Bundesverband der Deutschen Volksbanken und Raiffeisenbanken e. V.
Bundesverband deutscher Banken e. V.
Bundesverband Öffentlicher Banken Deutschlands e. V.
Deutscher Sparkassen- und Giroverband e. V.
Verband deutscher Pfandbriefbanken e. V.

Die Deutsche Kreditwirtschaft

FinTS

Financial Transaction Services

Schnittstellenspezifikation

Security

Alternative Sicherheitsverfahren

Herausgeber:

Bundesverband deutscher Banken e.V., Berlin

Deutscher Sparkassen- und Giroverband e.V., Bonn/Berlin

Bundesverband der Deutschen Volksbanken und Raiffeisenbanken e.V., Berlin

Bundesverband Öffentlicher Banken Deutschlands e.V., Berlin

Version: 3.0 Stand:22.01.2013 Final Version

Die vorliegende Schnittstellenspezifikation für eine automatisiert nutzbare multibankfähige Homebanking-Schnittstelle (im Folgenden: Schnittstellenspezifikation) wurde im Auftrag der Deutschen Kreditwirtschaft entwickelt. Sie wird hiermit zur Implementation in Kunden- und Kreditinstitutssysteme freigegeben.

Dieses Dokument kann im Internet abgerufen werden unter http://www.fints.org.

Versionsführung

Das vorliegende Dokument wurde von folgenden Personen erstellt bzw. geändert:

Name	Organisation	Datum	Version	Anmerkungen
Haubner	für SIZ	15.12.2009	1.0	Erstellung Final Version
Haubner	für SIZ	22.01.2013	1.0	Entfernen nicht benötigter Sicherungsverfahren

Änderungen gegenüber der Vorversion

Hinzufügungen und Änderungen sind im Dokument in dieser Farbe und zusätzlich durch Unterstreichung und einen Randbalken markiert. Löschungen sind aufgrund der besseren Übersichtlichkeit nur durch einen Randbalken markiert. Hypertextlinks sind in dieser Farbe markiert. Falls sich die Kapitelnummerierung geändert hat, bezieht sich die Kapitelangabe auf die neue Nummerierung. Aufgrund der umfangreichen Textumstellungen wurden nicht alle Änderungen markiert.

lfd. Nr.	Kapitel	Seiten-nummer	Ken-nung[1]	Art[2]	Beschreibung
1					
2					
3					
4					

