 Struktur</p> <pre>0000 65 3A: [APPLICATION 5] { 0002 80 1C: [0] : 30 1A 30 0C 0C 0A 48 61 6E 73 2D 50 65 74 65 72 : 30 0A 30 08 0C 06 4D 75 73 74 65 72 0020 82 8: [2] '19650722' 002A 83 D: [3] '7561234567890' 0039 84 1: [4] : 01 : }</pre>
[SPEC-SASIS]	Beschreibung
Kapitel 4.4	<p>Die [SPEC-SASIS] legt eine eigene Kodierung für die Daten aus EF.ID fest, welche sich grundsätzlich an [ISO 21549-5:2006] orientiert. Im Gegensatz zu [ISO 21549-5:2006] trennt die [SPEC-SASIS] den Nachnamen mit den Vornamen nicht durch einzelne Felder sondern durch ein Komma gefolgt von einem Leerzeichen.</p> <p>Hex-Dump</p> <pre>EF.ID 0x2f, 0x06 SELECT response: ResponseAPDU: 2 bytes, SW=9000 READ BINARY response: ResponseAPDU: 86 bytes, SW=9000 00000000 65 2B 80 0D 53 75 74 65 72 2C 20 4D 61 72 6B 75 e+..Suter, Marku 00000010 73 82 08 32 30 31 30 30 31 30 31 83 0D 37 35 36 s..20100101..756 00000020 39 39 39 39 39 39 37 39 31 31 84 01 01 00 00 00 9999997911..... 00000030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00000040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00000050 00 00 00 00</pre> <p>ASN.1 Struktur</p> <pre>0000 65 2B: [APPLICATION 5] { 0002 80 D: [0] 'Suter, Markus' 0011 82 8: [2] '20100101' 001B 83 D: [3] '7569999997911' 002A 84 1: [4] : 01 : }</pre>

5.3.6 EF.AD

A6.	EF.AD
[eCH64]	Beschreibung
Kapitel 3.4.2	<p>Das File EF.AD enthält die administrativen Daten des Karteninhabers nach Anhang 1 der [VVK-EDI].</p> <p>[VVK-EDI] Kapitel 2.2.6.1 ist bezüglich der Definition der optionalen Parameter fehlerhaft. Die angegebenen Tags können nicht ohne zusätzliche Informationen BER kodiert werden. Es muss daher davon ausgegangen werden, dass diesbezüglich eigene Interpretationen in die Detailspezifikationen einfließen werden. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>
[SPEC-POST]	Beschreibung
Kapitel 2.15	<p>Die [SPEC-POST] setzt die Vorgaben von [eCH64] korrekt um und definiert neben den zwingenden Feldern auch verschiedene optionale Felder (siehe Kapitel 10.2.4).</p> <p>ASN.1 Struktur</p> <pre> 0000 65 34: [APPLICATION 5] { 0002 90 2: [16] : 43 48 0006 91 7: [17] 'Helsana' 000F 92 5: [18] '01562' 0016 93 14: [19] '80756999999999999994' 002C 94 8: [20] '20130331' : } </pre> <p>Die Spezifikation der optionalen Felder ist aber aufgrund der eingangs erwähnten Fehler von [VVK-EDI] falsch¹⁹. Ein generischer ASN.1 Parser ist ohne zusätzliche Logik nicht in der Lage, die Daten korrekt auszulesen.</p>
[SPEC-SASIS]	Beschreibung
Kapitel 4.9	<p>Die [SPEC-SASIS] setzt die Vorgaben von [eCH64] korrekt um, definiert aber keine optionalen Felder (siehe Kapitel 10.2.4).</p> <p>ASN.1 Struktur</p> <pre> 0000 65 31: [APPLICATION 5] { 0002 90 2: [16] : 43 48 0006 91 4: [17] 'TEST' 000C 92 5: [18] '01234' 0013 93 14: [19] '80756012340000002085' 0029 94 8: [20] '20140630' : } </pre>

¹⁹ TAG 73 (National Extension) fehlt. TAG 09 ist nicht BER konform.

5.3.7 EF.LOG

A7.	EF.LOG
[eCH64]	Beschreibung
Kapitel 3.4.2	Das File EF.LOG ist eine anwendungsspezifische Protokolldatei mit dem Ziel der Behebung eines Fehlers resp. der Wiederherstellung eines kartenspezifischen Zustandes im Falle von technischen oder applikatorischen Problemen. Dies setzt jedoch voraus, dass die Middleware resp. die Applikation kartenspezifische Details kennt und weiss, wie diese anzusteuern sind ²⁰ .
[SPEC-POST]	Beschreibung
Kapitel 2.7	Die [SPEC-POST] beschreibt die Struktur des Files EF.LOG, jedoch werden keine Einzelheiten offengelegt, welche Behebung eines Fehlers resp. die Wiederherstellung eines kartenspezifischen Zustandes im Falle von technischen oder applikatorischen Problemen erlauben würde.
[SPEC-SASIS]	Beschreibung
	Die [SPEC-SASIS] definiert kein File EF.LOG.

5.3.8 EF.CVC.PDC

A8.	EF.CVC.PDC
[eCH64]	Beschreibung
Kapitel 3.4.2	Das File EF.CVC.PDC enthält das CV Zertifikat der Versichertenkarte. Die VK-Post und die VK-SASIS spezifizieren und implementieren unterschiedliche Zertifikatsformate. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.
[SPEC-POST]	Beschreibung
Kapitel 2.9	Die [SPEC-POST] definiert grundsätzlich das CV Zertifikat gemäss [eCH64]. Siehe Kapitel 5.7.1.
[SPEC-SASIS]	Beschreibung
Kapitel 4.11	Die [SPEC-SAIS] definiert ein vom [eCH64] abweichendes Zertifikatsformat. Siehe Kapitel 5.7.1.

²⁰ Praxisfremder Mechanismus

5.3.9 EF.PK.CA_ORG_PDC

A9.	EF.PK.CA_ORG_PDC
[eCH64]	Beschreibung
Kapitel 3.4.2	<p>Das File EF.PK.CA_ORG_PDC enthält den öffentlichen Schlüssel der Root CA²¹.</p> <p>Die VK-Post und die VK-SASIS spezifizieren und implementieren unterschiedliche Signaturformate. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>
[SPEC-POST]	Beschreibung
Kapitel 2.8	<p>Die [SPEC-POST] definiert entgegen [eCH64] das File EF.PK.CA_ORG_PDC als „Working“ und ist somit durch eine Middleware auslesbar. Diese Erweiterung der [eCH64] Spezifikation wurde aufgrund der Eigenschaften der [ISO9796-2] basierten CV Zertifikate vorgenommen und ermöglicht es so einer Middleware, CV Zertifikatsketten ohne zusätzliche Parameter²² von aussen prüfen zu können. Die Schlüssellänge stimmt mit den Vorgaben von [eCH64] überein.</p> <p>Die [SPEC-POST] sollte noch die detaillierte Kodierung des öffentlichen Schlüssels beschreiben.</p> <p>Hex-Dump</p> <pre> EF.PuK.CA_ROOT_VK 0x0e, 0x02 SELECT response: ResponseAPDU: 2 bytes, SW=9000 READ BINARY response: ResponseAPDU: 142 bytes, SW=9000 00000000 81 82 00 80 A4 4D 1D 1C A8 8A 36 DE 97 BF 12 64M....6....d 00000010 1A CE 99 07 DB 18 09 33 98 36 E3 3D 86 81 2A B03.6.=.*. 00000020 73 6B BA E7 4E FC A0 15 AA 76 D9 7E 11 CB 9A 71 sk..N....v...~...q 00000030 47 0C 91 26 A2 2E E1 B0 8F 79 13 4B A2 83 39 81 G..&.....y.K..9. 00000040 C1 C4 77 E1 48 9D 4E 43 00 38 D4 B2 FD 35 79 45 ..w.H.NC.8...5yE 00000050 43 23 9C FO FA E4 57 D6 45 0C 00 28 A0 B0 9F D5 C#...W.E..(.... 00000060 18 BD 14 ED 5F 78 B6 26 7B 2B FC 4F 14 F8 A5 6Bx.&{+.O...k 00000070 5B 82 70 E6 5D F7 9B 05 A8 F9 AC D9 07 6C AF 64 [.p.].....l.d 00000080 B5 07 6A 7F 82 82 00 04 00 01 00 01j..... </pre> <p>Modulus n Exponent e</p>
[SPEC-SASIS]	Beschreibung
Kapitel 4.11	<p>Die [SPEC-SASIS] definiert entgegen [eCH64] das File EF.PK.CA_ORG_PDC als „Working“ und ist somit durch eine Middleware auslesbar. Diese Erweiterung der [eCH64] Spezifikation wäre aufgrund der Eigenschaften der [SPEC-SASIS] basierten CV Zertifikate nicht notwendig, da der öffentliche Schlüssel vollständig auf dem CV Zertifikat ausgelesen werden kann. Die Schlüssellänge weicht von den Vorgaben gemäss</p>

²¹ CVC.CA_ORG_PDC

²² Öffentlicher Schlüssel



	<p>[eCH64] ab Siehe Kapitel 5.7.1.</p> <p>Die [SPEC-SASIS] sollte noch die detaillierte Kodierung des öffentlichen Schlüssels beschreiben.</p> <p>Hex-Dump</p> <pre> EF.PuK.CA_ROOT_VK²³ 0x00, 0x1c SELECT response: ResponseAPDU: 2 bytes, SW=9000 READ RECORD response: ResponseAPDU: 301 bytes, SW=9000 00000000 10 43 48 52 56 4B 60 32 30 30 39 30 30 31 30 30 .CHRVK`200900100 00000010 31 00 00 01 00 00 09 01 02 00 03 01 07 01 02 00 1..... 00000020 03 56 55 01 00 C4 40 0A F4 CE 1B 21 20 FE 3F C6 .VU...@...! ??. 00000030 93 CF 81 4A 73 F4 3B 6D 70 DE 29 3D 56 9F B8 32 ...Js.;mp.)=V..2 00000040 3F A8 28 31 2D 61 23 6E 9F 67 FC 35 87 28 21 F3 ?. (1-a#n.g.5. (!. 00000050 79 B1 63 C1 63 7F 1D 42 98 77 99 26 6D 6D 4F D1 y.c.c..B.w.&mmO. 00000060 57 EE 84 3C 65 3F 6F 65 6A D7 51 7C D7 BB 6E 8F W...<e?oej.Q ..n. 00000070 E8 AA D6 5F EB 41 B9 BF 55 60 52 34 9C 7B 8C 18 ...A..U`R4.{.. 00000080 9E 48 E4 36 AF DB D4 81 05 FB CA EB 67 D9 52 E6 .H.6.....g.R. 00000090 81 A1 E6 DD 5A B3 55 B2 D0 35 BE 39 79 33 92 84 ...Z..U..5.9y3.. 000000A0 8D 39 E7 BF 2F D9 03 53 DC 46 62 80 4E 62 55 77 .9../..S.Fb.NbUw 000000B0 38 03 97 28 C1 D7 3D B1 4D 23 38 E5 59 4D 50 61 8..(..M#8.YMPa 000000C0 D2 56 00 36 86 5E B4 04 EF 9A 78 13 49 23 48 22 .V.6.^...x.I#H" 000000D0 31 93 4F 4A 5E E7 21 D9 19 0F CB 4E 3D E4 45 23 1.OJ^! ...N=.E# 000000E0 3F 93 87 3F F2 F9 48 3D 27 13 FB AB 80 34 E4 BC ?..?..H=#....4.. 000000F0 5C 2D CF E7 ED 60 3A D1 FC 1F E5 C4 9C 8E 82 00 \.....:..... 00000100 7F 6B 6F 8D 98 9B 9B B9 CF 10 33 A5 CF 5D 78 1C .ko.....3..]x. 00000110 41 97 8B 20 CB B1 D9 F3 ED 2F 13 56 71 56 9A 0A A../.VqV.. 00000120 9E 61 61 4B F7 00 01 00 01 FA 46 ..aaK.....F </pre> <p>Modulus n</p> <p>Exponent e</p>
--	---

5.3.10 EF.CVC.CA_ORG_PDC

A10.	EF.CVC.CA_ORG_PDC
[eCH64]	Beschreibung
Kapitel 3.4.2	<p>Das File EF.CVC.CA_ORG_PDC enthält das CV Root Zertifikat.</p> <p>Die VK-Post und die VK-SASIS spezifizieren und implementieren unterschiedliche Zertifikatsformate. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>
[SPEC-POST]	Beschreibung
Kapitel 2.8	<p>Die [SPEC-POST] definiert grundsätzlich das CV Zertifikat gemäss [eCH64]. Siehe Kapitel 5.7.1.</p>

²³ Öffentlicher Schlüssel von EF.CVC.CA_ROOT_VK

[SPEC-SASIS]	Beschreibung
Kapitel 4.11	<p>Die [SPEC-SASIS] definiert ein vom [eCH64] abweichendes Zertifikatsformat. Siehe Kapitel 5.7.1.</p> <p>Im Weiteren speichert das File EF.CVC.CA_ORG_PDC nicht das wie in [eCH64] definierte Root Zertifikat sondern das Herausgeber Zertifikat der Versicherer (Sub CA). Das eigentliche Root Zertifikat wird im File EF.CVC.CA_ROOT_VK gespeichert.</p>

5.3.11 EF.CVC.CA_ORG_HPC

A11.	EF.CVC.CA_ORG_HPC
[eCH64]	Beschreibung
Kapitel 3.4.2	<p>Im File EF.CVC.CA_ORG_HPC wird die HPC im Zusammenhang mit der <i>Card to Card Authentication</i> zwischengespeichert.</p> <p>Die VK-Post und die VK-SASIS spezifizieren und implementieren diesbezüglich die <i>Card to Card Authentication</i> abweichend von [eCH64]. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>
[SPEC-POST]	Beschreibung
	Die [SPEC-Post] definiert kein File EF.CVC.CA_ORG_HPC.
[SPEC-SASIS]	Beschreibung
	Die [SPEC-SASIS] definiert kein File EF.CVC.CA_ORG_HPC.

5.3.12 Verwaltung der Notfalldaten

A12. Verwaltung der Notfalldaten

[VVK-EDI] und [eCH64] legen technisch fest, wie spezifische Notfalldaten gespeichert werden müssen. Die folgenden Feststellungen gelten allgemein für alle Notfalldaten.

[eCH64]²⁴ fordert eine minimale Speicherkapazität von 32 KByte. Die in [VVK-EDI] spezifizierten Datenkataloge sind jedoch umfassender und können nicht in den minimal geforderten 32 KByte gespeichert werden. [eCH64] unterlässt es aber, genaue Vorgaben zu machen, wie mit allfällig reduzierten Speicherkapazitäten umzugehen ist.

5.3.12.1 Verwaltung der Records

[VVK-EDI] Kapitel 2.3 sowie [eCH64] definieren die Elemente im Zusammenhang Allergien oder Überempfindlichkeiten auf Medikamenten wie folgt:

Rekord [1..25]	Soforttypreaktionen, [VVK-EDI], Kapitel 2.3.1
Rekord [26..50]	Spättypreaktionen, [VVK-EDI], Kapitel 2.3.2

Eine Applikation, welche Daten im Zusammenhang mit Soforttypreaktionen speichern oder auslesen möchte, würde demnach auf Record 1 bis Record 25 zugreifen wollen. Analog würde die Applikation auf Record 26 bis Record 50 zugreifen wollen, wenn sie Daten im Zusammenhang mit Spättypreaktionen speichern oder auslesen wollte.

Vergleicht man die Spezifikationen im Zusammenhang mit EF.ALLG (siehe Kapitel 10.2.12) stellt man fest, dass die [SPEC-POST] total nur 6 Records und die [SPEC-SASIS] total nur 12 Records definiert. Aus den Spezifikationen geht nicht hervor, welche Records für Soforttypreaktionen und welche Records für Spättypreaktionen vorgesehen sind. Die einzelnen Elemente können einzig über die in [VVK-EDI] spezifizierten Tags, unabhängig von den Records, verwaltet werden.

Hinweis	<p>Auf Ebene APDU ohne zusätzliche Spezifikationen ist es nicht möglich, die Notfalldaten auf der Versichertenkarte zu verwalten.</p> <p>Die [SPEC-POST] und die [SPEC-SASIS] spezifizieren die Filestrukturen im Zusammenhang mit den Notfalldaten unterschiedlich. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden</p>
---------	---

²⁴ Kapitel 3.2.3

5.4 eCH-0064: Kapitel 3.6.4.1, Card to Card-Authentication and Authorization

Im Folgenden sind die Wesentlichen Unterschiede in Bezug auf die *Card to Card-Authentication and Authorization* festgehalten. Es wurden Sachverhalte beschrieben, deren unterschiedliche Spezifikationen einen Einfluss auf das System der Versichertenkarte haben.

5.4.1 Card to Card Authentication and Authorization

A13. Card to Card Authentication and Authorization

Die *Card to Card Authentication and Authorization* wird für die Freischaltung des Schreib-/Lesezugriffs auf die Notfalldaten verwendet. Dabei weist sich der Inhaber einer HPC über sein auf der HPC gespeichertes Zertifikat gegenüber der Versichertenkarte aus und erhält die seiner Leistungserbringergruppe entsprechenden Zugriffsrechte.

[SPEC-POST] und [SPEC-SASIS]²⁵ halten sich dabei beide grundsätzlich an das in [eCH64] Kapitel 3.6.4 spezifizierte Verfahren, zeigen aber individuelle Abweichungen in den im Verfahren verwendeten Daten und Datenformaten.

Auf APDU-Level zeigen die Implementierungen der VK-Post und der VK-SASIS Unterschiede. Diese sind aber wenig relevant, da [eCH64] auf dieser Stufe keine Vorgaben macht. Die Interoperabilität muss entsprechend durch eine Middleware sichergestellt werden.

Nachfolgende Tabelle beschreibt den Ablauf der *Card to Card Authentication* nach [eCH64] und listet die Abweichungen von [SPEC-POST] und [SPEC-SASIS] zu [eCH64] auf.

Bezeichnung	Abweichung gegenüber [eCH64]		
Farbcode	Keine Abweichung	Geringfügige Abweichung	Abweichung

#	[eCH64]	[SPEC-Post]	[SPEC-SASIS]
1	Selektieren und Auslesen des Chipcard Identifier Files	Liest die Kartenreferenznummer anstelle der ICCSN _B ²⁶	
2	Übertragung des Chipcard Identifier Files der Versichertenkarte		
3	Selektieren und Auslesen des CV Personenzertifikats des Versicherten		

²⁵ Verweis auf Technical Guideline TR-03110

²⁶ Record 2 anstelle von Record 1

4	Übertragung des CV Personenzertifikats des Versicherten		¹ Die VK-SASIS verwendet ein geändertes CV Zertifikatsformat, geändertes Signaturformat und längere RSA-Schlüssel
5	Setzen des öffentlichen Root-Schlüssels der Versichererorganisation des Versicherten		
6	Prüfen des CV Personenzertifikats des Versicherten		siehe ¹
6a	Überprüfung der Chipkartenseriennummer durch Vergleich der Nummern auf der Karte und im CV Personenzertifikat		
7	Selektieren und Auslesen des Chipcard Identifier Files der Leistungserbringerkarte		
8	Übertragung des Chipcard Identifier Files der Leistungserbringerkarte		
9	Selektieren und Auslesen des CV Leistungserbringerzertifikats		
10	Übertragung des CV Leistungserbringerzertifikat		siehe ¹
11	Auslesen und Speichern der notwendigen Leistungserbringer-Attribute aus dem CV Leistungserbringerzertifikat		
12	Anhand der Leistungserbringer-Attribute wird das entsprechende Leistungserbringerorganisationszertifikat selektiert.		siehe ¹
13	Der Container für das ausgewählte Leistungserbringerorganisationszertifikat wird ausgewählt und soll beschrieben werden.		
14	Übertragung und Speicherung des Leistungserbringerorganisationszertifikats	Speichert das Zertifikat nicht in EF.CVC_CA_ORG_HPC	Speichert das Zertifikat nicht in EF.CVC_CA_ORG_HPC
15	Das CV Leistungserbringerzertifikat wird der Versichertenkarte zur Verifikation bereitgestellt		siehe ¹
16	Setzen des Root-Schlüssels der Versichererorganisation des Versicherten		Die VK-SASIS verwendet aufgrund einer geänderten CA-Hierarchie einen anderen Schlüssel als den von CA_ORG_PDC
17	Prüfen des CV Zertifikats der entsprechenden Herausgeberorganisation		siehe ¹
18	Setzen des öffentlichen CA-Schlüssels der entsprechenden Leistungserbringerorganisation		
19	Prüfen des CV Leistungserbringerzertifikats		siehe ¹
20	Zwischenspeicherung des Karteninhaberautorisierungsmerkmalswertes CHA _n , welches im Leistungserbringerzertifikat eingebunden ist		

21	Zufallszahl erzeugen		
22	Zufallszahl übermitteln		
23	Zufallszahl signieren	Verwendet ICCSN _B als Hash-Input	² Verwendet ein anderes Signaturformat und längere RSA-Schlüssel
24	Signierte Zufallszahl übermitteln		
25	Verifikation der Signatur der signierten Zufallszahl		siehe ²
26	Der Karteninhaberautorisierungsmerkmalswert CHA _n wird freigeschaltet		
27	Datenaustausch zwischen dem Lesegerät/Anwendungsmodul und der Versichertenkarte		

5.4.2 Terminalauthentisierung

A14.	Terminalauthentisierung		
[eCH64]	Beschreibung		
Kapitel 3.6.4.1 Schritt Nr.5	<p>Mit der Terminalauthentisierung weist sich die Versichertenkarte gegenüber dem Lesegerät aus.</p> <p>Die optionale Terminalauthentisierung der Versichertenkarte gegenüber dem Lesegerät erfolgt über den Vergleich der ICCSN aus der Datei EF.ICCSN mit der ICCSN aus dem CV Personenzertifikat der Versichertenkarte.</p> <p>Die VK-Post und die VK-SASIS spezifizieren diesbezüglich unterschiedliche Verfahren. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>		
[SPEC-POST]	Beschreibung		
Kapitel 3.4.1 APDU-Seq. 1	<p>Die [SPEC-POST] ist in sich nicht konsistent und weicht von [eCH64] ab. Sie verwendet anstelle des in [eCH64] definierten ICCSN_B (10 Byte) die Kartenreferenznummer (8 Byte). Mit der Kartenreferenznummer kann keine „Authentisierung“ der Versichertenkarte durchgeführt werden.</p> <p>Die [SPEC-POST] muss angepasst werden²⁷.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="background-color: #e0f0ff; width: 15%;">Details</td> <td> <p>[eCH64] definiert dass in Schritt 1 der Inhalt von EF.ICCSN_B ausgelesen werden soll.</p> <p>Die in EF.ICCSN_B enthaltene ICCSN wird im optionalen Schritt 6a für den Vergleich mit der in CVC.PDC enthaltenen ICCSN benötigt.</p> <p>In den in [SPEC-POST], Kapitel 3.4.1 aufgelisteten APDU-Sequenzen wird Record 1 anstelle von Record 2 aus der Datei EF.ICCSN gelesen. D.h. es werden anstelle 10 Byte langen ICCSN die 8 Byte lange Referenznummer der Karte ausgelesen.</p> <p>In Kapitel 3.2 der [SPEC-POST] wird die ICCSN als Input der Hash-Funktion aufgelistet. Allerdings wiederum nur mit 8 Byte Länge anstelle der 10 Byte der ICCSN, was auf eine Verwendung der Referenznummer anstelle der ICCSN hindeutet.</p> </td> </tr> </table>	Details	<p>[eCH64] definiert dass in Schritt 1 der Inhalt von EF.ICCSN_B ausgelesen werden soll.</p> <p>Die in EF.ICCSN_B enthaltene ICCSN wird im optionalen Schritt 6a für den Vergleich mit der in CVC.PDC enthaltenen ICCSN benötigt.</p> <p>In den in [SPEC-POST], Kapitel 3.4.1 aufgelisteten APDU-Sequenzen wird Record 1 anstelle von Record 2 aus der Datei EF.ICCSN gelesen. D.h. es werden anstelle 10 Byte langen ICCSN die 8 Byte lange Referenznummer der Karte ausgelesen.</p> <p>In Kapitel 3.2 der [SPEC-POST] wird die ICCSN als Input der Hash-Funktion aufgelistet. Allerdings wiederum nur mit 8 Byte Länge anstelle der 10 Byte der ICCSN, was auf eine Verwendung der Referenznummer anstelle der ICCSN hindeutet.</p>
Details	<p>[eCH64] definiert dass in Schritt 1 der Inhalt von EF.ICCSN_B ausgelesen werden soll.</p> <p>Die in EF.ICCSN_B enthaltene ICCSN wird im optionalen Schritt 6a für den Vergleich mit der in CVC.PDC enthaltenen ICCSN benötigt.</p> <p>In den in [SPEC-POST], Kapitel 3.4.1 aufgelisteten APDU-Sequenzen wird Record 1 anstelle von Record 2 aus der Datei EF.ICCSN gelesen. D.h. es werden anstelle 10 Byte langen ICCSN die 8 Byte lange Referenznummer der Karte ausgelesen.</p> <p>In Kapitel 3.2 der [SPEC-POST] wird die ICCSN als Input der Hash-Funktion aufgelistet. Allerdings wiederum nur mit 8 Byte Länge anstelle der 10 Byte der ICCSN, was auf eine Verwendung der Referenznummer anstelle der ICCSN hindeutet.</p>		
[SPEC-SASIS]	Beschreibung		
Kapitel 6.2	Die Terminalauthentisierung wurde nach [eCH64] spezifiziert.		

²⁷ Gemäss Aussage der Post ist die Spezifikation falsch, die Umsetzung der Versichertenkarte jedoch korrekt. Diese Aussage konnte im Rahmen der Expertise nicht mehr überprüft werden.

5.4.3 CA-Hierarchie

A15.	CA-Hierarchie
[eCH64]	Beschreibung
Kapitel 6.1.2	<p>Die CA-Hierarchie, wie sie in [eCH64] definiert wurde, ist in Kapitel 4.2 dieses Dokuments beschrieben.</p> <p>Die VK-Post und die VK-SASIS spezifizieren diesbezüglich unterschiedliche Hierarchien. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>
[SPEC-POST]	Beschreibung
	Die [SPEC-POST] folgt den Vorgaben aus [eCH64].
[SPEC-SASIS]	Beschreibung
Kapitel 6.1.2.1 Kapitel 6.1.2.2	<p>Die [SPEC-SASIS] führt eine in [eCH64] nicht definierte Root-CA ein, welche Issuing CAs der Leistungserbringerorganisationen wie auch der Versichererorganisationen subordiniert.</p>
	<p>Die CA-Zertifikate der Versichererorganisationen und Leistungserbringerorganisationen werden gemäss [SPEC-SASIS] nicht durch die in [eCH64] Kapitel 6.1.2 spezifizierten CAs ausgestellt und signiert.</p> <p>Die Einführung der neuen Root-CA hat einen Einfluss auf die Zertifikatsvalidierungsprozesse im Bereich der Versichertenkarte. Für die <i>Card to Card Authentication</i> muss der Public-Key der Root-CA anstelle des Schlüssels der Issuing-CA verwendet und auf den von der SASIS</p>

herausgegebenen Versichertenkarten gespeichert werden.

Auf die optionale Validierung des Versichertenzertifikats durch das Terminal hat die zusätzliche CA keinen Einfluss. Hier kann das Zertifikat der SASIS Issuing CA als Trust-Anker verwendet werden. Optional kann aber auch eine Zertifikatsvalidierung bis zur Root-CA durchgeführt werden.

Hinweis

Durch die geänderte Hierarchie verliert die Issuing-CA der Versichertenorganisation ihre Rolle als Zertifizierungsstelle der CAs der Leistungserbringerorganisationen. Diese Rolle fällt nun der Root-CA zu. Diese kann damit offline betrieben werden.

5.5 eCH-0064: Kapitel 3.6.4.1, PIN-Management

Im Folgenden sind die wesentlichen Unterschiede in Bezug auf das PIN-Management festgehalten. Es werden Sachverhalte beschrieben, deren unterschiedliche Spezifikationen einen Einfluss auf das System der Versichertenkarte haben.

[eCH64] legt fest, dass der Inhaber der Versichertenkarte den Zugriff auf ausgewählte Kategorien der Notfalldaten mit einer PIN sperren können muss. Weiter muss die PIN mit einem PUK entsperrt werden können.

- Die VK-Post implementiert zwei PINs und zwei PUKs.
- Die VK-SASIS implementiert zwei PINs und einen gemeinsamen PUK.

Diese PINs und PUKs werden von den Karten wie folgt verwendet:

[SPEC-Post]	
Schutz-Mechanismus	Benötigt für:
PIN.NOT	<ul style="list-style-type: none"> ▪ Sperrung/Entsperrung des Zugriffs auf die Notfalldaten.
PUK.NOT	<ul style="list-style-type: none"> ▪ Entsperren von PIN.NOT
PIN.USER	<ul style="list-style-type: none"> ▪ Annahme: Freischaltung der Signaturerstellung mittels der X.509-Zertifikate für den kantonalen Modellversuch. (Verwendung aus [SPEC-POST] nicht ersichtlich)
PIN.SO	<ul style="list-style-type: none"> ▪ Annahme: Entsperren von PIN.NOT. (Verwendung aus [SPEC-POST] nicht ersichtlich)
[SPEC-SASIS]	
Schutz-Mechanismus	Benötigt für:
PIN1	<ul style="list-style-type: none"> ▪ Sperrung/Entsperrung des Zugriffs auf die Notfalldaten.
PIN2	<ul style="list-style-type: none"> ▪ Freischaltung der Signaturerstellung mittels der X.509-Zertifikate für den kantonalen Modellversuch.
PUK	<ul style="list-style-type: none"> ▪ Entsperren von PIN1 und PIN2

Auslieferungszustand der Karten:

- PIN1 ist bei der VK-SASIS bei Auslieferung deaktiviert, d.h. der Zugriff auf die Notfalldaten via Card-to-Card Authentisierung durch die HPC wird nicht blockiert. PIN1 muss bei der ersten Aktivierung des Zugriffsschutzes durch den Benutzer gesetzt werden. PIN2 ist auf einen zufälligen Wert gesetzt und muss vom Benutzer bei Teilnahme an einem kantonalen Modellversuch über den PUK neu gesetzt werden.
- Das PIN-Management ist in der [SPEC-POST] nur mangelhaft spezifiziert. Aussagen zum Auslieferungszustand der PINs und PUKs sind nicht möglich.

Die PIN-Mechanismen der VK-Post und der VK-SASIS sind unterschiedlich und müssen von der Middleware kartenspezifisch implementiert werden.

A16.	Kategorie-spezifischer Schutz der Notfalldaten.		
[eCH64]	Beschreibung		
Kapitel 2.3 Kapitel 3.5	<p>[eCH64] legt fest, dass der Inhaber der Versichertenkarte den Zugriff auf ausgewählte Kategorien der Notfalldaten mit einer PIN sperren können muss.</p> <table border="1" data-bbox="448 651 1447 748"> <tr> <td>Hinweis</td> <td>[eCH64] konkretisiert hierbei die Anforderung der [VVK]²⁸ und spezifiziert, dass der PIN Schutz auf einzelne Kategorien der medizinischen Notfalldaten angewendet werden muss²⁹.</td> </tr> </table>	Hinweis	[eCH64] konkretisiert hierbei die Anforderung der [VVK] ²⁸ und spezifiziert, dass der PIN Schutz auf einzelne Kategorien der medizinischen Notfalldaten angewendet werden muss ²⁹ .
Hinweis	[eCH64] konkretisiert hierbei die Anforderung der [VVK] ²⁸ und spezifiziert, dass der PIN Schutz auf einzelne Kategorien der medizinischen Notfalldaten angewendet werden muss ²⁹ .		
[SPEC-POST]	Beschreibung		
Kapitel 2.16	Mit der VK-Post lassen sich die Notfalldaten nur als Ganzes und nicht Kategorie-weise mit einem PIN-Schutz belegen. [eCH64] wird entsprechend nicht eingehalten.		
[SPEC-SASIS]	Beschreibung		
Kapitel 7.1.6	Die [SPEC-SASIS] erfüllt die Vorgaben von [eCH64].		

²⁸ Art. 7 Abs. 4 [VVK]

²⁹ [eCH64] Kapitel 2.3 und Kapitel 3.5.3.3

5.6 eCH-0064: Kapitel 5, Kantonale Modellversuche nach Artikel 16 [VVK]

Im Folgenden sind die wesentlichen Unterschiede in Bezug auf die Definition der Sachverhalte im Zusammenhang mit den kantonalen Modellversuchen festgehalten. Es wurden Sachverhalte beschrieben, deren unterschiedliche Spezifikationen einen Einfluss auf das System der Versichertenkarte haben.

A17.	Kantonale Modellversuche
[eCH64]	Beschreibung
Kapitel 5	<p>[eCH64] spezifiziert nur wenige Sachverhalte im Zusammenhang mit den kantonalen Modellversuchen. Im speziellen sind dies vier dedizierte Files unter DF.KtMV sowie den Anforderungen aus [eCH64] Kapitel 5.</p> <p>Aufgrund der wenig spezifischen Vorgaben unterscheiden sich die [SPEC-POST] und [SPEC-SASIS] entsprechend.</p> <p>Im Wesentlichen fordert [eCH64] die Bereitstellung eines leeren Containers für X.509 Zertifikate mit dem korrespondierenden privaten Schlüssel, welche über Standardapplikationen genutzt werden können. Die Schnittstelle für den Zugriff auf die diesbezüglichen Funktionen auf der Versichertenkarte erfolgt über eine anbieterspezifische Middleware³⁰. Da die zuvor genannten Schnittstellen genau spezifiziert sind und aus Sicht Applikation unabhängig von der Implementierung der Versichertenkarte aufgerufen werden können, fallen Abweichungen zwischen der [SPEC-POST] und der [SPEC-SASIS] nicht ins Gewicht.</p>
[SPEC-POST]	Beschreibung
Kapitel 4	<p>Detaillierte Untersuchungen auf Ebene APDU konnten im Zusammenhang mit kantonalen Modellversuchen nicht durchgeführt werden, da diese in der [SPEC-POST] nicht spezifiziert sind. Die Spezifikation ist dementsprechend zu ergänzen. Die diesbezüglichen Anwendungen sind nur über eine Middleware³¹ zugänglich.</p> <p>a) Die [SPEC-POST] bildet die Filestruktur im Zusammenhang mit den kantonalen Modellversuchen in einer PKCS#15 Struktur ab, welche unter DF.KtMV abgelegt ist.</p> <p>[eCH64] gibt vor, dass alle Files im Zusammenhang mit den kantonalen Modellversuchen, analog zu allen anderen Files, als EF unter DF.KtMV abgelegt werden müssen.</p> <p>b) [SPEC-POST] definiert, dass der zum X.509 Zertifikat korrespondierende private Schlüssel mit einem PIN / PUK Mechanismus</p>

³⁰ Microsoft CAPI, PKCS#11

³¹ AET SafeSign

	<pre> YXN1P29iamVjdENsYXNzPWNSTERpc3RyaWJldG1vb1BvaW50MIIBLAYDVR0gBIIB IzCCAR8wggEQBg1ghXQBWQEDAQQwggEBMEoGCCsGAQUFBwIBFj5odHRwOi8vcmlv b3NpdG9yeS5zd21zc3NpZ24uY29tL1N3aXNzU21nbi11TaWx2ZXItQ1AtQ1BTLVI0 LnBkZjCBsgYIKwYBBQUHAgIwgaUagaJnZXN0dWV0enQgYXVmieFydC4gNDJBIETW RywZw4gdmVydHUGZGUGbCdhcnQuIDQyQSBMQU1hbCwgdmLzdG8gbCdhcnQuIDQy QSBMQU1hbCwgYmFzZWQgb24gYXJ0LiA0MkEgSE1MLCBTUUA4MzIuMTAgLyBSUyA4 MzIuMTAgU2Nod2Vpei9TdWlzc2UvU3ZpenplcmEvU3dpdHplcmxhbmQwCQYHYIV0 AVkKazB5BggrBgEFBQcBAQRtMGswaQYIKwYBBQUHMAKGXWh0dHA6Ly9zd21zc3cy5z aWduZGVtb3Y5b20vY2dpLWJpbi9hdXRob3JpdHkvZG93bmxxvYwQvRTU1RjA0RkVF MUQ2OU11OTVEOTUyQjMxRkMyRUVFRTE5M0U2N0I0OTANBgkqhkiG9w0BAQUFAAOC AQEAduFam+d8eePDhV7nWmLhTTFYfukL8wkmChVaVeeS2uz8qlTL/Ileehibxzf 2c6Bgbj0RXTnWBkZGvLXLG5ynfJ91tIzEnw1ljqqulT1afURbYwibrFltE66g7m7 kbc0ZOY8OckAgXekDH4P5SkGY22BaQQOhGwBpnQJb+1A8SmIk1EB/K8hMYu36Ze dclmvyr8TAdht0ApRYez4+x/3NCOxIFWwCfW2v7miQDX8bcVFU+N/M3BvBlJc5oE 0+fDhHaBjMCFu4MOip72kZwBZNWfBfNTmJrnUmerPgJcVesGRst630vodhYSyO5a lpTicczc/AOW8RdNnCW8sD8U9g== -----END CERTIFICATE----- </pre>
[SPEC-SASIS]	Beschreibung
Kapitel 8	<p>Die [SPEC-SASIS] spezifiziert alle Gegebenheiten im Zusammenhang mit den kantonalen Modellversuchen detailliert auf Stufe APDU.</p> <ol style="list-style-type: none"> Die [SPEC-SASIS] bildet die Filestruktur im Zusammenhang mit den kantonalen Modellversuchen in einer PKCS#15 Struktur ab, welche unter DF.KtMV abgelegt ist. [eCH64] gibt vor, dass alle Files im Zusammenhang mit den kantonalen Modellversuchen, analog zu allen anderen Files, als EF unter DF.KtMV abgelegt werden müssen. Die [SPEC-SASIS] definiert, dass der zum X.509 Zertifikat korrespondierende private Schlüssel mit einem PIN / PUK Mechanismus geschützt wird. [eCH64] spezifiziert jedoch keinen Zugriffsschutz auf den privaten Schlüssel, der in EF.PrK_{X509} gespeichert ist. Die VK-SASIS hat kein persönliches X.509 Zertifikat vorinstalliert und erfüllt somit die Vorgaben von [eCH64]. Diese müssen über einen zusätzlichen, durch die Modellversuche zu definierenden Prozess auf der VK-SASIS gespeichert werden. Die [SPEC-SASIS] definiert zusätzlich zum Authentisierungszertifikat ein X.509 Zertifikat mit einem korrespondierenden, privaten Schlüssel für die Verschlüsselung resp. Entschlüsselung von Daten.

5.7 eCH-0064: Kapitel 6, Definition Zertifikate

Im Folgenden sind die wesentlichen Unterschiede in Bezug auf die Definition der Zertifikate festgehalten. Es wurden Sachverhalte beschrieben, deren unterschiedliche Spezifikationen einen Einfluss auf das System der Versichertenkarte haben.