[1] nur zur internen Zuordnung

[2] F = Fehler; Ä = Änderung; K = Klarstellung; E = Erweiterung

Inhaltsverzeichnis

Versionsführung 1

Änderungen gegenüber der Vorversion 2

Inhaltsverzeichnis 3

Abbildungsverzeichnis 5

Begriffe 6

Abkürzungen 8

Literaturhinweise 10

A. Einleitung 11

A.1 Secodervisualisierung 12

A.2 Secoder-Integration in FinTS 12

A.2.1 Secoder-Applikationen und Verwendungsmöglichkeiten 13

A.2.2 Secoder-Verwaltung / Abfrage der Secoder-Eigenschaften 13

A.2.2.1 Einführen einer Metasprache 13

B. Verfahrensbeschreibung 15

B.1 Allgemeines 15

B.2 AZS-Verfahren zur Secoder-Integration 16

B.2.1 AZS-Verfahren 16

B.2.2 Abläufe für AZS-Verfahren 19

B.2.2.1 Sicherheitsverfahren mit Secoder 20

B.3 Geschäftsvorfall HKAZS für AZS-Verfahren 26

B.3.1 Aufbau des Geschäftsvorfalls HKAZS 26

B.4 Erweiterung der Rückmeldungscodes 30

B.4.1 Beschreibung spezieller Rückmeldungen im AZS-Verfahren 31

B.5 Erweiterung der Bank- und Userparameterdaten (BPD / UPD) 32

B.5.1 Secoder-spezifische Visualisierungsinformationen (HIVISS) 33

B.5.1.1 Generelles Secoder-Visualisierungskonzept mit HIVISS 33

B.5.1.2 Struktur des Parametersegmentes HIVISS 41

B.5.1.3 Tabelle der Secodervisualisierungstexte 42

B.5.1.4 Geschäftsvorfallspezifische Visualisierungsinformationen für Secoder 45

B.5.1.5 Positionierung bei der Secodervisualisierung 47

B.5.1.6 Parametersegment HIVISS 52

B.5.2 Spezielle Festlegungen für die Dialoginitialisierung beim AZS-Verfahren 53

C. Dialogspezifikation für AZS-Verfahren 55

C.1 Allgemeines 55

C.1.1 Verschlüsselung des Dialoges 55

C.1.2 Institutssignaturen bei AZS-Verfahren 56

C.1.3 Key-Management bei AZS-Verfahren 56

C.1.4 Behandlung der Dialogendenachricht (HKEND) 56

C.2 Besondere Belegungsrichtlinien für AZS-Verfahren 56

C.2.1 Segment Sicherheitsverfahren, DEG Unterstützte Sicherheitsverfahren 57

C.2.2 DEG „Sicherheitsprofil“ 57

C.2.3 DEG „Schlüsselname“ 57

C.2.4 DEG „Sicherheitsidentifikation, Details“ 57

C.2.5 Segment „Signaturkopf“ 58

C.2.6 DEG „Hashalgorithmus“ 58

C.2.7 DEG „Signaturalgorithmus“ 58

C.2.8 Segment „Signaturabschluss“ 58

C.2.9 Segment „Verschlüsselungskopf“ 58

C.2.10 DEG „Verschlüsselungsalgorithmus“ 58

C.2.11 Segment „Verschlüsselte Daten“ 59

C.3 Nachrichtenaufbau für AZS-Verfahren 59

C.3.1 Kundennachricht bei der Dialoginitialisierung 59

C.3.1.1 Nachrichtenformat 60

C.3.2 Kreditinstitutsnachricht bei der Dialoginitialisierung 60

C.3.2.1 Nachrichtenformat 62

C.3.3 Kundenauftragsnachricht 63

C.3.4 Kreditinstitutsauftragsnachricht 64

C.3.5 Kundennachricht bei Mehrfachsignaturen 65

D. Secoder-Management 67

D.1 Übermitteln / Anzeigen von Secoder-Informationen 67

E. Data-Dictionary 70

F. Anlagen 84

F.1 Übersicht der Segmente 84

F.2 Übersicht Nachrichtenaufbau 84

Abbildungsverzeichnis

Abbildung 1: Mögliche logische Architekturen zur Integration des Secoders über eine Metadatenschnittstelle 14

Abbildung 3: Dialoginitialisierung beim AZS-Verfahren S-Fkt=811 (2 von 2) 21

Abbildung 4: Fortgeschrittene Signatur mit Secoder, S-Fkt=811 24

Abbildung 6: Zusammenhang zwischen Secoder MetaData und Secodervisualisierungstexten 36

Abbildung 7: Aufbau des Parametersegmentes HIVISS 41

Abbildung 8: Tabelle der Secodervisualisierungstexte und Secoder MetaData in HIVISS 42

Abbildung 9: Analogien zwischen MetaData und Secoder Data Confirmation 45

Abbildung 10: Definition der Secoder MetaData pro Geschäftsvorfall 46

Abbildung 11: Adressierung der Secodervisualisierungsdaten bei FinTS-Formaten 48

Abbildung 12: Adressierung der Secodervisualisierungsdaten bei DTA-Formaten 49

Abbildung 13: Adressierung der Secodervisualisierungsdaten bei DTAZV-Formaten 50

Abbildung 14: Adressierung der Secodervisualisierungsdaten bei SEPA-Formaten 51

Abbildung 15: Segmentaufbau der Dialoginitialisierungsnachricht (Kunde) bei AZS-Verfahren 59

Abbildung 16: Segmentaufbau der Dialoginitialisierungsnachricht (Kreditinstitut) bei AZS-Verfahren 61

Abbildung 18: Segmentaufbau der Auftragsnachricht (Kreditinstitut) bei AZS-Verfahren 64

Abbildung 19: Kundennachricht bei Mehrfachsignaturen 65

Abbildung 20: Reihenfolge der Sicherheitssegmente bei Mehrfachsignaturen 66

Begriffe

Abkürzung	Bedeutung
Anzeige-definition	=HIVISS Anzeigedefinition: Eine Beschreibung von Index, Secoder-Text und Secoder MetaData für eine Displayzeile im Rahmen eines Secodervisualisierungstextes.
AZS-Verfahren	Unter AZS-Verfahren (Alternative ZKA-Sicherheitsverfahren) werden alle Sicherheitsverfahren verstanden, deren Visualisierungsinformationen und Signaturen mit Hilfe des Geschäftsvorfalls HKAZS transportiert werden wie z. B. Secoder-Signaturen.
FinTS-Instanz	Komponente z. B. einer Finanzmanagementsoftware, die für die Abwicklung des FinTS-Protokolls zuständig ist.
FinTS-Füllwert	Als FinTS-Füllwert wird eine Belegung des entsprechenden Datenelementes betrachtet, welche den getroffenen Festlegungen (Formatvorgaben, Restriktionen, Belegungshinweisen) nicht widerspricht. Ein FinTS-Füllwert ist somit ein gültiger Wert im Sinne der Definition des Datenelementes. Trotzdem ist dieser FinTS-Füllwert des betroffenen Datenelements für die Verarbeitung nicht relevant und wird daher von den verarbeitenden Systemen auf Kreditinstitutsseite ignoriert. Handelt es sich um Datenelemente mit Status „O“, sollten diese leer gelassen werden. Auch hier gilt, dass Vorhandensein und Inhalt kreditinstitutsseitig nicht geprüft werden.
MetaData	(Secoder-) MetaData oder Metadaten werden als Eingangsschnittstelle zur Secoder-Anwendungsfunktion im Kundensystem benutzt. Sie entsprechen fachlich den Parameterstrukturen DSx, wie sie in den Data Confirmations des Secoders definiert sind. In FinTS werden die Metadaten im Rahmen des DE Secodervisualisierungstexte im BPD-Segment HIVISS abgebildet. Die Metadaten werden von der Secoder-Anwendungsfunktion verwendet und in ein logisches Secoder-Protokoll eingebettet, das physisch wiederum z. B. über PC/SC abgewickelt wird. Durch den Einsatz dieser Metadaten muss die Online-Banking-Applikation selbst kein Wissen bzgl. Protokoll und konkreter Datenschnittstelle zum Secoder besitzen.

Online-Banking-Applikation	Bei der Online-Banking-Applikation kann es sich um eine FinTS Finanzmanagementsoftware oder eine browserbasierte Komponente (Java-Applet / -Servlet, PlugIn, …) handeln. Die Online-Banking-Applikation kommuniziert mit der Secoder-Anwendungsfunktion auf Basis von Metadaten.
Secoder	Unter Secoder wird eine neue Generation von DK Chipkartenlesern verstanden, bei denen die Möglichkeit besteht, in einem so genannten Applikationsmodus Transaktionsdaten auf sichere Weise im Display anzuzeigen und mit Hilfe einer Bankensignaturkarte zu signieren. Zur Signaturbildung wird die Signatur-Anwendung auf der Karte verwendet. Im Rahmen der vorliegenden Spezifikation wird ein Secoder als Basis vorausgesetzt und durchgängig als „Secoder“ bezeichnet.
Secoder-Anwendungsfunktion	Anwendungsfunktion, welche auf Basis übergebener Metadaten einen z. B. über PC/SC angeschlossenen Secoder protokoll- und datenmäßig bedienen kann. Die Secoder-Anwendungsfunktion kann eine Komponente einer Finanzmanagementsoftware bzw. im Browserkontext ein Java Applet oder PlugIn sein. Teile der Secoder-Anwendungsfunktion können sich auch auf einem Web-oder Application-Server befinden.
Secoder-Kryptogramm	Alternativer Begriff für Secoder-Signaturen. Bei Secoder-Signaturen fließen in die Secoder-Kryptogrammbildung die durch den Kunden bestätigten Daten (VisData) ein.
Secoder-spezifische Kommandos (SecCmds)	Diese Secoder-spezifischen Kommandos werden an der Schnittstelle zwischen Secoder-Anwendungsfunktion und dem Secoder selbst z. B. über PC/SC ausgetauscht. Beispiele hierfür sind „Select Application“ oder „Data Confirmation“.
VisAuthSig	Visualisierungsauthentikationssignatur (auch VisAuth-Signatur) des Secoders. Diese Signatur dient bei Verwendung der Secoder-Applikationen „aut“ und „sig“ zum Nachweis gegenüber dem Kreditinstitut, dass – falls ein Secoder am Kundenendgerät angeschlossen war – dieser sich zum Zeitpunkt der VisData-Signatur im Applikationsmodus befunden hat.
VisData	Analog der Secoder-Spezifikation wird hierunter der Aufbau der zu signierenden Daten im Secoder, d. h. zwischen Lesereinheit und Chipkarte verstanden.
VisDataSig	(auch VisData-Signatur)Signatur über den VisData-Bereich des Secoders.