5.7.1 CV-Zertifikatsformat

A18.	CV-Zertifikatsformat
[eCH64]	Beschreibung
Kapitel 6 ff.	<p>Kapitel 6 ff. spezifizieren das Format und den Inhalt der CV Zertifikate. Die Zertifikate folgen der Norm ISO/EC 7816-8 mit Message Recovery nach [ISO 9796-2].</p> <p>[eCH64] spezifiziert zusätzlich den Aufbau und Inhalt der im CV Zertifikat verwendeten Datenfelder.</p> <p>Die VK-Post und die VK-SASIS spezifizieren und implementieren unterschiedliche Zertifikate. Entsprechend ist die Verwendung der Zertifikate unterschiedlich. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>
[SPEC-POST]	Beschreibung
Kapitel 3.1	<p>Die [SPEC-POST] definiert grundsätzlich das CV Zertifikat nach [eCH64], jedoch in einem leicht abweichenden Zertifikatsformat. Dies war notwendig, weil das in [eCH64] definierte Zertifikatsformat fehlerhaft und dadurch nicht ohne entsprechende Änderungen umsetzbar ist.</p> <p>a) [eCH64] weist die folgenden beiden Fehler auf:</p> <ol style="list-style-type: none"> 1. Falsche Längenangabe der OID (8 Bytes anstelle von 5 Bytes) 2. Falsches Padding resp. Fehler in der Zusammensetzung des Signaturteils des Zertifikats. Die Länge der Signaturdaten inkl. Padding entspricht nicht 1024 Bit, was ein Padding „6BB..“ anstelle von „6A“ zur Folge hätte. <p>Die [SPEC-POST] korrigiert den Fehler in [eCH64] über die Längenberechnung der Signaturdaten im CV Zertifikat. Sie korrigiert die zuvor genannten Fehler durch Hinzufügen des zusätzlichen Elements CISD in der Signatur.</p> <p>b) [SPEC-POST] spezifiziert das Feld CHA nicht gemäss [eCH64]. CHA hat den Zweck, die Zugriffsrechte des Karteninhabers in Bezug auf Daten, die in einer Versichertenkarte gespeichert sind, festzulegen.</p> <p>Gemäss [eCH64] Kapitel 6.1.6 muss beispielsweise CHA für eine PDC wie folgt kodiert werden:</p>

	<p>AID(DF.NOT) 00</p> <p>Für eine PDC ergäbe sich somit folgender Wert für den CHA</p> <p>d7 56 83 21 05 00 00</p> <p>[SPEC-POST] spezifiziert anstelle der AID(DF.NOT) die Zeichenfolge „DF.NOT“ mit dem zusätzlichen Wert 0x00.</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p>Hinweis [eCH64] Kapitel 6.1.6 definiert, dass der CHA den Zweck hat, die Zugriffsrechte des Karteninhabers in Bezug auf Daten, die in einer anderen Karte gespeichert sind, festzulegen.</p> <p>Falls die VK-Post für die Überprüfung der Zugriffsrechte den ganzen CHA Datensatz³⁵ verwendet, wäre ein einheitliches Leistungserbringerzertifikat, unabhängig von der Zertifikathierarchie, der Schlüssellänge, dem verwendeten Padding oder der Kodierung, mit der VK-SASIS nicht möglich.</p> </div> <p>c) [SPEC-POST] spezifiziert in Kapitel 3.1.2 fälschlicherweise den Wert 04 für CA Zertifikate und den Wert 03 für Enduser Zertifikate. Gemäss [eCH64] ist der Wert 03 für CA Zertifikate und der Wert 04 für Enduser Zertifikate reserviert. Die [SPEC-POST] ist diesbezüglich anzupassen, die Implementierung in der VK-Post ist jedoch korrekt.</p> <p>Hex-Dump des CV Zertifikats</p> <pre> 00000000 7F 21 81 D5 5F 37 81 80 9B 9F 00 BC 1D AC A7 F7 .!...7..... 00000010 69 6F C3 0E EF FB 4D 70 01 63 2A 64 15 A1 ED D5 io....Mp.c*d... 00000020 FF 5F EE 8A 3A 9D 97 8C 62 01 99 8C 71 D9 3B 38 ..:...b...q.;8 00000030 CB 8B 56 20 BD 0B 31 B1 B0 0B 0D 08 0D 42 7D CC ..V...1...B) 00000040 BD F9 EC 9E 6A 0C 9D 1F 9B BC 43 C1 44 38 C4 3C ...j....C.D8.< 00000050 20 87 14 ED FB 3B 5B 90 C1 C5 21 CC AD 50 67 13 ...;[...!...Pg. 00000060 7E 78 4B 25 7A 6E 8E 38 21 74 C1 E8 A8 92 64 F2 ~xK&zn.8!t....d. 00000070 E7 5A A0 B2 AC 5C D7 07 DB 09 93 92 6C 29 76 28 .Z...l...v(00000080 DC 03 3E 36 86 51 2C AE 5F 38 44 73 32 1A E4 FA ..>6.Q,..8Ds2... 00000090 4A 58 47 2B 78 3B 84 45 3C A3 FE E1 A8 F6 45 7B JXG+x;.E<.....E(000000A0 98 AD 99 7C 2D B3 02 79 1E C3 E2 14 E0 F3 F9 FC ... -...y..... 000000B0 51 D8 D1 46 5E 83 F6 FE 6B A9 65 1E 5A FD 12 5E Q..F...k.e.Z...^ 000000C0 1D 8B 7E 8D 08 A8 31 62 E9 09 33 00 01 00 01 42 ...~...1b..3....B 000000D0 08 43 48 44 53 50 60 01 09 .CHDSP`... </pre> <p>Hex-Dump des Signaturfeldes³⁶</p> <pre> 000000 6a 04 43 48 44 53 50 60 01 09 00 00 00 00 00 00 j.CHDSP`..... 000010 80 75 69 99 99 99 99 99 99 94 44 46 2e 4e 6f 74 .ui.....DF.Not 000020 00 2b 0e 03 02 0f 13 03 31 10 01 a0 4a d5 23 74 +.....1...J0#t 000030 ba eb 26 88 35 92 fa 5f b2 8e 1e e2 14 45 c5 5f 'e&.5'ú²..ä.EÄ_ 000040 9d e9 b7 4c 7a ba 4a d0 19 99 09 03 27 b4 94 82 .é.Lz°J0.....' 000050 bf 90 93 97 4f aa 6b a3 34 26 4d 14 05 8b 0f 8f g...O°kE4&M.... 000060 d5 01 80 8d 17 63 e7 be c8 a3 d4 6f 80 bd 7e 19 Ö....cç&fÖo..k~. 000070 bc cd 46 6a 82 ae 14 35 cd fd 1d 2f 7b b4 b5 bc *iFj.©.5iÿ./('µ* </pre>
[SPEC-SASIS]	Beschreibung
Kapitel 6.1.1.1	<p>Die [SPEC-SASIS] definiert entgegen [eCH64] ein Zertifikatsformat auf der Basis von [ISO 8825-1]. Das in der [SPEC-SASIS] definierte Zertifikat weicht vollständig vom dem in [eCH64] definierten Format ab. Folgend sind die wesentlichen Abweichungen aufgeführt:</p> <ol style="list-style-type: none"> 1. Zertifikatsstruktur: Das Auslesen der einzelnen Datenfelder erfolgt

³⁵ „AID(DF.NOT) || XY“

³⁶ Inhalt der zu signierenden Daten im CV Zertifikat (Tag 5F37)

grundsätzlich unterschiedlich zu [eCH64].

2. Padding: PKCS#1 V1.5 anstelle von ISO 9796-2 gemäss [eCH64]
3. Schlüssellängen: RSA Modulus n = 2048 anstelle von n = 1024 gemäss [eCH64]
4. Hash Algorithmen: SHA-256 anstelle von SHA-1 gemäss [eCH64]
5. Datenfelder

Die Grösse und der Inhalt der folgenden Datenfelder wurden in [SPEC-SASIS] abweichend von [eCH64] spezifiziert:

- **CPI**: Die [SPEC-SASIS] definiert von [eCH64] abweichende CPI Werte.
- **CAR**: Die [SPEC-SASIS] definiert von [eCH64] abweichende CAR Werte.
- **CHR**: Die [SPEC-SASIS] definiert von [eCH64] abweichende CAR Werte.
- **OID**: Die [SPEC-SASIS] definiert von [eCH64] abweichende CAR Werte.
- **CHA³⁷**: Die [SPEC-SASIS] definiert von [eCH64] abweichende CHA Werte.

Die oben genannten Datenfelder enthalten jedoch alle für einen Einsatz als Versichertenkarte notwendigen Informationen. Dies sind u.a. Daten zur Identifizierung der Versicherten (ICCSN), zur Bildung von Zertifikatsvalidierungsketten (CAR, CHR), sowie zur Autorisierung des Zugriffes auf die Notfalldaten (CHA).

Hex-Dump

```

000000 7f 21 82 02 65 7f 4e 82 01 5b 5f 29 01 55 42 10  !..e.N..[.]..B.
000010 43 48 53 41 53 62 32 30 30 39 30 30 33 30 30 31  CHSASb2009003001
000020 7f 49 82 01 12 06 07 60 85 74 05 22 02 01 81 82  .I.....t."....
000030 01 00 bf 64 6d 4f b3 25 c6 06 60 0a 3f 3e 64 31  ..çdmO³%Æ.´.?>d1
000040 e9 4b 66 c3 2e 5a 11 cb 57 7d e3 26 85 29 70 22  éKfÄ.Z.ÈW)â&.p"
000050 25 1d 04 e8 5d 6f 6e 06 4c bb ac 91 fd a8 37 5b  %..è|on.L»-'ý'7[
000060 ab a5 89 69 bd b7 93 23 ee cc f6 81 50 83 72 db  «¥.i¼. #iîö.P.rÛ
000070 9f 9c 8a 9b f4 05 93 13 1a 1d df e1 65 6f 68 de  ...ô....ßáeohP
000080 de 20 ca b3 a2 62 e6 96 da a9 06 86 98 ab ea c8  Þ Ê³çbæ.Ú@...«éÈ
000090 fa 31 85 e0 7c bf e9 4b a2 9f 2a 4a 57 ec 16 00  úl.à|çKç.*JWi..
0000a0 81 d3 18 e9 ca 59 c9 77 22 09 81 e6 ae 18 a0 bc  .Ó.éÈYÈw" ..æ@. ¼
0000b0 02 92 fa dc cd 15 a7 64 43 17 f3 4c 75 3a 16 38  .´úŰí.šdc.óLu:.8
0000c0 7e 96 c4 ad f8 8d 6f 11 a2 a6 b3 bb 85 0e fd 8c  ~.Ä-ø.o.ç|³»..ý.
0000d0 ed 96 c7 b7 33 14 4c a5 c7 eb 01 59 82 f0 96 b1  i.ç.3.LYÇè.Y.ð.±
0000e0 8e cd 38 86 18 ed 3a 76 7a 74 ca bf ac 78 2a ec  .î8..i:vztÈç-x*î
0000f0 60 4f c6 b9 92 52 45 6e 72 5d 55 b8 e9 ab 8b f1  `OE!`REnr]U.é«.ñ
000100 05 1e fc aa 57 bd 1d c6 30 87 ae a2 77 19 e2 36  ..ü*W¼.È0.çw.â6
000110 6b c2 18 52 bb 5c 1f c9 94 3b a0 5e 4b 33 2e 8f  KÄ.R»\..È.; ^K3..
000120 bc be 27 de e4 ee 9e e9 1f 78 0c 51 92 37 95 58  ¼¼'Páí.é..x.Q'7.X
000130 45 09 82 03 01 00 01 5f 20 0a 80 75 60 12 34 00  E.....u'.4.
000140 00 00 20 85 7f 4c 0c 06 07 60 85 74 05 22 01 01  .. .L...t."..
000150 53 01 00 5f 25 06 00 09 00 01 00 01 5f 24 06 01  S.. %.....$.
000160 04 00 06 03 00 5f 37 82 01 00 4b 25 f0 34 b2 0d  .... 7...K%04².
    
```

³⁷ Vergleiche auch CHA der [SPEC-POST]. [SPEC-SASIS] verzichtet vollständig auf den Prefix AID(DF.NOT)

```

000170 dd 9d 2b 0e fd 8a 55 14 b0 39 04 82 b5 30 1e ac Y.+ÿ.U.°9..µ0.-
000180 84 f6 0f cf de c4 a7 63 6b a1 21 72 87 8d cb cd .ó.IPÅ$ck;!r..Ei
000190 ff b3 61 51 06 a9 72 a7 31 59 32 0c b3 90 c1 ef ý'aQ.ør$1Y2.³.Åi
0001a0 9d 09 a5 9c 65 01 d2 42 af ae 39 5d c0 9c 34 6a ..Y.e.ÖB@9jÄ.4j
0001b0 5e 2e 92 44 cf 46 2b ce ce 34 6b 3b 2f 3e 47 ff ^.'DIF+îî4k;/>Gý
0001c0 93 63 d4 a7 94 ad 3c 3f 08 4d 4f f6 26 c2 7a 57 .c0$.-<?.MOö&ÅzW
0001d0 2e 9e 2b e1 c2 db 3a 05 fe 62 83 20 cb 8d 63 73 ..+áÂÛ:;bb. E.cs
0001e0 9c b7 31 5d d4 28 c0 28 94 a6 61 4e 02 30 d5 34 .·l]Ô(À(.!aN.004
0001f0 1c f7 05 7b fd 34 5e 5c 84 66 d6 4f a4 3d 79 d1 .+. {ý4^\.f00H=yÑ
000200 4a b9 be c7 00 04 70 f3 a9 59 6b 91 1f b9 42 03 J+¼Ç..pó@Yk',¹B.
000210 28 01 39 9e 46 13 7f 7c 13 29 46 74 92 80 ad f7 (.9.F..|.)Ft'.->þ
000220 1d 37 e8 2e cb 08 e3 67 ac df 88 75 27 75 3f 84 .7è.Ë.äg-ß.u'u??.
000230 9e d2 6b 0a 73 9d 46 14 61 6e b0 9e cd 07 b0 87 .Ök.s.F.an°.î.°.
000240 e2 95 7e cb 78 c6 bc 19 36 55 03 c2 e9 d2 d0 cc ä.~ËxEM.6U.ÄêÖDİ
000250 ad 20 52 d0 94 b6 70 f8 8d db 9f 8f 5c ff 6e ee -RB.Tpø.Û..ÿnî
000260 f8 9f 38 7f be e1 21 6e 38 90 ø.8.¾á!n8.

```

ASN.1 Struktur

```

0000 61 265: [APPLICATION 33] {
0005 6E 15B: [APPLICATION 78] {
000A 69 1: [APPLICATION 41]
:
:
000E 42 10: [APPLICATION 2] 'CHSASb2009003001'
0020 69 112: [APPLICATION 73] {
0025 06 7: OBJECT IDENTIFIER '2 16 756 5 34 2 1'
002E 81 100: [1]
:
: BF 64 6D 4F B3 25 C6 06 60 0A 3F 3E 64 31 E9 4B
:
: 66 C3 2E 5A 11 CB 57 7D E3 26 85 29 70 22 25 1D
:
: 04 E8 5D 6F 6E 06 4C BB AC 91 FD A8 37 5B AB A5
:
: 89 69 BD B7 93 23 EE CC F6 81 50 83 72 DB 9F 9C
:
: 8A 9B F4 05 93 13 1A 1D DF E1 65 6F 68 DE DE 20
:
: CA B3 A2 62 E6 96 DA A9 06 86 98 AB EA C8 FA 31
:
: 85 E0 7C BF E9 4B A2 9F 2A 4A 57 EC 16 00 81 D3
:
: 18 E9 CA 59 C9 77 22 09 81 E6 AE 18 A0 BC 02 92
:
: FA DC CD 15 A7 64 43 17 F3 4C 75 3A 16 38 7E 96
:
: C4 AD F8 8D 6F 11 A2 A6 B3 BB 85 0E FD 8C ED 96
:
: C7 B7 33 14 4C A5 C7 EB 01 59 82 F0 96 B1 8E CD
:
: 38 86 18 ED 3A 76 7A 74 CA BF AC 78 2A EC 60 4F
:
: C6 B9 92 52 45 6E 72 5D 55 B8 E9 AB 8B F1 05 1E
:
: FC AA 57 BD 1D C6 30 87 AE A2 77 19 E2 36 6B C2
:
: 18 52 BB 5C 1F C9 94 3B A0 5E 4B 33 2E 8F BC BE
:
: 27 DE E4 EE 9E E9 1F 78 0C 51 92 37 95 58 45 09
0132 82 3: [2]
:
: 01 00 01
:
: }
0137 60 A: [APPLICATION 32]
:
: 80 75 60 12 34 00 00 00 20 85
0144 6C C: [APPLICATION 76] {
0147 06 7: OBJECT IDENTIFIER '2 16 756 5 34 1 1'
0150 53 1: [APPLICATION 19]
:
: 00
:
: }
0153 65 6: [APPLICATION 37]
:
: 00 09 00 01 00 01
015C 64 6: [APPLICATION 36]
:
: 01 04 00 06 03 00
:
: }
0165 77 100: [APPLICATION 55]
:
: 4B 25 F0 34 B2 0D DD 9D 2B 0E FD 8A 55 14 B0 39
:
: 04 82 B5 30 1E AC 84 F6 0F CF DE C4 A7 63 6B A1
:
: 21 72 87 8D CB CD FF B3 61 51 06 A9 72 A7 31 59
:
: 32 0C B3 90 C1 EF 9D 09 A5 9C 65 01 D2 42 AF AE
:
: 39 5D C0 9C 34 6A 5E 2E 92 44 CF 46 2B CE CE 34
:
: 6B 3B 2F 3E 47 FF 93 63 D4 A7 94 AD 3C 3F 08 4D
:
: 4F F6 26 C2 7A 57 2E 9E 2B E1 C2 DB 3A 05 FE 62
:
: 83 20 CB 8D 63 73 9C B7 31 5D D4 28 C0 28 94 A6
:
: 61 4E 02 30 D5 34 1C F7 05 7B FD 34 5E 5C 84 66
:
: D6 4F A4 3D 79 D1 4A B9 BE C7 00 04 70 F3 A9 59
:
: 6B 91 1F B9 42 03 28 01 39 9E 46 13 7F 7C 13 29
:
: 46 74 92 80 AD F7 1D 37 E8 2E CB 08 E3 67 AC DF
:
: 88 75 27 75 3F 84 9E D2 6B 0A 73 9D 46 14 61 6E
:
: B0 9E CD 07 B0 87 E2 95 7E CB 78 C6 BC 19 36 55
:
: 03 C2 E9 D2 D0 CC AD 20 52 D0 94 B6 70 F8 8D DB
:
: 9F 8F 5C FF 6E EE F8 9F 38 7F BE E1 21 6E 38 90
:
: }

```

5.7.2 Signaturformat im Card to Card-Authentisierungsverfahren

A19.	Signaturformat im Card to Card-Authentisierungsverfahren			
[eCH64]	Beschreibung			
Kapitel 3.6 ff. Kapitel 6.1	<p>Gemäss [eCH64] muss die HPC während der für den Zugriff auf die Notfalldaten notwendigen Card to Card Authentication eine Signatur einer Zufallszahl erstellen, welche anschliessend von der PDC geprüft wird.</p> <p>Die genauen Signaturparameter (Signatureingangsdaten, Signaturformat) werden von [eCH64] nicht explizit spezifiziert. Es wird jedoch auf [ISO 9798-3] verwiesen. [eCH64] ist diesbezüglich zu wenig präzise und sollte überarbeitet werden.</p> <p>Die VK-Post und die VK-SASIS spezifizieren und implementieren unterschiedliche Signaturformate. Die Interoperabilität der unterschiedlichen Spezifikationen kann über eine geeignete Middleware hergestellt werden.</p>			
[SPEC-POST]	Beschreibung			
Kapitel 3.2	<p>Die [SPEC-POST] definiert als Hash-Eingangsdaten der Signatur nicht nur die 8 Byte lange Zufallszahl, sondern fügt in Anlehnung an [ISO 9798-3] zusätzlich noch eine von der HPC erzeugte Zufallszahl sowie die ICCSN_B der Versichertenkarte hinzu.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="background-color: #e0f0ff; width: 20%;">Details</td> <td> <p>[SPEC-SASIS] Kapitel 6.2.23 weist darauf hin, dass aufgrund der erstellten Signatur kein Rückschluss auf eine einmal hergestellte Verknüpfung zwischen einer bestimmten Versichertenkarte und der Leistungserbringerkarte möglich sein sollte.</p> <p>Das Signaturverfahren gemäss [SPEC-POST] lässt aufgrund der signierten ICCSN_B der Versichertenkarte eine logisch nachweisbare Verknüpfung zur HPC zu.</p> </td> </tr> </table>		Details	<p>[SPEC-SASIS] Kapitel 6.2.23 weist darauf hin, dass aufgrund der erstellten Signatur kein Rückschluss auf eine einmal hergestellte Verknüpfung zwischen einer bestimmten Versichertenkarte und der Leistungserbringerkarte möglich sein sollte.</p> <p>Das Signaturverfahren gemäss [SPEC-POST] lässt aufgrund der signierten ICCSN_B der Versichertenkarte eine logisch nachweisbare Verknüpfung zur HPC zu.</p>
Details	<p>[SPEC-SASIS] Kapitel 6.2.23 weist darauf hin, dass aufgrund der erstellten Signatur kein Rückschluss auf eine einmal hergestellte Verknüpfung zwischen einer bestimmten Versichertenkarte und der Leistungserbringerkarte möglich sein sollte.</p> <p>Das Signaturverfahren gemäss [SPEC-POST] lässt aufgrund der signierten ICCSN_B der Versichertenkarte eine logisch nachweisbare Verknüpfung zur HPC zu.</p>			
[SPEC-SASIS]	Beschreibung			
Kapitel 6.1.1.2	<p>Obwohl [eCH64] Kapitel 3.6 ff. das Padding der Signatur im Zusammenhang mit der <i>Card to Card Authentication</i> nicht explizit definiert, muss aufgrund der Signatureigenschaften der in [eCH64] definierten CV Zertifikate ausgegangen werden, dass die HPC das Padding gemäss [ISO 9796-2] anwenden muss.</p> <p>[SPEC-SASIS] definiert das Padding und die RSA Signatur gemäss PKCS#1 V1.5³⁸.</p>			

³⁸ Analoges Verfahren wie für die gemäss [SPEC-SASIS] definierten CV Zertifikate.

6 HPC

[eCH64] spezifiziert Einzelheiten im Zusammenhang mit den Versichertenkarten. Die Anforderungen an die HPC sind in [eCH64], Kapitel 3.1.4 nur kurz beschrieben. Die detaillierte Beurteilung der [FMH-HPC] analog zur [SPEC-POST] oder [SPEC-SASIS] war auch nicht Bestandteil der Expertise.