Abkürzungen

Abkürzung	Bedeutung
AZS	Alternative ZKA-Sicherheitsverfahren
ATC	Application Transaction Counter
AUT-SIG	Authentikations-Signatur
BPD	Bankparameterdaten
C	Datenstruktur ist konditional
CR	Carriage-Return (Wagenrücklauf)
DDV	DES-DES-Verfahren
DE	Datenelement
DEG	Datenelementgruppe
DES	Data Encryption Standard
DK	Die Deutsche Kreditwirtschaft
DS	Digitale Signatur (z. B. Schlüsselart)
EF	Elementary File
EU	Elektronische Unterschrift; basiert auf dem asymmetrischen RSA-Verfahren
GD	Gruppendatenelement
GDG	Gruppendatenelementgruppe
HBCI	Homebanking Computer Interface
I	Information (z.B. Schlüsselart)
ID	Identifikationsmerkmal (Nummer oder alphanumerischer Code)
ISO	International Organisation for Standardisation
LF	Line-Feed (neue Zeile)
M	Datenstruktur muss vorhanden sein und ist inhaltlich korrekt zu füllen
MAC	Message Authentication Code; Symmetrisches Verfahren zur Erzeugung einer elektronischen Signatur (derzeit für die DK-Chipkarte eingesetzt)
N	Nachricht
N	Nicht erlaubt (not allowed) (Datenstruktur darf nicht vorhanden sein)
O	Datenstruktur ist optional
OBanking-PIN	Online-Banking-PIN
PIN	Private Identifikationsnummer
RAH	RSA-AES-Hybridverfahren
RDH	RSA-DES-Hybridverfahren
RFC	Request for Comment
RSA	Asymmetrischer Algorithmus für die elektronische Unterschrift (EU) (vgl. MAC), benannt nach den Erfindern Rivest, Shamir und Adleman.
SecCmds	Secoder-spezifische Kommandos

Abkürzung	Bedeutung
SEG	Segment
SEQ	Sequenznummer
SF	Segmentfolge
S-Fkt(. bzw. =)	Sicherheitsfunktion, kodiert (DE aus dem HBCI Signaturkopf)
SSL	Secure Socket Layer
TAN	Transaktionsnummer
UPD	Userparameterdaten
VisAuthSig	Visualisierungsbestätigungssignatur des Secoders
VisDataSig	Signaturdaten – Auftragssignatur des Secoders

Literaturhinweise

[Formals] Financial Transaction Services (FinTS) – Formals (Allgemeine Festlegungen für multibankfähige Online-Verfahren der deutschen Kreditwirtschaft), Version 3.0, 14.06.2011, Zentraler Kreditausschuss

[RMCode] Financial Transaction Services (FinTS) – Rückmeldungscodes (Ergänzung zum Band „FinTS Formals“), Version 3.0, 22.01.2013, Die Deutsche Kreditwirtschaft

[HBCI] Financial Transaction Services (FinTS) – Security (Sicherheitsverfahren HBCI), Version 3.0, 25.09.2012, Zentraler Kreditausschuss

[HHD] Schnittstellenspezifikation für die ZKA Chipkarte – HandHeld-Device (HHD) zur TAN-Erzeugung, Version 1.4, 07.05.2010, Zentraler Kreditausschuss

[HHD-Erweiterung] HHD-Erweiterung für unidirektionale Kopplung, Version 1.4, Final Version, 07.05.2010, Zentraler Kreditausschuss

[HHD-Belegung] ZKA TAN-Generator – Belegungsrichtlinien zur Dynamisierung der TAN, Version 1.4 inkl. Errata 1 bis 4, Final Version, 31.01.2012, Zentraler Kreditausschuss

[Messages] Financial Transaction Services (FinTS) – Messages (Multibankfähige Geschäftsvorfälle), Version 3.0, 06.08.2010, Zentraler Kreditausschuss