Um die Funktionsfähigkeit des Systems mit effektiv produzierten Karten (FMH-HPC, VK-Post und VK-SASIS) beurteilen zu können, wurde die FMH-HPC im Zusammenhang mit der Card to Card Authentisierung einbezogen.

6.1 Technische Beschreibung der FMH-HPC

Die technische Beschreibung der [FMH-HPC] hat zum Ziel, die notwendigen Informationen bereitzustellen, damit die FMH-HPC in Applikationen integriert werden kann. Die [FMH-HPC] ist keine Detailspezifikation vergleichbar mit [SPEC-POST] oder [SPEC-SASIS] und enthält entsprechend wenige technische Einzelheiten über die Struktur der Datenobjekte³⁹ und Schnittstellen der FMH-HPC.

Es kann davon ausgegangen werden, dass sich die Autoren der [FMH-HPC] mit den Autoren der Versichertenkarten abgestimmt haben. Die in [eCH64] fehlerhaft definierten Strukturen wurden analog zu den Spezifikationen der [SPEC-POST] korrigiert⁴⁰. Analog verhält es sich mit interpretierbaren Strukturen.

Zu bemerken ist, dass die [FMH-HPC] keine Angaben zu der Zertifikatstruktur macht, wie sie in der [SPEC-SASIS] definiert wurde. Die praktischen Untersuchungen haben aber gezeigt, dass die FMH-HPC mit der VK-Post und der VK-SASIS interoperabel ist.

³⁹ Insbesondere keine Angaben zu den FID

⁴⁰ Hinweis: Die abgestimmten Strukturen weisen Unterschiede auf. Im Vergleich zur [SPEC-POST] beschreibt die [FMH-HPC] die Kodierung CED im CISD unterschiedlich. CED [SPEC-POST] := YYYY, CED [FMH-HPC] := MMY

Hinweis	<p>Aufgrund der in Kapitel 5 aufgeführten Unterschiede der Spezifikationen und Implementationen der Versichertenkarten kommt der HPC eine entscheidende Bedeutung für das korrekte Funktionieren des gesamten Systems zu.</p> <p>In diesem Dokument wurde mehrfach erwähnt, dass die Interoperabilität der unterschiedlich implementierten Versichertenkarten über eine geeignete Middleware hergestellt werden kann. Dies ist möglich, weil die FMH-HPC die folgenden zusätzlichen Eigenschaften gegenüber [eCH64] erfüllt⁴¹:</p> <ol style="list-style-type: none">1. Die HPC kann RSA RAW Signaturen ausführen.2. Die HPC unterstützt 1024 Bit und 2048 Bit RSA Schlüssel. <p>Die durchgeführten praktischen Tests haben gezeigt, dass die FMH-HPC die oben genannten Anforderungen erfüllt und somit das Funktionieren des Systems sichert. Folglich sind die VK-Post und die VK-SASIS unter Verwendung einer geeigneten Middleware und der FMH-HPC, bestückt mit zwei unterschiedlichen Leistungserbringerzertifikaten, interoperabel. Siehe [Expertise VK - Tech].</p>
---------	--

⁴¹ Diese Anforderungen ergeben sich nicht aus [eCH64] und sind demnach nicht verbindlich für die Umsetzung der HPC.

7 Ursachen der Abweichungen

In diesem Kapitel werden die Ursachen der unterschiedlichen Spezifikationen und Versichertenkarten aus Sicht der Autoren der Expertise beschrieben.

7.1 Kompetenzen, Verantwortung und Absprachen

Die Koordination und Zusammenarbeit zwischen den Versicherern untereinander sowie zwischen den Versicherern und dem EDI war unzureichend und führte zu den unterschiedlichen Spezifikationen und Umsetzungen der Versichertenkarte.

7.1.1 Aus Sicht EDI resp. BAG

Gemäss Art. 2 Abs. 2 VVK müssen die von den Versicherern herausgegebenen Karten untereinander kompatibel sein. Die Festlegung der hierfür notwendigen technischen Einzelheiten, inklusive der Normen und Standards, wurde gemäss Art. 17 VVK an das EDI delegiert. Die [VVK-Erläuterung] zu Art. 17 [VVK] präzisieren, dass das EDI (resp. das BAG) die Vorgaben unter Einbezug der interessierten Kreise festlegen soll. Im Speziellen sollen die Normen und Standards auf Empfehlung einer Fachgruppe des Vereins eCH festgelegt werden.

eCH⁴² ist ein Verein im Sinn von Art. 60 ff. des Schweizerischen Zivilgesetzbuches mit dem Zweck zur Förderung von eGovernment-Standards. Problematisch hierbei war, dass die „Empfehlung“ der Fachgruppe des Vereins eCH direkt über [eCH64] in die [VVK-EDI] eingeflossen sind. Das EDI hat somit die ihr gemäss Art. 17 VVK übertragene Verantwortung zur Regelung der technischen Standards⁴³ an einen Verein übertragen, auf den das EDI kaum mehr Einfluss hat.

Eine neue Version von [eCH64], welche allfällig festgestellte Fehler korrigiert oder Erweiterungen und Änderungen zum bestehenden Standard definiert, kann das EDI nur unter Einbezug des Vereins eCH veranlassen. Alternativ hierzu könnte das EDI, unabhängig vom Verein eCH, Korrekturen, Erweiterungen oder Änderungen über dedizierte Dokumente definieren. Dies führt aber zu mehreren lose gekoppelten Dokumenten und erschwert die Umsetzung der entsprechenden Vorgaben.

7.1.2 Aus Sicht der Versicherer

Gemäss [VVK-Erläuterung] zu Abs. 2 [VVK] sorgen die Versicherer dafür, dass die von ihnen herausgegebenen Versichertenkarten untereinander kompatibel sind.

Der Gesetzgeber hatte die Absicht, die Versicherer darauf zu verpflichten, die technische Umsetzung der Versicherungskarte auf der Basis des durch das EDI vorgegebenen Standards so zu gestalten, dass diese untereinander kompatibel sind.

⁴² Statuten vom 31. Oktober 2006

⁴³ Erarbeitung und Weiterführung

Da es sich bei [eCH64] nicht um einen Implementationsstandard, sondern um einen Konzeptions-, Struktur- und Verfahrensstandard handelt (siehe Kapitel 4.1), mussten die Versicherer entsprechende Detailspezifikationen verfassen, welche eine Umsetzung der Versichertenkarte erst ermöglichen.

Aufgrund der unterschiedlichen Spezifikationen und Umsetzung der Versichertenkarten muss davon ausgegangen werden, dass die Versicherer der Verpflichtung zur Wahrung der Kompatibilität nicht oder nur unzureichend nachgekommen sind und sich entsprechend unzureichend untereinander und mit dem EDI resp. mit dem BAG abgesprochen haben.

7.2 Technische Verfahren und Standards

Die technischen Verfahren und Standards haben sich über die Dauer der Erarbeitung von [eCH64] und den entsprechenden Detailspezifikationen [SPEC-POST] und [SPEC-SASIS] weiterentwickelt. Insbesondere haben sich die Eigenschaften und Kapazitäten der Chipkarte weiterentwickelt.

[eCH64] hatte zum Ziel, Minimalanforderungen festzulegen, die von möglichst vielen Kartenanbietern erfüllt werden können. Diese Minimalanforderungen hatten u.a. Einfluss auf die Festlegung der Zertifikatsstrukturen, Schlüssellängen und Padding-Verfahren.

Die [SPEC-POST] beschreibt die in [eCH64] definierten Verfahren und Standards im Zusammenhang mit den Zertifikatsstrukturen, Schlüssellängen und Padding-Verfahren, obwohl diese heute kaum mehr eingesetzt werden. Die [SPEC-SASIS] hingegen beschreibt Zertifikatsstrukturen, Schlüssellängen und Padding-Verfahren, die nicht in [eCH64] enthalten sind, dafür aber in modernen Applikationen Einsatz finden.

Die interdisziplinäre und organisationsübergreifende Erarbeitung von Standards und Spezifikationen, die fortwährenden technischen Weiterentwicklungen unterliegen und über einen längeren Zeitraum dauern, setzen eine klar definierte Projektorganisation voraus, die allfällige Änderungen oder Erweiterungen formell koordiniert und für verbindlich erklärt. Eine solche Projektorganisation fehlte oder agierte unzureichend.

8 Empfehlungen für das weitere Vorgehen

Die Ziele des Systems der Versichertenkarte sind gemäss [VVK-Erläuterung] folgend aufgeführt:

- Effiziente Administration durch Reduktion des administrativen Aufwandes bei der Abrechnung von Leistungen
- Verbesserung der medizinischen Qualität und Sicherheit der Versorgung mit der Möglichkeit, persönliche Daten auf der Versichertenkarte zu speichern
- Stärkung der Eigenverantwortung der Versicherten (Entscheid über Umfang und Verwendung der persönlichen Daten)

Die zuvor genannten Ziele können mit den aktuell verfügbaren Versichertenkarten der Post und SASIS unter Verwendung einer geeigneten Middleware umgesetzt werden⁴⁴. Um die Nachhaltigkeit der Lösung sicherzustellen, sollten jedoch die Grundlagen im Zusammenhang mit den technischen Spezifikationen neu geregelt werden.

8.1 Allgemeine Empfehlungen

8.1.1 TAV VVK-EDI

Die Verantwortlichkeiten bez. der Definition der Anforderungen an das System der Versichertenkarte sollte neu geregelt werden. Das BAG hat die Möglichkeit, die Definitionen der Anforderungen als technische und administrative Vorschriften (TAV VVK-EDI) zu verfassen und über [VVK-EDI] zu referenzieren. Mit diesem Vorgehen sichert sich das BAG die alleinige Verantwortung über die Festlegung der technischen Anforderungen und kann allfällige Korrekturen, Änderungen oder Erweiterungen über ein ordentliches Vernehmlassungsverfahren einleiten.

Die TAV VVK-EDI wird in enger Zusammenarbeit mit den jeweiligen Anbietern und Experten verfasst. Eine Fachgruppe des Vereins eCH kann, wie in den [VVK-Erläuterung] festgelegt, entsprechende Empfehlungen einbringen.

Es kann z.B. aus politischen Gründen sinnvoll sein, die Regelung nicht auf Stufe Amt sondern auf Stufe Departement vorzunehmen. In diesem Fall werden die Vorgaben nicht in eine TAV sondern in die VVK-EDI aufgenommen.

8.1.2 Testdaten und Referenzimplementation

Die verwendeten Normen und Standards sind interpretierbar. Eine Referenzimplementation mit entsprechenden Testdaten soll sicherstellen, dass die einzelnen Spezifikationen und Umsetzungen der Versichertenkarten kompatibel zueinander im Sinne des Gesetzgebers sind.

⁴⁴ Die nach Art. 2 Abs. 2 VVK geforderte Kompatibilität der Versichertenkarten im Sinne des Gesetzgebers ist aus Sicht der Autoren dieser Expertise jedoch nicht gegeben.

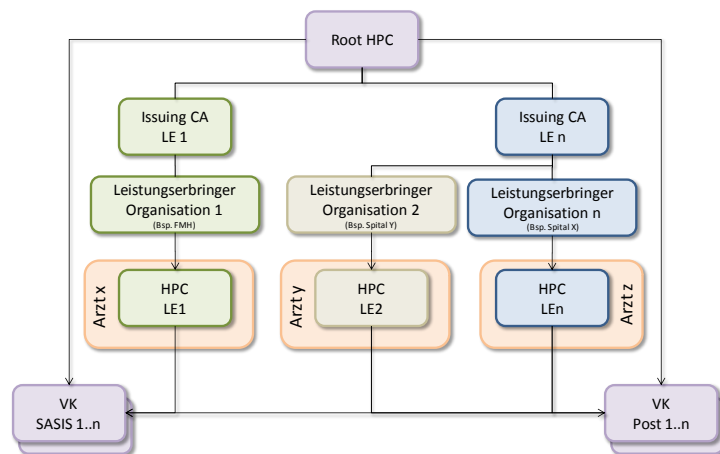
Die Referenzimplementation sollte vom BAG übergeordnet als verbindlich erklärt werden und von einem unabhängigen Anbieter⁴⁵ im Auftrag der Versicherer im Sinne von Art. 2 Abs. 2 [VVK] bereitgestellt werden. Bevor eine Versichertenkarte oder eine HPC ausgerollt werden darf, muss sie die Testprozeduren der Referenzimplementation bestehen⁴⁶.

Mit dem Einsatz einer Referenzimplementation werden die Versicherer in ihrer Verpflichtung unterstützt, die Kompatibilität der Versichertenkarten sicherzustellen.

Hinweis	Ein analoger Sachverhalt ist Zusammenhang mit ICAO-konformen Reisepässen gegeben. Das Golden Reader Tool ⁴⁷ (GRT) ist eine Referenzanwendung zum Auslesen elektronischer Ausweisdokumente. Es wurde im Auftrag des BSI entwickelt und wird weltweit genutzt, um die richtige Umsetzung von Technischen Richtlinien und Spezifikationen bei der Erstellung von elektronischen Ausweisdokumenten nachzuprüfen und die Interoperabilität der Ausweisdokumente verschiedener Nationen nachzuweisen.
---------	---

8.1.3 Public Key Infrastruktur

Die PKI sollte so harmonisiert werden, dass ein spezifischer Leistungserbringer nur ein einzelnes Zertifikat auf seiner HPC benötigt. Die PKI soll bezüglich der Registrierung der Leistungserbringer und Verwaltung der Zertifikate möglichst flexibel gehalten werden. Folgend ist ein Beispiel aufgeführt, welches die zuvor genannten Anforderungen erfüllt.



Alternative Modelle mit zusätzlichen CA-Hierarchien sind denkbar. Entscheidend ist, dass alle Versichertenkarten einen definierten öffentlichen Schlüssel speichern, über den die jeweiligen Leistungserbringer authentisiert und autorisiert werden können.

⁴⁵ Der Anbieter sollte nicht an der Umsetzung oder Bereitstellung der Versichertenkarte teil haben

⁴⁶ Formelle Abnahme des BAG

⁴⁷ https://www.bsi.bund.de/DE/Themen/ElektronischeAusweise/Projekte/projekteGRT/GRT_node.html

8.2 Weiteres Vorgehen

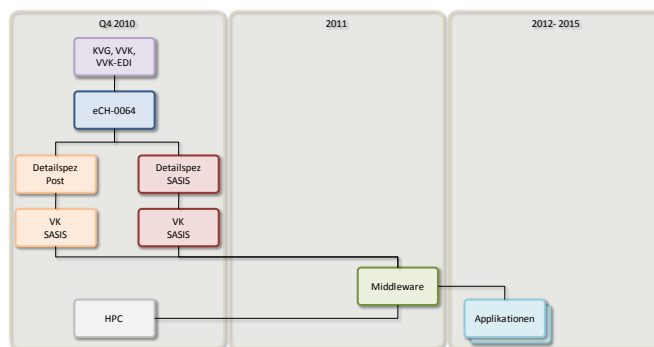
Die Aktivitäten für das weitere Vorgehen können so aufgeteilt werden, dass diese die Weiterentwicklung des bestehenden Systems sowie die Spezifikation des zukünftigen Systems gewährleisten. Die Arbeiten an den beiden Aktivitäten sollten parallel durchgeführt werden.

Die zeitlichen Angaben sind grobe Abschätzungen und müssen mit den jeweiligen Anbietern abgesprochen werden. Die einzelnen Projektphasen müssen sorgfältig geplant und aufeinander abgestimmt werden. Entsprechende Verantwortlichkeiten und Meilensteine sind zu definieren.

8.2.1 Weiterentwicklung des bestehenden Systems

Die Weiterentwicklung des bestehenden Systems hat zum Ziel, möglichst rasch verschiedene Fachapplikationen an das System anzubinden. So können praktische Erfahrungen im Umgang mit den Versichertenkarten im Zusammenhang mit Leistungserbringern und Leistungsempfängern gesammelt werden.

Folgend ist ein möglicher zeitlicher Ablauf der Weiterentwicklung des bestehenden Systems aufgezeigt:



Komponente	Beschreibung		
VK-Post VK-SASIS HPC	Aktuell verfügbare Komponenten.		
Middleware	<p>Voraussetzung für die Interoperabilität ist eine Middleware, welche die unterschiedlichen technischen Umsetzungen der Versichertenkarten abstrahiert. Die Middleware sollte bis Ende 2011 definiert und umgesetzt werden können.</p> <p>a) Die Middleware kann eine einheitliche, anbieterunabhängige Schnittstelle definieren, welche durch das BAG in Zusammenarbeit mit den jeweiligen Anbietern vorgegeben wird.</p> <table border="1" data-bbox="571 1865 1442 1957"> <tr> <td>Hinweis</td> <td>Die anbieterunabhängige Schnittstelle könnte künftig als technische und administrative Vorschrift (TAV) verfasst werden und aus [VVK-EDI] referenziert werden.</td> </tr> </table>	Hinweis	Die anbieterunabhängige Schnittstelle könnte künftig als technische und administrative Vorschrift (TAV) verfasst werden und aus [VVK-EDI] referenziert werden.
Hinweis	Die anbieterunabhängige Schnittstelle könnte künftig als technische und administrative Vorschrift (TAV) verfasst werden und aus [VVK-EDI] referenziert werden.		

	<p>b) Die Middleware kann eine anbieterspezifische Schnittstelle definieren.</p> <p>Entscheidend hierbei ist, dass die Schnittstelle so weit wie möglich mit der jeweiligen Schnittstelle des zukünftigen Systems kompatibel ist. Dies ermöglicht eine einfache Migration auf zukünftige Versichertenkarten.</p>		
Applikationen	<p>Die Fachapplikationen sprechen die Funktionalität der Versichertenkarte über die Middleware an. Die Middleware sollten bis Mitte 2012 in die Applikationen integriert werden können.</p> <table border="1" data-bbox="528 752 1442 925"> <tr> <td style="background-color: #f8d7da;">Hinweis</td> <td>Die Fachapplikationen können Funktionalitäten der Versichertenkarten bis zur Einführung der zukünftigen Versichertenkarten nutzen. Abhängig von der Kompatibilität der Schnittstellen der zukünftigen Middleware bedarf es bei einer Migration auf die zukünftigen Versichertenkarten mehr oder weniger Anpassungen an der Fachapplikation.</td> </tr> </table>	Hinweis	Die Fachapplikationen können Funktionalitäten der Versichertenkarten bis zur Einführung der zukünftigen Versichertenkarten nutzen. Abhängig von der Kompatibilität der Schnittstellen der zukünftigen Middleware bedarf es bei einer Migration auf die zukünftigen Versichertenkarten mehr oder weniger Anpassungen an der Fachapplikation.
Hinweis	Die Fachapplikationen können Funktionalitäten der Versichertenkarten bis zur Einführung der zukünftigen Versichertenkarten nutzen. Abhängig von der Kompatibilität der Schnittstellen der zukünftigen Middleware bedarf es bei einer Migration auf die zukünftigen Versichertenkarten mehr oder weniger Anpassungen an der Fachapplikation.		

8.2.1.1 Anpassungen der geplanten Anpassung der VVK-EDI

Am 9. August 2010 hat das BAG in einem Schreiben⁴⁸ die geplanten Änderungen an der [VVK-EDI] erläutert, welche am 1. November 2010 in Kraft hätten treten sollen⁴⁹. Bei den geplanten Anpassungen handelte sich in erster Linie um technische Einzelheiten, welche bei der Umsetzung der Vorgaben festgestellt wurden und an das BAG gemeldet worden sind.

Grundsätzlich betreffen die Änderungen der [VVK-EDI] nur Dateninhalte, unabhängig von der technischen Struktur der Versichertenkarten. Es wäre aber vorab zu prüfen, ob die Daten unter den geänderten Vorgaben in den bisherigen Strukturen der Versichertenkarten abgelegt werden können. Da die [SPEC-POST] und die [SPEC-SASIS] transparente oder linear variable Datenfelder für die administrativen und medizinischen Daten definieren, ist zu untersuchen, ob für die neu definierten Daten ausreichend Speicherplatz auf der Karte zur Verfügung steht.

Die geplanten Änderungen haben Einfluss auf die Middleware, welche die Kompatibilität der verschiedenen Versichertenkarten herstellen kann. Die Softwarehersteller müssten entsprechende Änderungen an der jeweiligen Middleware vornehmen.

⁴⁸ [VVK-EDI-Änderung]

⁴⁹ Die Änderungen wurden bis Ende 2010 nicht umgesetzt.