[PINTAN] Financial Transaction Services (FinTS) – Security (Sicherheitsverfahren PIN/TAN), Version 3.0, 27.10.2010, Zentraler Kreditausschuss

[Secoder] Secoder – Connected Mode Reader Applications, Version 2.2 Final Version, 05.08.2011, Zentraler Kreditausschuss

[Secoder Impl] Secoder 2 – User Interface und Implementation Guide, Draft 0.5, 14.09.2012, Zentraler Kreditausschuss

A. EINLEITUNG

In dieser Spezifikation wird ein multibankfähiges FinTS-Protokoll für die Integration „alternativer ZKA Sicherheitsverfahren“ (im Folgenden als „AZS-Verfahren“ bezeichnet) beschrieben. Darunter werden Verfahren verstanden, deren Sicherungsinformationen wie z. B. Signaturen nicht ausschließlich im FinTS-Signaturkopf bzw. –Abschluss übertragen werden, sondern über einen speziellen, neuen Geschäftsvorfall „HKAZS“ ergänzt werden. Grund dafür ist, dass bei diesen Verfahren zusätzliche Informationen wie z. B. Visualisierungsdaten übertragen werden müssen. Im Speziellen handelt es sich hierbei um die Einbindung des Secoders (Begriffsdefinition siehe Kapitel “Begriffe“) in den verschiedenen Ausprägungen und Einsatzszenarien, von denen momentan nur die Anwendung „Signaturmodus“ unterstützt ist.

AZS-Verfahren verhalten sich vom Protokoll her wie normale FinTS-Transaktionen im Ein-Schritt-Verfahren. Informationen bzgl. Nachrichtenaufbau und Dialogablauf sind dem Dokument [Formals] zu entnehmen.

Um ein möglichst hohes Maß an Synergie nutzen zu können, wird für die Kommunikation zwischen Kundenprogramm und Kreditinstitut weitestgehend auf der FinTS-Spezifikation V3.0 (Sicherheitsverfahren HBCI bzw. PIN/TAN) [HBCI], [PIN/TAN] aufgesetzt, insbesondere bzgl. Syntax, Datenformaten und Abläufen. Sofern nicht anders vermerkt gelten für den Nachrichtenaufbau, Dialogablauf etc. die dort getroffenen Regelungen. Das vorliegende Dokument beschreibt daher nur die für AZS-Verfahren abweichenden Festlegungen.

Bei AZS-Verfahren werden die bestehenden FinTS-Sicherheitsverfahren ersetzt; daher enthalten die FinTS-Signaturen nur geeignete FinTS-Füllwerte.

Ob ein Kreditinstitut AZS-Verfahren anbietet, erkennt das Kundenprodukt in den Bankparameterdaten am Vorhandensein des Geschäftsvorfallparametersegments HIAZSS („Parameter Alternative ZKA Sicherheitsverfahren“, vgl. Kapitel B.5).

Grundsätzlich können mit AZS-Verfahren alle im Dokument [Messages] aufgeführten Geschäftsvorfälle verwendet werden. Dies gilt auch für verbandsindividuelle Erweiterungen. Welche Geschäftsvorfälle konkret mit welchem AZS-Verfahren zulässig sind, teilt das Kreditinstitut im Segment HIAZSS (s. Kap. B.5) mit.

Als Verschlüsselungsverfahren wird beim spezifizierten Verfahren mit der Sicherheitsfunktion 811 die HBCI-Verschlüsselung verwendet.

Bei AZS-Verfahren handelt es sich wie bei HBCI um Ein-Schritt-Verfahren, d. h. der Geschäftsvorfall wird in einem Prozess-Schritt zusammen mit allen benötigten Signaturinformationen eingereicht und somit in einem Dialogschritt bestehend aus Auftrag und Antwort wird ein Geschäftsvorfall komplett abgewickelt.

Diese Verfahrensweise entspricht dem Vorgehen beim Sicherheitsverfahren HBCI und war bis zur Einführung des Zwei-Schritt-Verfahrens für PIN/TAN die einzige Möglichkeit, Aufträge über das FinTS-Protokoll einzureichen.

Durch die neu eingeführten Secoder Metadaten (vgl. Abschnitt „Begriffe“), die über die Parameter-Segmente HIAZSS und HIVISS abgebildet werden, können die Secoder-Verfahren für fortgeschrittene Signaturen ebenfalls das Ein-Schritt-Verfahren einsetzen.