8.2.2 Spezifikation des zukünftigen Systems

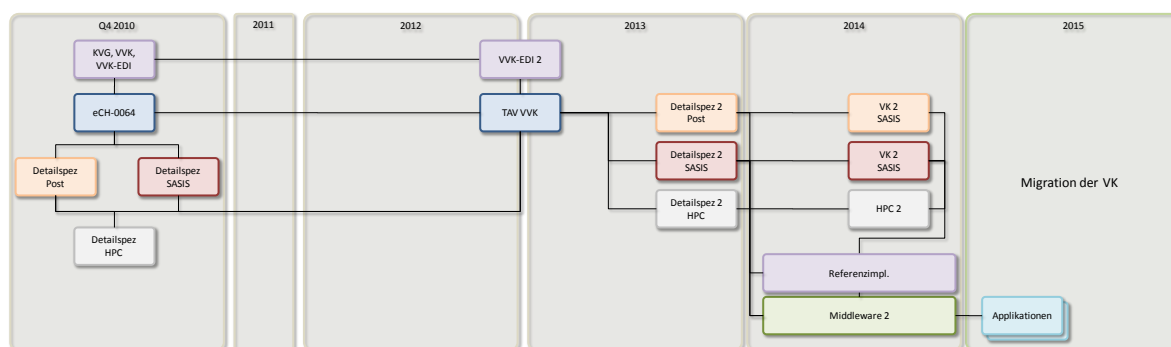
Die Spezifikation des zukünftigen Systems hat zum Ziel, die technische Spezifikation und Umsetzung des Systems zu vereinheitlichen. Das zukünftige System soll im Jahr 2015 bereitstehen, um so die bestehenden Versichertenkarten im ordentlichen Austauschverfahren ersetzen zu können.

Grundsätzlich kann das zukünftige System in zwei Varianten spezifiziert und umgesetzt werden.

8.2.2.1 Variante 1 – TAV VVK-EDI

In der Variante 1 werden die zukünftigen technischen Grundlagen überarbeitet und in der Form von technischen und administrativen Vorschriften (TAV VVK-EDI) als Anhang zu [VVK-EDI] verfasst. Der Umfang der Überarbeitung muss vom BAG in Absprache mit den Anbietern vorgegeben werden.

Folgend ist ein möglicher zeitlicher Ablauf der Spezifikation des zukünftigen Systems aufgezeigt:



Komponente	Beschreibung
[VVK] [VVK-EDI] [SPEC-POST] [SPEC-SASIS] [FMH-HPC]	Aktuell verfügbare Komponenten.
VVK-EDI 2	Verschiedene Datenstrukturen der [VVK-EDI] können aktualisiert oder erweitert werden. Idealerweise fließen bereits Erkenntnisse in VVK-EDI 2 ein, welche im Zusammenhang mit Anwendungen der Fachapplikationen des bestehenden Systems gewonnen werden konnten. Die überarbeitete Verordnung sollte bis Ende 2012 fertiggestellt sein.

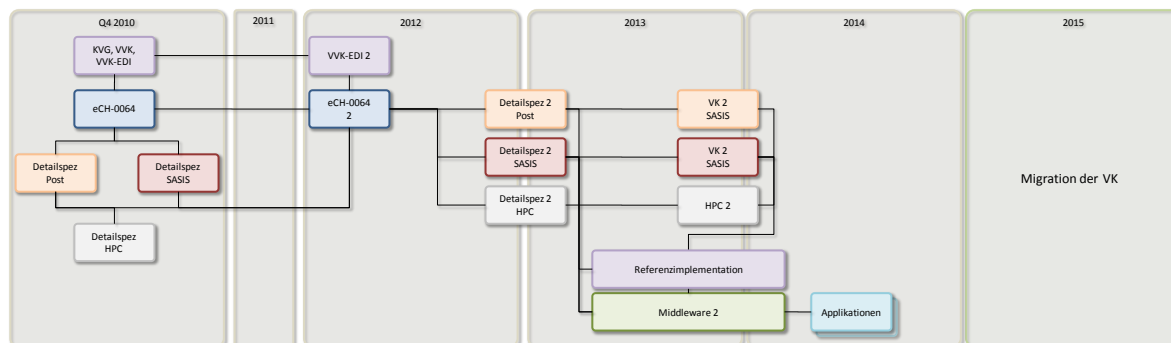
TAV VVK-EDI	Die technischen Vorgaben aus [eCH64] werden in enger Zusammenarbeit mit den jeweiligen Anbietern und Experten neu überarbeitet und als technische und administrative Vorschriften (TAV VVK-EDI) als Anhang zu [VVK-EDI] verfasst. Die neue TAV VVK-EDI sollte bis Ende 2012 fertiggestellt sein.		
Detailspez 2 <ul style="list-style-type: none"> ▪ SASIS ▪ Post ▪ HPC 	Die jeweiligen Detailspezifikationen müssen der neu erarbeiteten TAV VVK-EDI angepasst werden. Diese sollten bis Ende 2013 fertiggestellt werden können.		
VK 2 <ul style="list-style-type: none"> ▪ SASIS ▪ Post HPC 2	Die jeweiligen Umsetzungen der Versichertenkarten müssen den neuen Detailspezifikationen angepasst werden. Die Umsetzung sollte bis Mitte 2014 fertiggestellt werden können.		
Middleware 2	Die Middleware bildet die Schnittstelle zwischen der Applikation und den Versichertenkarten. Die Umsetzung sollte bis Anfangs 2014 fertiggestellt werden können.		
Referenzimpl.	Die Referenzimplementation stellt definierte Testprozesse bereit und sichert so die Interoperabilität der Versichertenkarten und der HPC. Bevor eine Versichertenkarte oder eine HPC ausgerollt werden darf, muss sie die Testprozeduren der Referenzimplementation bestehen. Die Umsetzung sollte bis Anfangs 2014 fertiggestellt werden können.		
Applikationen	Die Fachapplikationen sprechen die Funktionalität der Versichertenkarte über die Middleware an. Die Middleware sollten bis Mitte 2015 in die Applikationen integriert werden können. <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <table border="0"> <tr> <td style="background-color: #f8d7da; padding: 2px;">Hinweis</td> <td>Abhängig von der Kompatibilität der Schnittstellen zur bestehenden Middleware bedarf es bei einer Migration auf die zukünftigen Versichertenkarten mehr oder weniger Anpassungen an der Fachapplikation.</td> </tr> </table> </div>	Hinweis	Abhängig von der Kompatibilität der Schnittstellen zur bestehenden Middleware bedarf es bei einer Migration auf die zukünftigen Versichertenkarten mehr oder weniger Anpassungen an der Fachapplikation.
Hinweis	Abhängig von der Kompatibilität der Schnittstellen zur bestehenden Middleware bedarf es bei einer Migration auf die zukünftigen Versichertenkarten mehr oder weniger Anpassungen an der Fachapplikation.		

8.2.2.2 Variante 2 – eCH-0064 V2

In der Variante 2 wird der aktuelle [eCH64] überarbeitet. Im speziellen werden Fehler oder interpretationswürdige Passagen von [eCH64] in enger Zusammenarbeit mit den jeweiligen Anbietern und Experten aktualisiert und als neue Version des eCH-0064 Standards veröffentlicht. Das BAG muss die Fachgruppe des Vereins eCH neu einberufen und die Aktualisierung des Standards veranlassen⁵⁰. Der Umfang der Überarbeitung muss vom BAG in Absprache mit den Anbietern vorgegeben werden.

Folgend ist ein möglicher zeitlicher Ablauf der Spezifikation des zukünftigen Systems aufgezeigt:

⁵⁰ Der Verweis auf eCH-0064 in VVK-EDI muss ebenfalls aktualisiert werden.



Komponente	Beschreibung
[VVK] [VVK-EDI] [SPEC-POST] [SPEC-SASIS] [FMH-HPC]	Aktuell verfügbare Komponenten.
VVK-EDI 2	Verschiedene Datenstrukturen der [VVK-EDI] können aktualisiert oder erweitert werden. Idealerweise fließen bereits Erkenntnisse in VVK-EDI 2 ein, welche im Zusammenhang mit Anwendungen der Fachapplikationen des bestehenden Systems gewonnen werden konnten. Die überarbeitete Verordnung sollte bis Anfangs 2012 fertiggestellt sein.
eCH-0064 V2	Die technischen Vorgaben aus [eCH64] werden in enger Zusammenarbeit mit den jeweiligen Anbietern und Experten überarbeitet. Die Ergebnisse fließen in eine neue Version von eCH-0064. Die überarbeitete Version von eCH-0064 sollte bis Anfangs 2012 fertiggestellt sein.
Detailspez 2 <ul style="list-style-type: none"> SASIS Post HPC 	Die jeweiligen Detailspezifikationen müssen der neu erarbeiteten TAV VVK-EDI angepasst werden. Diese sollten bis Ende 2012 fertiggestellt werden können.
VK 2 <ul style="list-style-type: none"> SASIS Post HPC 2 	Die jeweiligen Umsetzungen der Versichertenkarten müssen den neuen Detailspezifikationen angepasst werden. Die Umsetzung sollte bis Mitte 2013 fertiggestellt werden können.
Middleware 2	Die Middleware bildet die Schnittstelle zwischen der Applikation und den Versichertenkarten. Die Umsetzung sollte bis Ende 2013 fertiggestellt werden können.

Referenzimpl.	<p>Die Referenzimplementation stellt definierte Testprozesse bereit und sichert so die Interoperabilität der Versichertenkarten und der HPC.</p> <p>Bevor eine Versichertenkarten oder eine HPC ausgerollt werden darf, muss sie die Testprozeduren der Referenzimplementation bestehen. Die Umsetzung sollte bis Ende 2013 fertiggestellt werden können.</p>		
Applikationen	<p>Die Fachapplikationen sprechen die Funktionalität der Versichertenkarte über die Middleware an. Die Middleware sollten bis Mitte 2014 in die Applikationen integriert werden können.</p> <table border="1" data-bbox="566 786 1466 907"> <tr> <td data-bbox="566 786 686 907">Hinweis</td> <td data-bbox="686 786 1466 907"> <p>Abhängig von der Kompatibilität der Schnittstellen zur bestehender Middleware bedarf es bei einer Migration auf die zukünftiger Versichertenkarten mehr oder weniger Anpassungen an der Fachapplikation.</p> </td> </tr> </table>	Hinweis	<p>Abhängig von der Kompatibilität der Schnittstellen zur bestehender Middleware bedarf es bei einer Migration auf die zukünftiger Versichertenkarten mehr oder weniger Anpassungen an der Fachapplikation.</p>
Hinweis	<p>Abhängig von der Kompatibilität der Schnittstellen zur bestehender Middleware bedarf es bei einer Migration auf die zukünftiger Versichertenkarten mehr oder weniger Anpassungen an der Fachapplikation.</p>		

9 Anhang 1: Hex-Dump der Versicherungskarten

9.1 VK-Post

Die VK-Post spezifischen Analysen wurden mit der Versicherungskarte von Hans-Peter Muster durchgeführt. Sie wird auf Anfrage von Keyon für spezifische Analysen ausgehändigt.





9.1.1 Hex-Dump

Folgend ist der Hex-Dump der spezifizierten und öffentlich auslesbaren Files der VK-Post aufgeführt.

```
Broadcom Corp Contacted SmartCard 0: 3bdb96ff8131fe458067041bb42a000a02810553
VK Post
SELECT response: ResponseAPDU: 2 bytes, SW=9000

EF.DIR 0x2f, 0x00
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ RECORD 1 response: ResponseAPDU: 12 bytes, SW=6282
00000000 61 08 4F 06 D7 56 83 21 05 00 a.O..V!..
READ RECORD 2 response: ResponseAPDU: 18 bytes, SW=6282
00000000 61 0E 4F 0C A0 00 00 00 63 50 4B 43 53 2D 31 35 a.O.....cPKCS-15
READ RECORD 3 response: ResponseAPDU: 17 bytes, SW=6282
00000000 61 0D 4F 0B F7 56 50 6F 73 74 4F 54 50 00 00 a.O..VPostOTP..
READ RECORD 4 response: ResponseAPDU: 2 bytes, SW=6a83

EF.ATR 0x2f, 0x01
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 36 bytes, SW=9000
00000000 E0 10 02 02 04 AF 02 02 04 AF 02 02 04 AF 02 02 .....
00000010 04 AF 66 0E 46 0C 04 44 45 47 2B 44 1B B4 02 00 ..f.F..DEG+D...
00000020 0A 02 ..

EF.VERSION 0x2f, 0x10
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 6 bytes, SW=9000
00000000 44 53 50 80 DSP.

EF.ICCSN 0x2f, 0x05
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ RECORD 1 response: ResponseAPDU: 14 bytes, SW=6282
00000000 5A 0A 80 75 69 99 99 99 99 99 99 94 Z..ui.....
READ RECORD 2 response: ResponseAPDU: 14 bytes, SW=6282
00000000 69 99 99 99 99 99 99 94 00 00 00 00 i.....
READ RECORD 3 response: ResponseAPDU: 14 bytes, SW=6282
00000000 20 10 01 12 10 13 42 5A 00 00 00 00 .....BZ....
READ RECORD 4 response: ResponseAPDU: 2 bytes, SW=6a83

EF.ID 0x2f, 0x06
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 62 bytes, SW=6282
00000000 65 3A 80 1C 30 1A 30 0C 0C 0A 48 61 6E 73 2D 50 e:...0...Hans-P
00000010 65 74 65 72 30 0A 30 08 0C 06 4D 75 73 74 65 72 eter0.0...Muster
00000020 82 08 31 39 36 35 30 37 32 32 83 0D 37 35 36 31 ..19650722..7561
00000030 32 33 34 35 36 37 38 39 30 84 01 01 234567890...
```

EF.AD 0x2f, 0x07

```
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 56 bytes, SW=6282
00000000 65 34 90 02 43 48 91 07 48 65 6C 73 61 6E 61 92 e4..CH..Helsana.
00000010 05 30 31 35 36 32 93 14 38 30 37 35 36 39 39 39 .01562..80756999
00000020 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 9999999999999999..20
00000030 31 33 30 33 33 31 130331
```

EF.PuK.CA_ROOT_VK 0x0e, 0x02

```
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 142 bytes, SW=9000
00000000 81 82 00 80 A4 4D 1D 1C A8 8A 36 DE 97 BF 12 64 .....M....6....d
00000010 1A CE 99 07 DB 18 09 33 98 36 E3 3D 86 81 2A B0 .....3.6.=.*.
00000020 73 6B BA E7 4E FC A0 15 AA 76 D9 7E 11 CB 9A 71 sk..N....v~...q
00000030 47 0C 91 26 A2 2E E1 B0 8F 79 13 4B A2 83 39 81 G..&.....y.K..9.
00000040 C1 C4 77 E1 48 9D 4E 43 00 38 D4 B2 FD 35 79 45 ..w.H.NC.8...5yE
00000050 43 23 9C F0 FA E4 57 D6 45 0C 00 28 A0 B0 9F D5 C#....W.E..(....
00000060 18 BD 14 ED 5F 78 B6 26 7B 2B FC 4F 14 F8 A5 6B ....x.&{+.O...k
00000070 5B 82 70 E6 5D F7 9B 05 A8 F9 AC D9 07 6C AF 64 [.p.].....l.d
00000080 B5 07 6A 7F 82 82 00 04 00 01 00 01 ..j].....
```

EF.CVC.PDC 0x2f, 0x03

```
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 219 bytes, SW=9000
00000000 7F 21 81 D5 5F 37 81 80 98 9F 00 BC 1D AC A7 F7 !..._7.....
00000010 69 6F C3 0E EF FB 4D 70 01 63 2A 64 15 A1 ED D5 io....Mp.c*d....
00000020 FF 5F EE 8A 3A 9D 97 8C 62 01 99 8C 71 D9 3B 38 ._.:..b...q.;8
00000030 CB 8B 56 20 BD 0B 31 B1 B0 0B 0D 08 0D 42 7D CC ..V ..l.....B).
00000040 BD F9 EC 9E 6A 0C 9D 1F 9B BC 43 C1 44 38 C4 3C .j....C.D8.<
00000050 20 87 14 ED FB 3B 5B 90 C1 C5 21 CC AD 50 67 13 .;[...!..Pg.
00000060 7E 78 4B 25 7A 6E 8E 38 21 74 C1 E8 A8 92 64 F2 ~xK%zn.8!t....d.
00000070 E7 5A A0 B2 AC 5C D7 07 DB 09 93 92 6C 29 76 28 .Z...Q.....l)v(
00000080 DC 03 3E 36 86 51 2C AE 5F 38 44 73 32 1A E4 FA .>6.Q,._8Ds2...
00000090 4A 58 47 2B 78 3B 84 45 3C A3 FE E1 A8 F6 45 7B JXG+x;.E<....E{
000000A0 98 AD 99 7C 2D B3 02 79 1E C3 E2 14 E0 F3 F9 FC ...|-.y.....
000000B0 51 D8 D1 46 5E 83 F6 FE 6B A9 65 1E 5A FD 12 5E Q..F^...k.e.Z..^
000000C0 1D 8B 7E 8D 08 A8 31 62 E9 09 33 00 01 00 01 42 ..~...lb..3....B
000000D0 08 43 48 44 53 50 60 01 09 .CHDSP`..
```

EF.CVC.CA_ROOT_VK (EF.CVC.CA_ORG_PDC) 0x2f, 0x04

```
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 219 bytes, SW=9000
00000000 7F 21 81 D5 5F 37 81 80 0A 53 24 7A 53 84 0F 45 !..._7...S$zS..E
00000010 29 16 BB BA DC EE D1 73 50 DA 71 D8 7F F5 AD 14 ).....sP.q....
00000020 D6 11 5D AF 7A 90 BB 95 30 C2 87 39 AF CF 5A 44 .].z...0..9..ZD
00000030 2B F7 F3 9A 3A 58 F5 36 22 27 C8 45 D8 F0 B1 47 +...:X.6"'.E...G
00000040 8F 11 31 B4 2F B2 00 19 A9 56 D3 EC 94 5E C6 49 .l./...V...^..I
00000050 B7 BD 12 DA 76 F1 40 4C 4B FB AF 78 CA 69 F0 40 ....v.@LK..x.i.@
00000060 ED 64 A2 5E C4 20 39 CE DA 20 89 FD 59 B7 CB 48 .d.^..9...Y..H
00000070 27 80 F0 6D D8 EB B8 30 09 60 68 1F EA 3A A7 F0 '.m...0.'h....
00000080 DE F0 70 68 46 BA AC 0A 5F 38 44 48 9D 4E 43 00 ..phF...8DH.NC.
00000090 38 D4 B2 FD 35 79 45 43 23 9C F0 FA E4 57 D6 45 8...5yEC#...W.E
000000A0 0C 00 28 A0 B0 9F D5 18 BD 14 ED 5F 78 B6 26 7B ..(.....x.&{
000000B0 2B FC 4F 14 F8 A5 6B 5B 82 70 E6 5D F7 9B 05 A8 +.O...k[p.]...
000000C0 F9 AC D9 07 6C AF 64 B5 07 6A 7F 00 01 00 01 42 ....l.d.].....B
000000D0 08 43 48 44 53 50 60 01 09 .CHDSP`..
```

9.2 VK-SASIS

Die VK-SASIS spezifischen Analysen wurden mit der Versicherungskarte von Markus Suter durchgeführt. Sie wird auf Anfrage von Keyon für spezifische Analysen ausgehändigt.



9.2.1 Hex-Dump

Folgend ist der Hex-Dump der spezifizierten und öffentlich auslesbaren Files der VK-Post aufgeführt.

```
Broadcom Corp Contacted SmartCard 0: 3b9f1381b180371f038031f8694d54434f5370020102810786
VK SASIS
SELECT response: ResponseAPDU: 2 bytes, SW=9000

EF.DIR 0x2f, 0x00
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 86 bytes, SW=6282
00000000 61 19 4F 06 D7 56 83 21 05 00 50 09 45 6D 65 72 a.O..V.!..P.Emer
00000010 67 65 6E 63 79 51 04 3F 00 DF 01 61 1D 4F 0C A0 gencyQ?...a.O..
00000020 00 00 00 63 50 4B 43 53 2D 31 35 50 07 50 4B 43 ...cPKCS-15P.PKC
00000030 53 2D 31 35 51 04 3F 00 DF 02 61 18 4F 0A F7 56 S-15Q?...a.O..V
00000040 83 21 05 4B 74 4D 56 00 50 04 4B 74 4D 56 51 04 .!.KtMV.P.KtMVQ.
00000050 3F 00 DF 03 ?...

EF.ATR 0x2f, 0x01
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 45 bytes, SW=9000
00000000 E0 10 02 02 04 00 02 02 04 00 02 02 04 00 02 02 .....
00000010 04 00 66 17 46 15 02 49 6E 74 65 72 31 20 20 20 ..f.F..Interl
00000020 4D 54 43 4F 53 20 70 20 32 2E 31 MTCOS p 2.1

EF.VERSION 0x56, 0x00
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 6 bytes, SW=9000
00000000 53 41 53 80 SAS.

EF.ICCSN 0x2f, 0x05
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ RECORD 1 response: ResponseAPDU: 14 bytes, SW=6282
00000000 5A 0A 80 75 60 12 34 00 00 00 20 85 z..u`.4...
READ RECORD 2 response: ResponseAPDU: 10 bytes, SW=6282
00000000 39 39 39 39 39 30 30 31 99999001
READ RECORD 3 response: ResponseAPDU: 15 bytes, SW=6282
00000000 32 30 30 39 30 31 30 31 30 30 30 30 5A 200901010000Z
READ RECORD 4 response: ResponseAPDU: 2 bytes, SW=6a83

EF.ID 0x2f, 0x06
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 86 bytes, SW=9000
00000000 65 2B 80 0D 53 75 74 65 72 2C 20 4D 61 72 6B 75 e+..Suter, Marku
00000010 73 82 08 32 30 31 30 30 31 30 31 83 0D 37 35 36 s..20100101..756
00000020 39 39 39 39 39 39 37 39 31 31 84 01 01 00 00 00 9999997911.....
00000030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000050 00 00 00 00 .....
```

```
EF.AD 0x2f, 0x07
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 97 bytes, SW=9000
00000000 65 31 90 02 43 48 91 04 54 45 53 54 92 05 30 31 e1..CH..TEST..01
00000010 32 33 34 93 14 38 30 37 35 36 30 31 32 33 34 30 234...80756012340
00000020 30 30 30 30 30 32 30 38 35 94 08 32 30 31 34 30 000002085..20140
00000030 36 33 30 00 00 00 00 00 00 00 00 00 00 00 00 00 630.....
00000040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

EF.CVC.PDC 0x2f, 0x03
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 620 bytes, SW=9000
00000000 7f 21 82 02 65 7f 4e 82 01 5b 5f 29 01 05 42 10 !...e.N.[ ]..B.
00000010 43 48 53 41 53 62 32 30 30 39 30 30 33 30 30 31 CHSASb2009003001
00000020 7f 49 82 01 12 06 07 60 85 74 05 22 02 01 81 82 .I.....t."....
00000030 01 00 bf 64 6d 4f b3 25 c6 06 60 0a 3f 3e 64 31 ...dmo.%..?>di
00000040 e9 4b 66 c3 2e 5a 11 cb 57 7d e3 26 85 29 70 22 .Kf..Z..W)&.)p"
00000050 25 1d 04 e8 5d 6f 6e 06 4c bb ac 91 fd a8 37 5b %...]on.L.....7[
00000060 ab a5 89 69 bd b7 93 23 ee cc f6 81 50 83 72 db ...i.#....P.r.
00000070 9f 9c 8a 9b f4 05 93 13 1a 1d df e1 65 6f 68 de .....eoh.
00000080 de 20 ca b3 a2 62 e6 96 da a9 06 86 98 ab ea c8 ...b.....
00000090 fa 31 85 e0 7c bf e9 4b a2 9f 2a 4a 57 ec 16 00 .l.|..K..*JW...
000000a0 81 d3 18 e9 ca 59 c9 77 22 09 81 e6 ae 18 a0 bc .....Y.w".....
000000b0 02 92 fa dc cd 15 a7 64 43 17 f3 4c 75 3a 16 38 .....dC..Lu:.8
000000c0 7e 96 c4 ad f8 8d 6f 11 a2 a6 b3 bb 85 0e fd 8c ~.....o.....
000000d0 ed 96 c7 b7 33 14 4c a5 c7 eb 01 59 82 f0 96 e1 ...3.L...Y....
000000e0 8e cd 38 86 18 ed 3a 76 7a 74 ca bf ac 78 2a ec ..8...:vzt...x*.
000000f0 60 4f c6 b9 92 52 45 6e 72 5d 55 b8 e9 ab 8b f1 `O...REnr]U....
00000100 05 1e fc aa 57 bd 1d c6 30 87 ae a2 77 19 e2 36 ...W...0...w..6
00000110 6b c2 18 52 bb 5c 1f c9 94 3b a0 5e 4b 33 2e 8f k..R.\...;^k3..
00000120 bc be 27 de e4 ee 9e e9 1f 78 0c 51 92 37 95 58 ...!.....x.Q.7.X
00000130 45 09 82 03 01 00 01 5f 20 0a 80 75 60 12 34 00 E....._..u'.4.
00000140 00 00 20 85 7f 4c 0c 06 07 60 85 74 05 22 01 01 ...L...t"...
00000150 53 01 00 5f 25 06 00 09 00 01 00 01 5f 24 06 01 S...%.....$.
00000160 04 00 06 03 00 5f 37 82 01 00 4b 25 f0 34 b2 0d ....7...K%.4..
00000170 dd 9d 2b 0e fd 8a 55 14 b0 39 04 82 b5 30 1e ac ..+...T.U..9...0..
00000180 84 f6 0f cf de c4 a7 63 6b a1 21 72 87 8d cb cd .....ck!r....
00000190 ff b3 61 51 06 a9 72 a7 31 59 32 0c b3 90 c1 ef ..aQ..r.lY2.....
000001a0 9d 09 a5 9c 65 01 d2 42 af ae 39 5d c0 9c 34 6a ....e..B..9]..4j
000001b0 5e 2e 92 44 cf 46 2b ce ce 34 6b 3b 2f 3e 47 ff ^...D.F+..4k;/>G.
000001c0 93 63 d4 a7 9a ad 3c 3f 08 4d 4f f6 26 c2 7a 57 .c....<?.MO.&.zW
000001d0 2e 9e 2b e1 c2 db 3a 05 fe 62 83 20 cb 8d 63 73 ..+...:..b. .cs
000001e0 9c b7 31 5d d4 28 c0 28 94 a6 61 4e 02 30 d5 34 ..1]..(..aN.0.4
000001f0 1c f7 05 7b fd 34 5e 5c 84 66 d6 4f a4 3d 79 d1 ...{.4^\.f.O.=y.
00000200 4a b9 be c7 00 04 70 f3 a9 59 6b 91 1f b9 42 03 J....p..Yk...B.
00000210 28 01 39 9e 46 13 7f 7c 13 29 46 74 92 80 ad f7 (.9.F..|.)Ft...
00000220 1d 37 e8 2e cb 08 e3 67 ac df 88 75 27 75 3f 84 .7.....g.....u'?.
00000230 9e d2 6b 0a 73 9d 46 14 61 6e b0 9e cd 07 b0 87 ..k.s.F.an.....
00000240 e2 95 7e cb 78 c6 bc 19 36 55 03 c2 e9 d2 d0 cc ..~.x...6U.....
00000250 ad 20 52 d0 94 b6 70 f8 8d db 9f 8f 5c ff 6e ee .R...p.....\n.
00000260 f8 9f 38 7f be e1 21 6e 38 90 ..8...!n8.
```

```
EF.CVC.CA_ORG_PDC 0x2f, 0x08
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 626 bytes, SW=9000
00000000 7F 21 82 02 6B 7F 4E 82 01 61 5F 29 01 04 42 10 .!.k.N.a.)..B.
00000010 43 48 52 56 4B 60 32 30 30 39 30 30 31 30 30 31 CHRVK`2009001001
00000020 7F 49 82 01 12 06 07 60 85 74 05 22 02 01 81 82 .I.....t."....
00000030 01 00 B4 8F 5D 36 DD A8 44 B6 80 42 B0 EB ED 0A ....]6..D..B....
00000040 94 E1 C7 75 95 81 39 40 74 C7 B3 B0 E4 4F B9 B6 ...u..9@t...O..
00000050 B2 71 2B C1 C1 DF 1F 70 4A EE 4B 02 09 F3 C6 53 .q+....pJ.K....S
00000060 85 9E 39 DE D2 15 F0 D3 36 C7 4A 1A 2C 78 AC DF ..9.....6.J.,x..
00000070 F5 35 38 95 D4 33 D1 8E 8E AD A3 7A E1 22 0D CD .58..3.....z"...
00000080 85 07 2D 45 1F DC 0D 88 EB 05 5A 02 E7 B0 BA 63 ..-E.....Z....c
00000090 72 5B 9C 08 3D C6 DF 28 D4 75 95 51 4B C3 C4 70 r[...=(.u.QK..p
000000A0 9C D2 AB 30 B7 A4 8C 51 21 80 41 E2 CD 7A 4F 3D ...0...Q!A..zO=
000000B0 48 84 60 6A 47 A8 BD 98 11 14 CA FE 82 31 FF 1A H.`jG.....1..
000000C0 BD 0B F1 A3 75 7D A8 0D 62 23 B7 A4 DD 76 35 FF ...u)..b#...v5.
000000D0 06 4F BA 19 E2 AB 3B 2E 36 15 FF A7 90 69 D7 E6 .O....;..6...i..
000000E0 92 9D 8E F8 BF C5 46 25 DC 78 39 66 2E C1 45 B9 .....F%.x9f..E.
000000F0 DD 8E 0F 69 C1 0D 3F E0 84 BC 3C A4 FF 4E BD 79 .....?..<..N.Y.
00000100 F8 3F BC 5C CA 76 26 77 17 0D E3 2E 3E 02 2D 57 .?.\..v&w....>..-W
00000110 F1 05 C8 33 83 C3 74 F9 99 7F C4 DA 91 16 A2 54 ...3..t.....T
00000120 71 1A 17 0E A5 EA 36 1C 5B DA E5 8E 15 DD 61 18 q.....6.[.....a.
00000130 98 3B 82 03 01 00 01 5F 20 10 43 48 53 41 53 62 .;.....CHSASb
00000140 32 30 30 39 30 30 33 30 30 31 7F 4C 0C 06 07 60 2009003001.L...`
00000150 85 74 05 22 01 01 53 01 00 5F 25 06 00 09 01 02 .t."..S...%.....
00000160 00 03 5F 24 06 01 07 01 02 00 02 5F 37 82 01 00 .._$......7....
00000170 98 5E F9 E3 1E 01 68 AE A2 CB B0 A1 08 CC 86 CE .^.....h.....
00000180 B3 A6 65 30 AF 1B D9 58 54 57 03 78 D9 6E 97 81 ..e0...XTW..x.n..
00000190 94 5A EA 19 F7 AE 38 14 D4 D8 09 08 57 FD 37 61 .Z.....8....W.7a
000001A0 52 68 43 7E 79 6F 3F C1 D6 B9 A5 EF 79 F7 AE 14 RhC~yo?....y...
000001B0 AA 16 BB 05 D0 51 E0 50 7D F1 1D 3F 68 47 83 34 ....Q.P].?hG.4
000001C0 15 E4 A3 69 F2 42 F6 F3 66 33 BF B2 0F 31 13 63 ...i..B...f3...1.c
000001D0 72 A8 30 D2 70 FA B0 D3 B8 AB 96 9F 6C 09 BE 92 r.o.p.....1...
000001E0 DC 70 C5 84 81 38 D8 FE 10 43 C5 4E 32 E6 70 1B .p...8...C.N2.p.
000001F0 F0 1F D8 02 B5 5C DB C1 19 C4 C3 87 F6 49 3B A6 .....\.I;
00000200 40 07 42 4A A9 DD 24 C6 D1 0D CE 89 4E D2 65 5D @.B.J.$.....N.e]
00000210 D7 38 F9 4C 63 C2 88 61 EC 87 B6 AF 22 CA 06 66 .8.Lc..a....".f
00000220 2F 2A 26 EE DA 95 64 7E 99 54 C4 52 E2 68 C0 CA /*&.d~T.R.h..
00000230 12 B9 C8 39 F9 4C FD 55 AB 9E 73 E2 51 E3 F5 C8 ...9.L.U..s.Q...
00000240 5F 4A A3 46 5A 96 D3 F7 93 70 BD C0 96 BC 07 F5 _J.FZ....p.....
00000250 61 AD C2 A1 F0 3A E8 AD 20 72 D6 91 FA F9 C0 45 a.....r.....E
00000260 72 A4 AF 56 61 2B 26 5B 11 B0 08 B7 80 A1 0B CB r..Va+&[.....
```

```
EF.CVC.CA_ROOT_VK 0x2E, 0x04
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 626 bytes, SW=9000
00000000 7F 21 82 02 6B 7F 4E 82 01 61 5F 29 01 01 42 10 .!.k.N.a.)..B.
00000010 43 48 52 56 4B 60 32 30 30 39 30 30 31 30 30 31 CHRVK`2009001001
00000020 7F 49 82 01 12 06 07 60 85 74 05 22 02 01 81 82 .I.....t."....
00000030 01 00 C4 40 0A F4 CE 1B 21 20 FE 3F C6 93 CF 81 ...@.....!..?....
00000040 4A 73 F4 3B 6D 70 DE 29 3D 56 9F B8 32 3F A8 28 Js.;mp.)=V..2?.(
00000050 31 2D 61 23 6E 9F 67 FC 35 87 28 21 F3 79 B1 63 1 -a#n.g.5.(!.y.c
00000060 C1 63 7F 1D 42 98 77 99 26 6D 6D 4F D1 57 EE 84 .c..b.w.&mmO.W..
00000070 3C 65 3F 6F 65 6A D7 51 7C D7 BB 6E 8F E8 AA D6 <e?oej.Q].n....
00000080 5F EB 41 B9 BF 55 60 52 34 9C 7B 8C 18 9E 48 E4 _A..U`R4.[...H.
00000090 36 AF DB D4 81 05 FB CA EB 67 D9 52 E6 81 A1 E6 6.....g.R....
000000A0 DD 5A B3 55 E2 D0 35 BE 39 79 33 92 84 8D 39 E7 .Z.U..5.9y3...9.
000000B0 BF 2F D9 03 53 DC 46 62 80 4E 62 55 77 38 03 97 ./...S.Fb.NbUw8..
000000C0 28 C1 D7 3D B1 4D 23 38 E5 59 4D 50 61 D2 56 00 (.=.M#8.YMPa.V.
000000D0 36 86 5E B4 04 EF 9A 78 13 49 23 48 22 31 93 4F 6.^.....x.I#H"1.O
000000E0 4A 5E E7 21 D9 19 0F CB 4E 3D E4 45 23 3F 93 87 J^!....N=-E#?..
000000F0 3F F2 F9 48 3D 27 13 FB AB 80 34 E4 BC 5C 2D CF ?..H="....4..\.
00000100 E7 ED 60 3A D1 FC 1F E5 C4 9C 8E 82 00 7F 6B 6F ..":.....3.....ko
00000110 8D 98 9B 9B B9 CF 10 33 A5 CF 5D 78 1C 41 97 8B .....3...]x.A..
00000120 20 CB B1 D9 F3 ED 2F 13 56 71 56 9A 0A 9E 61 61 ...../.VqV...aa
00000130 4B F7 82 03 01 00 01 5F 20 10 43 48 52 56 4B 60 K....._CHRVK`
00000140 32 30 30 39 30 30 31 30 30 31 7F 4C 0C 06 07 60 2009001001.L...`
00000150 85 74 05 22 01 01 53 01 00 5F 25 06 00 09 01 02 .t."..S...%.....
00000160 00 03 5F 24 06 01 07 01 02 00 03 5F 37 82 01 00 .._$......7....
00000170 2D DD B6 C3 D0 D3 8C 98 2C BC 99 5E 9D 66 28 ED -.....;.....^..f(.
00000180 8B 09 FE 86 8E 72 32 13 B2 4F 3C C4 41 5F 7B 71 .....r2..0<.A(q
00000190 1A 4D E3 9B 53 EC BC 4D 6D 7C 14 1C BB 52 23 82 .M..S..Mm|...R#.
000001A0 AA 61 3C E8 7C 00 78 94 8D 89 80 7C AF C6 9A D6 .a<|.x....|....
000001B0 7A 9B 02 3E 21 C4 90 D7 07 5D 30 36 3B 7E E1 A0 z..>!.....]06;~..
000001C0 D2 19 FF 26 49 4A 79 D1 F9 1F CD A8 F6 06 1F D5 ...&IjY.....
000001D0 72 17 4F 89 89 3B F9 BC DF 09 EE 12 C4 6A 1B 9B r.O.;.....j..
000001E0 B6 A1 80 03 4F 6B 54 C1 39 12 22 96 98 8A 3C A6 ...OkT.9".<..
000001F0 60 52 CC 81 E1 C9 47 3B 27 96 3A 67 DA 8B 61 96 `R...G;'.!g.a.
00000200 AE 02 31 90 44 A7 91 84 D6 D8 00 AD 47 9A 4D AD ..1.D.....G.M.
00000210 48 61 20 5F 6A C1 B7 BD 2D 20 C0 38 04 DA 23 CF Ha_j...-..8..#.
00000220 45 F1 E3 EF 94 D6 19 2F 80 08 A6 CA 87 D6 13 61 E...../.....a.
00000230 22 01 7B D0 72 B5 8F AE AC 92 23 C3 31 BA 01 F2 ".{r.....#.1...
00000240 6D BC E6 51 9D 87 13 23 95 73 90 24 37 FC A1 CA m..Q...#.s.$7...
00000250 F6 C2 3F 0B 0D 26 75 0E D0 ED 07 EF C0 70 95 AB ..?..&u.....p..
00000260 04 70 48 D3 99 F6 CA 78 9F 1C 79 0C 63 A4 03 45 .pH.....x..y.c..E
```



```

EF.PuK.CA_ROOT_VK 0x00, 0x1c
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ RECORD response: ResponseAPDU: 301 bytes, SW=9000
00000000 10 43 48 52 56 4B 60 32 30 30 39 30 30 31 30 30 .CHRVK`200900100
00000010 31 00 00 01 00 00 09 01 02 00 03 01 07 01 02 00 1.....
00000020 03 56 55 01 00 C4 40 0A F4 CE 1B 21 20 FE 3F C6 .VU...@...! .?.
00000030 93 CF 81 4A 73 F4 3B 6D 70 DE 29 3D 56 9F B8 32 ...Js.;mp.)=V..2
00000040 3F A8 28 31 2D 61 23 6E 9F 67 FC 35 87 28 21 F3 ?. (1-a#n.g.5. (!
00000050 79 B1 63 C1 63 7F 1D 42 98 77 99 26 6D 6D 4F D1 y.c.c..B.w.&mmO.
00000060 57 EE 84 3C 65 3F 6F 65 6A D7 51 7C D7 BB 6E 8F W...<e?oej.Q|..n.
00000070 E8 AA D6 5F EB 41 B9 BF 55 60 52 34 9C 7B 8C 18 ..._.A..U`R4.(.
00000080 9E 48 E4 36 AF DB D4 81 05 FB CA EB 67 D9 52 E6 .H.6.....g.R.
00000090 81 A1 E6 DD 5A B3 55 B2 D0 35 BE 39 79 33 92 84 ....Z.U..5.y3..
000000A0 8D 39 E7 EF 2F D9 03 53 DC 46 62 80 4E 62 55 77 .9../.S.Fb.NbUw
000000B0 38 03 97 28 C1 D7 3D B1 4D 23 38 E5 59 4D 50 61 8..(..=..M#8.YMPa
000000C0 D2 56 00 36 86 5E B4 04 EF 9A 78 13 49 23 48 22 .V.6.^.....x.I#H"
000000D0 31 93 4F 4A 5E E7 21 D9 19 0F CB 4E 3D E4 45 23 1.OJ^!.....N=.E#
000000E0 3F 93 87 3F F2 F9 48 3D 27 13 FB AB 80 34 E4 BC ?..?..H="....4..
000000F0 5C 2D CF E7 ED 60 3A D1 FC 1F E5 C4 9C 8E 82 00 \-...`.....
00000100 7F 6B 6F 8D 98 9B 9B B9 CF 10 33 A5 CF 5D 78 1C .ko.....3..]x.
00000110 41 97 8B 20 CB B1 D9 F3 ED 2F 13 56 71 56 9A 0A A.. ..../.VqV..
00000120 9E 61 61 4B F7 00 01 00 01 FA 46 .aaK.....F
SELECT response: ResponseAPDU: 2 bytes, SW=9000

PKCS#15::EF.CIAInfo 0x50, 0x32
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 66 bytes, SW=6282
00000000 30 3D 02 01 00 04 0A 80 75 60 12 34 00 00 00 20 0=.....u`.4...
00000010 85 0C 10 53 41 53 49 53 2D 49 6E 74 65 72 63 61 ...SASIS-Interca
00000020 72 64 20 A0 16 0C 14 38 30 37 35 36 30 31 32 33 rd ....807560123
00000030 34 30 30 30 30 30 32 30 38 35 03 02 04 60 00 40000002085....`

PKCS#15::EF.GPKeys 0x1f, 0x01
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 186 bytes, SW=6282
00000000 30 3A 30 1B 0C 12 53 69 67 6E 61 74 75 72 65 20 0:0...Signature
00000010 4B 65 79 20 4B 74 4D 56 03 02 06 80 04 01 02 30 Key KtmV.....0
00000020 0B 04 01 45 03 03 06 30 40 02 01 81 A1 0E 30 0C ...E...0@.....0.
00000030 30 06 04 04 3F 00 DF 03 02 02 08 00 30 35 30 16 0...?.....050.
00000040 0C 0D 53 69 67 6E 61 74 75 72 65 20 4B 65 79 03 ..Signature Key.
00000050 02 06 80 04 01 02 30 0B 04 01 46 03 03 06 30 40 .....0...F...0@
00000060 02 01 97 A1 0E 30 0C 30 06 04 04 3F 00 DF 02 02 ....0.0...?....
00000070 02 08 00 30 39 30 1A 0C 11 41 75 74 68 6F 72 69 ...090...Authori
00000080 7A 61 74 69 6F 6E 20 4B 65 79 03 02 06 80 04 01 zation Key.....
00000090 02 30 0B 04 01 47 03 03 06 74 40 02 01 96 A1 0E .0...G...t@.....
000000A0 30 0C 30 06 04 04 3F 00 DF 02 02 02 08 00 00 00 0.0...?.....
000000B0 00 00 00 00 00 00 00 00 .....

PKCS#15::EF.CD 0x1f, 0x03
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 254 bytes, SW=9000
00000000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
...
000000F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

PKCS#15::EF.DCOD 0x1f, 0x04
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 102 bytes, SW=9000
00000000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
...
00000060 00 00 00 00 .....

PKCS#15::EF.AOD 0x1f, 0x05
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 93 bytes, SW=6282
00000000 30 2F 30 0D 0C 04 50 49 4E 32 03 02 06 C0 04 01 0/0...PIN2.....
00000010 04 30 03 04 01 02 A1 19 30 17 03 03 04 0C 00 0A .0.....0.....
00000020 01 01 02 01 06 02 01 08 02 01 08 80 01 02 04 01 .....
00000030 00 30 28 30 09 0C 03 50 55 4B 03 02 06 80 30 03 .0(0...PUK....0.
00000040 04 01 04 A1 16 30 14 03 03 04 3E 00 0A 01 01 02 .....0.....>.....
00000050 01 08 02 01 08 02 01 08 80 01 04 .....

PKCS#15::EF.CERT 0x1f, 0x06
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 2057 bytes, SW=9000
00000000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
...
00000800 00 00 00 00 00 00 00 .....

```



```

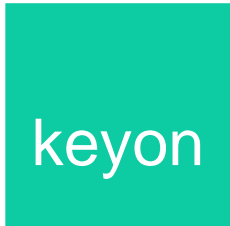
PKCS#15:EF.PuK.X509 0x1f, 0x08
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 272 bytes, SW=9000
00000000 30 82 01 0A 02 82 01 01 00 A3 67 C3 FD C4 A5 15 0.....g....
00000010 B4 74 6E 6E 45 37 26 6E 38 33 45 EE 05 93 F6 33 .tnnE7&n83E....3
00000020 A0 AE E0 28 04 3C C6 00 A7 47 C4 4D 1F 6A D3 35 ...(<...G.M.j.5
00000030 EE AC C6 CF 08 F7 87 C2 37 13 86 23 D0 92 4A 0C .....7.#...J.
00000040 40 67 64 FE F4 A9 02 0F BE 65 4A 46 A1 64 12 FB @gd.....eJF.d..
00000050 D9 35 2F EB 6F B9 53 98 7E C1 9F 4F 99 11 74 E0 .5/.o.s~..O..t.
00000060 9D 0A 5C DF BC 39 4E 59 57 0A 7C 2B 67 05 9A 16 ..\..9NYW.|+g...
00000070 B1 64 A3 4E 70 34 53 98 09 DC 02 DB A2 E1 A6 C0 .d.Np4S.....
00000080 83 CF 0D F9 BC E1 DD 38 B4 5C 52 99 2E B8 45 7A .....8.\R...Ez
00000090 61 5A 04 08 55 B7 F1 33 74 BF 43 E4 64 00 AB E4 az..U..3t.C.d...
000000A0 89 66 02 37 5D D8 F7 42 7E 37 1E 93 32 CE D7 7E .f.7]..B~7..2..~
000000B0 2C 5B 83 F1 58 45 F0 6A 61 06 75 BB 10 38 F7 3C ,[.XE.ja.u..8.<
000000C0 73 38 FD F6 61 47 19 72 41 19 67 64 75 ED 1D 18 s8..aG.rA.gdu...
000000D0 21 DA 3D EF B6 FD BD EF 06 8C 69 81 3E 5E 6C 67 !.=.....i.>^lg
000000E0 E2 D2 CD EB B1 1E A2 AF 81 3F E1 E3 A8 8A 48 E0 .....?....H.
000000F0 3A BD 19 D4 FD 2F 9A 58 FD F2 F0 8B 38 4B EC F4 :..../.X....8K..
00000100 1A 35 71 6F 03 B0 32 83 87 02 03 01 00 01 .5qo..2.....

```

```

PKCS#15:EF.PuK.DEC 0x1f, 0x07
SELECT response: ResponseAPDU: 2 bytes, SW=9000
READ BINARY response: ResponseAPDU: 272 bytes, SW=9000
00000000 30 82 01 0A 02 82 01 01 00 BC FE 20 EE 73 91 96 0..... .s..
00000010 63 77 AD 66 68 A4 91 24 45 0A 80 30 3A B3 53 26 cw.fh...$E..0:.$&
00000020 A1 10 19 8A B0 75 1A 22 07 E5 1A B0 C4 F6 9B 8F .....u.".....
00000030 29 FE 06 E1 EE E4 D5 77 DD BA E7 FB FC 2A 29 A9 ).....w.....*).
00000040 66 D0 4C 8C ED 90 E7 84 31 31 DF BF CA 40 48 95 f.L.....11...@H.
00000050 FB 65 02 CC 58 F7 44 EC C0 A5 22 FF 75 61 50 E5 .e.X.D..."uaP.
00000060 FC 92 54 D9 A7 24 A3 B3 36 81 9D FE 5A 38 DA 43 ..T..$.6..Z8.C
00000070 4F C3 C1 C0 B6 76 85 2D 0A 43 47 DE 24 64 A6 63 O....v.-.CG.$d.c
00000080 07 99 FE C4 F6 52 FC 21 68 68 89 05 3C A3 11 B3 .....R.!hh.<...
00000090 B3 2D A7 09 83 D0 C8 21 54 A0 30 0D C7 A0 5F 49 -.!!!T.O...I
000000A0 D0 2D BC C7 DA 24 8B 21 B9 42 40 C1 0C 3B AA 46 .-...$.!.B@...;F
000000B0 0E 78 15 DE DF 38 19 E5 BD 88 42 84 F0 D0 36 FC .x...8....B...6.
000000C0 F1 E7 2E 72 BD 62 91 AB 79 62 73 17 DB F2 D7 AC ...r.b...ybs....
000000D0 DF 98 30 DE 16 95 A8 AB C3 30 71 D8 D6 AA 72 25 .0.....0q...r%
000000E0 8C 90 9C 62 AD 9F 3A B0 7B BD 4E F7 1C 81 4C 11 ...b...{.N...L.
000000F0 CF 36 08 25 94 F2 7C F2 99 93 E5 0C 42 33 90 76 .6.%...|....B3.v
00000100 1C 70 BD 7A AB 46 94 D7 D9 02 03 01 00 01 .p.z.F.....

```



10 Anhang 2: Analyse Dateisystem

Folgend sind die EF spezifischen Angaben aus der [SPEC-POST] und [SPEC-SASIS] gegenübergestellt.

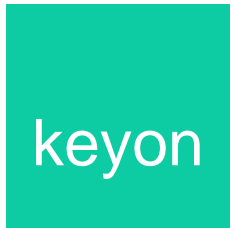
10.1 Definitionen

Bezeichnung	Beschreibung	
Bewertungen	Geringfügige Abweichung	Abweichung

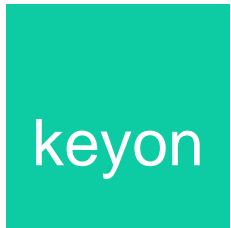
10.2 Dateisystem [eCH64]

10.2.1 EF.DIR

[eCH64]::Seite 20		[VVK-EDI]::				
	[eCH64]	[POST]		[SASIS]		
Struktur	Linear variable	Linear variable		Transparent		
Type	Record					
	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R		Read Record	Always	SELECT	Always
			Update Record	SK.Admin && SM.MAC	READ BINARY	Always
			Erase Record	SK.Admin && SM.MAC		
			Append Record	SK.Admin && SM.MAC		
			Delete EF	Never		
FID		2F00		2F00		
SFID		1E		F0		
Anz. Records	3	Max. 10 Records à max. 36 Bytes				
Record Grösse		200 Bytes		84 Bytes		
	Wert	Länge	Wert	Länge	Wert	Länge
Record 1		[POST]::Seite 7, Kapitel 2.4		[SASIS]::Seite8, Kapitel 4.2		

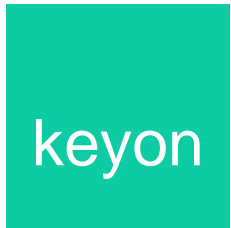


Record 2		Folgende Records werden während der Produktion der Karte angelegt. Record 1(DF.Not): 61 08 4F 06 D7 56 83 21 05 00 Record 2(DF.KtmV): 61 0E 4F 0C A0 00 00 00 63 50 4B 43 53 2D 31 35	<table border="1"> <thead> <tr> <th>Data-Object</th> <th>EF.DIR</th> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td></td> <td>L</td> <td>61-L-(4F-L-V-50-L-V-51-L-V)</td> <td></td> </tr> <tr> <td>61</td> <td>25</td> <td>Variable [V]</td> <td></td> </tr> <tr> <td>4F</td> <td>6</td> <td>D7 56 83 21 05 00</td> <td>National AID: DF NOT</td> </tr> <tr> <td>50</td> <td>9</td> <td>Emergency</td> <td>Application Label</td> </tr> <tr> <td>51</td> <td>4</td> <td>3F 00 DF 01</td> <td>Ref. to DF</td> </tr> <tr> <td></td> <td>L</td> <td>61-L-(4F-L-V-50-L-V-51-L-V)</td> <td></td> </tr> <tr> <td>61</td> <td>29</td> <td>Variable [V]</td> <td></td> </tr> <tr> <td>4F</td> <td>12</td> <td>A0 00 00 00 63 50 4B 43 53 2D 31 35</td> <td>AID PKCS#15</td> </tr> <tr> <td>50</td> <td>7</td> <td>PKCS-15</td> <td>Application Label</td> </tr> <tr> <td>51</td> <td>4</td> <td>3F 00 DF 02</td> <td>Ref. to DF</td> </tr> </tbody> </table>		Data-Object	EF.DIR				L	61-L-(4F-L-V-50-L-V-51-L-V)		61	25	Variable [V]		4F	6	D7 56 83 21 05 00	National AID: DF NOT	50	9	Emergency	Application Label	51	4	3F 00 DF 01	Ref. to DF		L	61-L-(4F-L-V-50-L-V-51-L-V)		61	29	Variable [V]		4F	12	A0 00 00 00 63 50 4B 43 53 2D 31 35	AID PKCS#15	50	7	PKCS-15	Application Label	51	4	3F 00 DF 02	Ref. to DF
Data-Object	EF.DIR																																															
	L	61-L-(4F-L-V-50-L-V-51-L-V)																																														
61	25	Variable [V]																																														
4F	6	D7 56 83 21 05 00	National AID: DF NOT																																													
50	9	Emergency	Application Label																																													
51	4	3F 00 DF 01	Ref. to DF																																													
	L	61-L-(4F-L-V-50-L-V-51-L-V)																																														
61	29	Variable [V]																																														
4F	12	A0 00 00 00 63 50 4B 43 53 2D 31 35	AID PKCS#15																																													
50	7	PKCS-15	Application Label																																													
51	4	3F 00 DF 02	Ref. to DF																																													
Kommentar	1.																																															



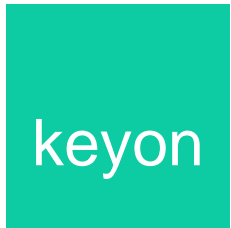
10.2.2 EF.ICCSN

[eCH64]::Seite 20		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Struktur	Linear variable	Linear Fixed		Linear variable		
Type	Record					
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R		Read Binary	Always	SELECT	Always
			Update Binary	Never	READ RECORD	Always
			Erase Binary	Never		
			Delete EF	Never		
FID		2F05		2F05		
SFID		05		05		
Anz. Records	3	3 Records à 12 Bytes		3		
Record Grösse		36 Bytes		45 Bytes		
	Wert	Länge	Wert	Länge	Wert	Länge
Record 1	DO ICCSN		ICCSN	10 Bytes	ICCSN	10 Oktetts
Record 2	Reference Number	8 Byte	Reference Number	8 Bytes	Reference Number	8 Oktetts
Record 3	generalTime	15BCD	generalTime	8 Bytes	generalTime	13 Oktetts
Kommentar	1.	Wie wird das „Z“ BCD kodiert? „YYYYMMDDHHMMSS“ benötigt 7 Bytes. Es wird angenommen, dass das „Z“ im 8. Byte als UTF8 Charakter kodiert wird.				



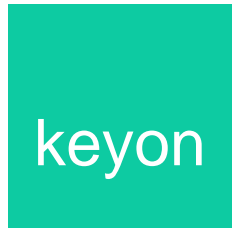
10.2.3 EF.ID

[eCH64]::Seite 21				[VVK-EDI]:: Kap. 2.1						
		[eCH64]		[POST]			[SASIS]			
Struktur	Transparent			Transparent			Transparent			
Type	BER-TLV			TLV			TLV			
Kategorie	Working						Working			
AM	AM	SC		AM	SC		AM	SC		
	R			Read Binary	Always		SELECT	Always		
				Update Binary	SK.Admin && SM.MAC		READ BINARY	Always		
				Erase Binary	SK.Admin && SM.MAC		GET DATA	Always		
			Delete EF	Never						
FID				2F06			2F06			
SFID				06			06			
Grösse				95 Byte			84 Byte			
	TAG	Länge	Optional	TAG	Länge	Optional	TAG	Länge	Optional	
cardholderRelated-Template	65			65	85		65			
Name (2.1.1)	80			80	63	Nein	80	50	Nein	
Geburtstag (2.1.2)	82		Nein	82	8	Nein	82	8	Nein	
Versichertennummer (2.1.3)	83		Nein	83	13	Nein	83	13	Nein	
Geschlecht (2.1.4)	84		Nein	84	1	Nein	84	1	Nein	
Kommentar	1.				In Tag 80 (Name) ist Name und Vorname in einer ASN.1 Struktur kodiert			In Tag 80 (Name) ist Name und Vorname durch Komma und Leerzeichen getrennt.		

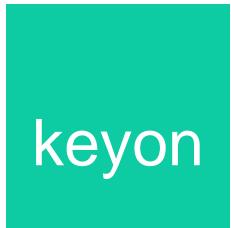


10.2.4 EF.AD

[eCH64]::Seite 21				[VVK-EDI]:: Kap. 2.2					
[eCH64]		[POST]		[SASIS]					
Struktur	Transparent			Transparent			Transparent		
Type	BER-TLV			TLV			TLV		
Kategorie	Working						Working		
AM	AM	SC		AM	SC		AM	SC	
	R			Read Binary	Always		SELECT	Always	
				Update Binary	SK.Admin && SM.MAC		READ BINARY	Always	
				Erase Binary	SK.Admin && SM.MAC		GET DATA	Always	
			Delete EF	Never					
FID				2F07			2F07		
SFID				07			07		
Grösse				1661 Byte			95 Byte		
	TAG	Länge	Optional	TAG	Länge	Optional	TAG	Länge	Optional
cardholderRelated-Template	65			65	95-1661		65		
Identifikation des ausstellenden Staates (2.2.1)	90		Nein	90	2	Nein	90	2	Nein
Name des Versicherers (2.2.2)	91		Nein	91	45	Nein	91	50	Nein
BAG-Nummer (2.2.3)	92		Nein	92	10	Nein	92	10	Nein
Kennnummer der Versichertenkarte (2.2.4)	93		Nein	93	30	Nein	93	20	Nein
Ablaufdatum (2.2.5)	94		Nein	94	8	Nein	94	8	Nein

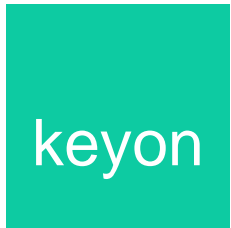


Zustelladresse (2.2.6.1)	09		Ja	09	576	Ja				
Angaben zur oblig. Kranken- pflegeversi. (2.2.6.2)	10		Ja	10	58	Ja				
Angaben zum OKP-Versicherer (2.2.6.3)	16		Ja	16	493	Ja				
VVG Informationen (2.2.6.4)	34		Ja	34	627	Ja				
Kommentar	1.							Es sind nur die Pflicht-Tags implementiert, die optionalen Tags haben keinen Platz.		



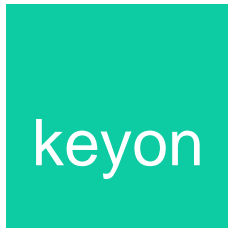
10.2.5 EF.LOG

[eCH64]::Seite 21		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Struktur	Linear cyclic	Cyclic				
Type	Record	Record				
Kategorie	Working					
AM	AM	SC	AM	SC	AM	SC
	R		Read Record	Always		
	UP		Append Record	Always		
			Update Record	Never		
			Delete EF	Never		
FID		2F09				
SFID		09				
Anz. Records		20				
Record Grösse		70 Bytes				
Grösse		1400 Bytes				
Kommentar	1.			Nicht implementiert		



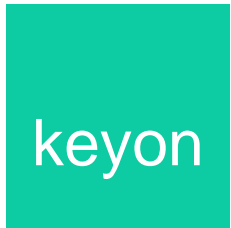
10.2.6 EF.BGTD

[eCH64]::Seite 24		[VVK-EDI]:: Kap. 2.6				
[eCH64]		[POST]		[SASIS]		
Struktur	Transparent	Transparent		Transparent		
Type	BER-TLV	TLV		TLV		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R	EF.ARR ₂ (Schlüssel_1 + PIN)	Read Binary	Schlüssel_1 + PIN	SELECT READ BINARY GET DATA	Always Schlüssel_1 + Pin 1
	UP	EF.ARR ₂ (Schlüssel_2 + PIN)	Update Binary	Schlüssel_2 + PIN	UPDATE RECORD APPEND RECORD	Schlüssel_2 + Pin 1 Schlüssel_2 + Pin 1
	W	EF.ARR ₂ (Schlüssel_2 + PIN)	Erase Binary	Schlüssel_2 + PIN		
					ENABLE VERIFICATION REQUIREMENT	Always
					DISABLE VERIFICATION REQUIREMENT	Always
FID		1F01		1F01		
SFID		--		01		
Anz. Records		1		1		
Record Grösse		302 Bytes		300 Bytes		
Grösse		302 Byte		302 Byte		
Kommentar	1.					



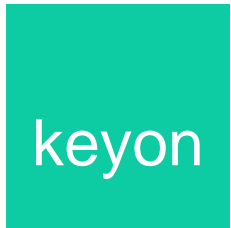
10.2.7 EF.IMMD

[eCH64]::Seite 24		[VVK-EDI]:: Kap. 2.5				
[eCH64]		[POST]		[SASIS]		
Struktur	Linear variable	Linear variable		Linear variable		
Type	Record (Simple-TLV)	Record		Record		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R	EF.ARR ₂ (Schlüssel_1 + PIN)	Read	Schlüssel_1 + PIN	SELECT READ RECORD	Always Schlüssel_1 + Pin 1
	UP	EF.ARR ₂ (Schlüssel_2 + PIN)	Update	Schlüssel_2 + PIN	UPDATE RECORD APPEND RECORD	Schlüssel_2 + Pin 1 Schlüssel_2 + Pin 1
	W	EF.ARR ₂ (Schlüssel_2 + PIN)	Erase Record	Schlüssel_2 + PIN		
					ENABLE VERIFICATION REQUIREMENT	Always
					DISABLE VERIFICATION REQUIREMENT	Always
FID		1F02		1F02		
SFID		--		02		
Anz. Records	[1..50]	max. 10		max. 21		
Record Grösse		264 Bytes		263 Bytes		
Grösse		2640 Byte		5628 Byte		
Kommentar	1.			.		



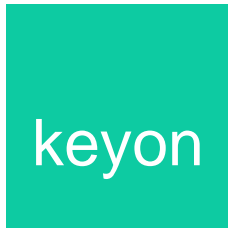
10.2.8 EF.TPLD

[eCH64]::Seite 25		[VVK-EDI]:: Kap. 2.2				
[eCH64]		[POST]		[SASIS]		
Struktur	Linear variable	Linear variable		Linear variable		
Type	Record (Simple-TLV)	Record		Record		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R	EF.ARR ₂ (Schlüssel_1 + PIN)	Read	Schlüssel_1 + PIN	SELECT READ RECORD	Always Schlüssel_1 + Pin 1
	UP	EF.ARR ₂ (Schlüssel_2 + PIN)	Update	Schlüssel_2 + PIN	UPDATE RECORD APPEND RECORD	Schlüssel_2 + Pin 1 Schlüssel_2 + Pin 1
	W	EF.ARR ₂ (Schlüssel_2 + PIN)	Erase Record	Schlüssel_2 + PIN		
					ENABLE VERIFICATION REQUIREMENT	Always
					DISABLE VERIFICATION REQUIREMENT	Always
FID		1F03		1F03		
SFID		--		03		
Anz. Records Tag 300	10			2		
Anz. Records Tag 400	10	6		1		
Record Grösse Tag 300				132 Bytes		
Record Grösse Tag 400		191 Bytes		190 Bytes		
Grösse		1146 Byte		333 Byte		
Kommentar	1.	Spezifikation sagt nicht, welcher Record von [VVK-EDI] implementiert wurde		.		



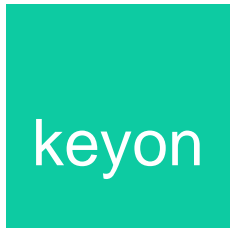
10.2.9 EF.KHUF

[eCH64]::Seite 25		[VVK-EDI]:: Kap. 2.1				
[eCH64]		[POST]		[SASIS]		
Struktur	Linear variable	Linear variable		Linear variable		
Type	Record (Simple-TLV)	Record		Record		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R	EF.ARR ₂ (Schlüssel_1 + PIN)	Read	Schlüssel_1 + PIN	SELECT READ RECORD	Always Schlüssel_1 + Pin 1
	UP	EF.ARR ₂ (Schlüssel_2 + PIN)	Update	Schlüssel_2 + PIN	UPDATE RECORD APPEND RECORD	Schlüssel_2 + Pin 1 Schlüssel_2 + Pin 1
	W	EF.ARR ₂ (Schlüssel_2 + PIN)	Erase Record	Schlüssel_2 + PIN		
					ENABLE VERIFICATION REQUIREMENT	Always
					DISABLE VERIFICATION REQUIREMENT	Always
FID		1F04		1F04		
SFID		--		04		
Anz. Records	50	5		20		
Record Grösse		135 Bytes		136 Bytes		
Grösse		675 Byte		2820 Byte		
Kommentar	1.			.		



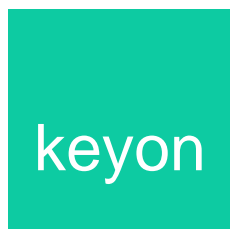
10.2.10 EF.ZUSE

[eCH64]::Seite 25		[VVK-EDI]:: Kap. 2.7				
[eCH64]		[POST]		[SASIS]		
Struktur	Linear variable	Linear variable		Linear variable		
Type	Record (Simple-TLV)	Record		Record		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R	EF.ARR ₂ (Schlüssel_1 + PIN)	Read	Schlüssel_1 + PIN	SELECT READ RECORD	Always Schlüssel_1 + Pin 1
	UP	EF.ARR ₂ (Schlüssel_2 + PIN)	Update	Schlüssel_2 + PIN	UPDATE RECORD APPEND RECORD	Schlüssel_2 + Pin 1 Schlüssel_2 + Pin 1
	W	EF.ARR ₂ (Schlüssel_2 + PIN)	Erase Record	Schlüssel_2 + PIN		
					ENABLE VERIFICATION REQUIREMENT	Always
					DISABLE VERIFICATION REQUIREMENT	Always
FID		1F04		1F04		
SFID		--		04		
Anz. Records Tag 1200	10	2		5		
Anz. Records Tag 1300	25			17		
Record Grösse Tag 1200		492 Bytes		500 Bytes		
Record Grösse Tag 1300				154 Bytes		
Grösse		984 Byte		2622 Byte		
Kommentar	1.			.		



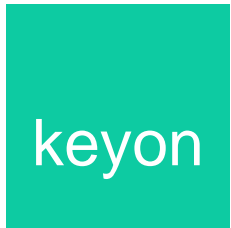
10.2.11 EF.MEDI

[eCH64]::Seite 26		[VVK-EDI]:: Kap. 2.4				
[eCH64]		[POST]		[SASIS]		
Struktur	Linear variable	Linear variable		Linear variable		
Type	Record (Simple-TLV)	Record		Record		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R	EF.ARR ₂ (Schlüssel_1 + PIN)	Read	Schlüssel_1 + PIN	SELECT READ RECORD	Always Schlüssel_1 + Pin 1
	UP	EF.ARR ₂ (Schlüssel_2 + PIN) oder (Schlüssel_3 + PIN)	Update	(Schlüssel_2 + PIN) oder (Schlüssel_3 + PIN)	UPDATE RECORD APPEND RECORD	(Schlüssel_2 + Pin 1) oder (Schlüssel_3 + PIN) (Schlüssel_2 + Pin 1) oder (Schlüssel_3 + PIN)
	W	EF.ARR ₂ (Schlüssel_2 + PIN) oder (Schlüssel_3 + PIN)	Erase Record	(Schlüssel_2 + PIN) oder (Schlüssel_3 + PIN)		
					ENABLE VERIFICATION REQUIREMENT	Always
					DISABLE VERIFICATION REQUIREMENT	Always
FID		1F06		1F06		
SFID		--		06		
Anz. Records	50	7		18		
Record Grösse		492 Bytes		270 Bytes		
Grösse		3444 Byte		4986 Byte		
Kommentar	1.	Warum ist der Record viel größer als benötigt?		.		



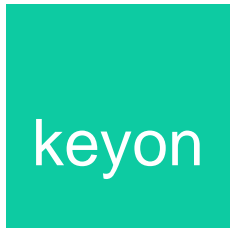
10.2.12 EF.ALLG

[eCH64]::Seite 26		[VVK-EDI]:: Kap. 2.3				
[eCH64]		[POST]		[SASIS]		
Struktur	Linear variable	Linear variable		Linear variable		
Type	Record (Simple-TLV)	Record		Record		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R	EF.ARR ₂ (Schlüssel_1 + PIN)	Read	Schlüssel_1 + PIN	SELECT READ RECORD	Always Schlüssel_1 + Pin 1
	UP	EF.ARR ₂ (Schlüssel_2 + PIN)	Update	Schlüssel_2 + PIN	UPDATE RECORD APPEND RECORD	Schlüssel_2 + Pin 1 Schlüssel_2 + Pin 1
	W	EF.ARR ₂ (Schlüssel_2 + PIN)	Erase Record	Schlüssel_2 + PIN		
					ENABLE VERIFICATION REQUIREMENT	Always
					DISABLE VERIFICATION REQUIREMENT	Always
FID		1F07		1F07		
SFID		--		07		
Anz. Records Tag 500	25	6		12		
Anz. Records Tag 600	25					
Record Grösse Tag 500		361 Bytes		345 Bytes		
Record Grösse Tag 600				350 Bytes		
Grösse		2166 Byte		4380 Byte		
Kommentar	1.	In der Implementierungsanleitung ist unklar, wie viele Tags 500 bzw. Tags 600 unterstützt werden		In der Detailspezifikation ist unklar, wie viele Tags 500 bzw. Tags 600 unterstützt werden		



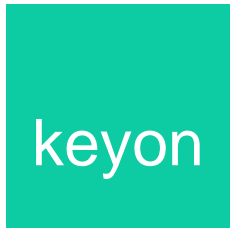
10.2.13 EF.ADDR

[eCH64]::Seite 26		[VVK-EDI]:: Kap. 2.8				
[eCH64]		[POST]		[SASIS]		
Struktur	Linear variable	Linear variable		Linear variable		
Type	Record (Simple-TLV)	Record		Record		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R	EF.ARR ₂ (Schlüssel_1)	Read	Schlüssel_1	SELECT READ RECORD	Always Schlüssel_1
	UP	EF.ARR ₂ (Schlüssel_1)	Update	Schlüssel_1	UPDATE RECORD APPEND RECORD	Schlüssel_1 Schlüssel_1
	W	EF.ARR ₂ (Schlüssel_1)	Erase Record	Schlüssel_1		
FID		1F08		1F08		
SFID		--		08		
Anz. Records	10	2		2		
Record Grösse		368 Bytes		374 Bytes		
Grösse		736 Byte		844 Byte		
Kommentar	1.					



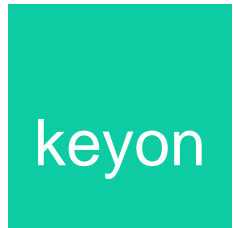
10.2.14 EF.VERF

[eCH64]::Seite 27		[VVK-EDI]:: Kap. 2.9				
[eCH64]		[POST]		[SASIS]		
Struktur	Linear variable	Linear variable		Linear variable		
Type	Record (Simple-TLV)	Record		Record		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R	EF.ARR ₂ (Schlüssel_1)	Read	Schlüssel_1	SELECT READ RECORD	Always Schlüssel_1
	UP	EF.ARR ₂ (Schlüssel_1)	Update	Schlüssel_1	UPDATE RECORD APPEND RECORD	Schlüssel_1 Schlüssel_1
	W	EF.ARR ₂ (Schlüssel_1)	Erase Record	Schlüssel_1		
FID		1F09		1F09		
SFID		--		09		
Anz. Records	10	2		2		
Record Grösse		470 Bytes		476 Bytes		
Grösse		940 Byte		1048 Byte		
Kommentar	1.					



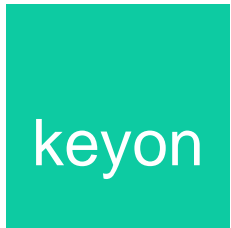
10.2.15 EF.ATR

[eCH64]::Seite 20		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Struktur	transparent	transparent		transparent		
Type	BER-TLV	TLV		TLV		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R		Read Binary	Always	SELECT	Always
			Update Binary	Never	Read Binary	Always
			Erase Binary	Never		
			Delete EF	Never		
FID		2F01		2F01		
SFID		--		01		
Grösse	maximal 33 Byte	34 Byte		43 Byte		
Kommentar	1.	Codierte APDU Kommando/Antwort-Länge =04AF -> 1199 Byte		Codierte APDU Kommando-/Antwort-Länge =0400 -> 1024 Byte		



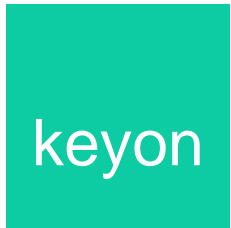
10.2.16 EF.PIN

[eCH64]::Seite 20		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Dateiname	EF.PIN	PIN.Not		iEF.PIN1		
Struktur	Transparent oder linear variable			Linear fixed		
Type	Bit-String oder Record					
Kategorie	Internal			Internal		
AM	AM	SC	AM	SC	AM	SC
	R		VERIFY	Always	SELECT	Always
	UP		CHANGE REFERENCE DATA	Always (mit alter PIN.Not)	VERIFY	Always
			RESET RETRY COUNTER	Always (mit PUK.Not)	CHANGE REFERENCE DATA	mit PIN1
			ENABLE VERIFICATION REQUIREMENT	Always (mit PIN.Not)	RESET RETRY COUNTER	mit PUK
			DISABLE VERIFICATION REQUIREMENT	Always (mit PIN.Not)	ENABLE VERIFICATION REQUIREMENT	Always
					DISABLE VERIFICATION REQUIREMENT	Always
FID		01		0011		
SFID		--		11		
Grösse				10 Byte		
Kommentar	1.					



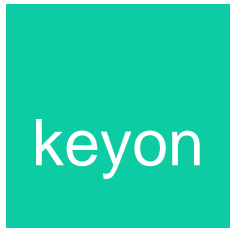
10.2.17 EF.StatusPIN

[eCH64]::Seite 21		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Struktur	Transparent oder linear variable					
Type	Bit-String oder Record					
Kategorie	Internal					
AM	AM	SC	AM	SC	AM	SC
	R					
	UP					
FID						
SFID						
Grösse						
Kommentar	1.	Nicht vorhanden		Nicht vorhanden		



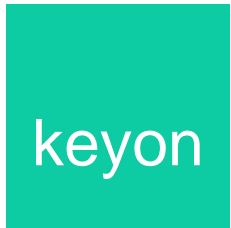
10.2.18 EF.S_B

[eCH64]::Seite 21		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Dateiname	EF.S _B			iEF.PrK.SB		
Struktur	Transparent oder linear variable			Linear variable		
Type	Bit-String oder Record					
Kategorie	Internal			Internal		
AM	AM	SC	AM	SC	AM	SC
	R				Select	Always
					INTERNAL AUTHENTICATE	Always
FID				0015		
SFID				15		
Grösse				1224 Bytes		
Kommentar	1.			Der Schlüssel S _B wird nicht explizit beschrieben		



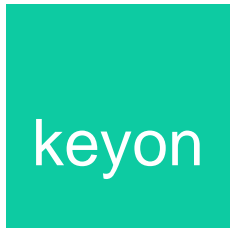
10.2.19 EF.ARR₁

[eCH64]::Seite 19		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Struktur	linear variable					
Type	Record					
Kategorie	Internal					
AM	AM	SC	AM	SC	AM	SC
	R					
FID						
SFID						
Grösse						
Kommentar	1.	Nicht vorhanden		Nicht vorhanden		



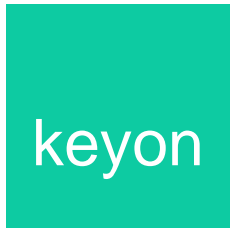
10.2.20 EF.ARR₂

[eCH64]::Seite 23		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Struktur	linear variable					
Type	Record					
Kategorie	Internal					
AM	AM	SC	AM	SC	AM	SC
	R					
FID						
SFID						
Grösse						
Kommentar	1.	Nicht beschrieben		Nicht beschrieben		



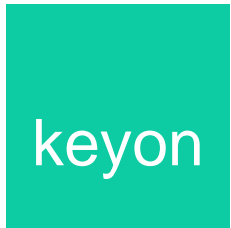
10.2.21 EF.CVC.PDC

[eCH64]::Seite 22		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Struktur	Transparent	Transparent		Transparent		
Type	Binary	Binary		Binary		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R		READ BINARY	Always	READ BINARY	Always
			UPDATE BINARY	Never	SELECT	Always
			ERASE BINARY	Never		
			DELETE EF	Never		
FID		2F03		2F03		
SFID		03		03		
Grösse		217		618		
Kommentar	1.					



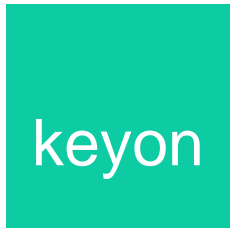
10.2.22 EF.PK.CA_ORG_PDC

[eCH64]::Seite 22		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Filename	EF.PK.CA_ORG_PDC	EF.PuK.CA_ROOT_VK		EF.PuK.CA_ROOT_VK		
Struktur	Transparent oder Linear variable	Transparent		Linear fixed		
Type	Record oder Bit-String	Binary		Record		
Kategorie	Internal			Working		
AM	AM	SC	AM	SC	AM	SC
	R		READ BINARY	Always	SELECT	Always
			UPDATE BINARY	Never	MSE	Always
			ERASE BINARY	Never	READ RECORD	Always
			DELETE EF	Never		
FID		0E02		001C		
SFID				1C		
Grösse		140 Bytes		299 Bytes		
Kommentar	1.					



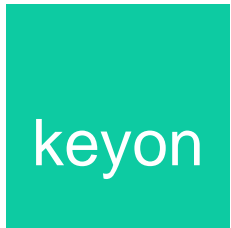
10.2.23 EF.CVC.CA_ORG_PDC

[eCH64]::Seite 22		[VVK-EDI]::				
[eCH64]		[POST]		[SASIS]		
Filename	EF.CVC.CA_ORG_PDC	EF.CVC.CA_ORG_PDC		EF.CVC.CA_ORG_PDC		
Struktur	Transparent	Transparent		Transparent		
Type	Binary	Binary		Binary		
Kategorie	Working			Working		
AM	AM	SC	AM	SC	AM	SC
	R		READ BINARY	Always	SELECT	Always
			UPDATE BINARY	Never	READ BINARY	Always
			ERASE BINARY	Never		
			DELETE EF	Never		
FID		2F04		2F08		
SFID		04		08		
Grösse		217 Bytes		624 Bytes		
Kommentar	1.					



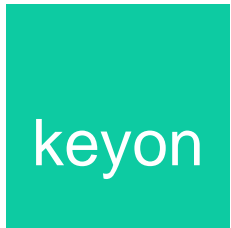
10.2.24 EF.CVC.CA_ORG_HPC

[eCH64]::Seite 22		[VVK-EDI]::			
[eCH64]		[POST]		[SASIS]	
Filename	EF.CVC.CA_ORG_HPC				
Struktur	Transparent				
Type	Binary				
Kategorie	Working				
AM	AM	SC	AM	SC	
	R				
FID					
SFID					
Grösse					
Kommentar	1.		Nicht beschrieben		Nicht beschrieben



10.2.25 EF.AUT

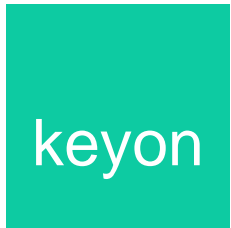
[eCH64]::Seite 24		[VVK-EDI]::			
[eCH64]		[POST]		[SASIS]	
Struktur	Linear fixed				
Type	Record				
Kategorie	Internal				
AM	AM	SC	AM	SC	
	R				
FID					
SFID					
Grösse					
Kommentar	1.	Nicht beschrieben		Nicht beschrieben	



10.3 Dateien ausserhalb des Standards

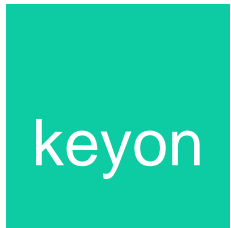
10.3.1 EF.Version

[eCH64]::		[VVK-EDI]::			
	[eCH64]	[POST]::Seite 10		[SASIS]::Seite 14	
Dateiname		EF.VERSION		EF.VERSION	
Struktur		Transparent		Transparent	
Type		Binary		Binary	
Kategorie				Working	
AM	AM	SC	AM	SC	
			READ BINARY	Always	SELECT
			UPDATE BINARY	Never	READ BINARY
			ERASE BINARY	Never	
			DELETE EF	Never	
FID		2F10		5600	
SFID		10		B0	
Grösse		4 Bytes		4 Bytes	
Kommentar	1.	Nicht spezifiziert			



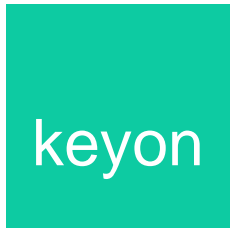
10.3.2 IEF.C2CSTATE

[eCH64]::		[VVK-EDI]::			
[eCH64]		[POST]::Seite 10		[SASIS]::Seite 16	
Dateiname				IEF.C2CSTATE	
Struktur				Transparent	
Type					
Kategorie				Internal	
AM	AM	SC	AM	SC	
					SELECT
					PSO
FID				001D	
SFID				1D	
Grösse				6 Bytes	
Kommentar	1.	Nicht spezifiziert		Nicht spezifiziert	



10.3.3 EF.GPKeys

[eCH64]::		[VVK-EDI]::				[SASIS]::Seite 16	
		[eCH64]		[POST]::Seite 10		[SASIS]::Seite 16	
Dateiname						EF.GPKeys	
Struktur						Transparent	
Type							
Kategorie						Internal	
AM	AM	SC	AM	SC		No Access	Always
							Always
FID					0001		
SFID					01		
Grösse					6 Bytes		
Kommentar	1.	Nicht spezifiziert		Nicht spezifiziert			



10.3.4 Personal Unblocking Key

[eCH64]::			[VVK-EDI]::			
	[eCH64]		[POST]::Seite 10		[SASIS]::Seite 16	
Dateiname			PUK.Not		iEF.PUK	
Struktur					Linear fixed (Chipkatenbetriebssystemabhängig)	
Type						
Kategorie					Internal	
AM	AM	SC	AM	SC		
					SELECT	Always
					VERIFY	Always
FID					0014	
SFID					14	
Grösse					10 Bytes	
Kommentar	1.	Nicht spezifiziert	Nicht genau beschrieben			

10.3.5 SK.Admin

Das File SK.Admin mit der Key ID '10' ist nur bei der Post implementiert und ist für den Verwendungszweck 'desSessionkey4SM' vorgesehen.