A.1 Secodervisualisierung

Bereits bei der Integration von chipTAN oder mobileTAN wurden Teile des Auftrags als Challenge visualisiert und durch den Kunden bestätigt. Bei AZS-Verfahren wird dieser Ansatz weiter verfeinert, da hier ggf. größere Datenmengen zu visualisieren sind. Beim Secoder können bei maximaler Puffergröße und einer 2 x 16 Anzeigeeinheit z. B. bis zu 9 Elemente im Display durchgeblättert und bestätigt werden (Secodervisualisierung).

Bei Einzelaufträgen ist die Secodervisualisierung auf einfache Art zu bewerkstelligen, wobei die Bankanwendung die Secodertexte frei definieren kann. So können beispielsweise auch in Anlehnung an die Belegungsrichtlinien von HHD V1.4 Daten aus dem Auftrag in geeigneter Weise angezeigt und bestätigt werden. Der Secoder erlaubt mit seinen Techniken hierfür noch weitreichendere Möglichkeiten als das HHD.

Bei Sammelaufträgen müssen Konstrukte geschaffen werden, deren Secodervisualisierung und Bestätigung dem Kunden einerseits noch zumutbar sind und ein vertretbares Sicherheitsniveau garantieren. Solche Lösungen können jedoch immer nur einen Kompromiss darstellen.

Als Möglichkeiten der Auswahl der Secodervisualisierungsdaten bei solch großen Datenmengen kommen im Wesentlichen Summenwerte über verschiedene Daten wie Anzahl der Sätze oder Beträge in Frage. Auch die Signatur von Hashwerten und Vergleich mit einem zur Verfügung gestellten „Hashwert-Tool“ ist Praxis. Generell ist die Thematik nicht neu und im Firmenkundengeschäft generell zu betrachten.

Die gesamte Themenstellung ist jedoch nicht Inhalt dieser Spezifikation, sondern ist Betreiber-spezifisch zu lösen. Die Möglichkeiten hierfür sind durch die Verwendung der Metadaten in der BPD gegeben.

A.2 Secoder-Integration in FinTS

Zur Unterstützung des Secoders werden neue Funktionalitäten in das FinTS-Protokoll integriert wie z. B. die Visualisierung eines Teils der Transaktionsdaten und das daraus resultierende Secoder-Kryptogramm. Die Secoder-Integration hat aber auch inhaltliche Auswirkungen, wie in den folgenden Abschnitten dargestellt ist.

A.2.1 Secoder-Applikationen und Verwendungsmöglichkeiten

In FinTS wird derzeit nur die folgende Secoder-Applikation unterstützt:

- „aut“ Signaturapplikation unter Verwendung des AUT-Schlüssels

Die Applikation „aut“ wird in FinTS durch das Verfahren gemäß Sicherheitsfunktion 811 beschrieben.

In den folgenden Prozessabläufen wird davon ausgegangen, dass bei der Signaturbildung sowohl die Auftragsdaten (in Form eines übergebenen Hashwerts über den FinTS-Geschäftsvorfall) kombiniert mit den Secoder-Visualisierungsdaten signiert werden als auch eine Visualisierungsauthentikationssignatur gebildet wird, durch das FinTS-Protokoll also zwei Signaturen zu übertragen sind. Dies wird durch zwei Secoder-Aufrufe der Applikationen aut/aut erreicht.

A.2.2 Secoder-Verwaltung / Abfrage der Secoder-Eigenschaften

Grundsätzlich muss ein Kreditinstitut die physischen Eigenschaften des vom Kunden genutzten Secoders nicht kennen. Durch die Einführung einer Metasprache (siehe nächster Abschnitt) erfolgt eine logische Entkopplung der fachlichen von den physischen Eigenschaften. Durch die Angabe im DE „Sicherheitsfunktion, kodiert“ gibt das Kreditinstitut zwar die unterstützten Varianten von AZS-Verfahren vor, es werden dort aber keine physischen Eigenschaften des Secoders vorgegeben. Die Secoder-Anwendungsfunktion ist dafür verantwortlich, aus den übergebenen Metadaten passende SecCmds für den Secoder-Aufruf zu erzeugen. Da mit der aktuellen Version der Secoder-Spezifikation 2.2 (Erratum vom 16.08.2012) auch größere Displays als 2 x 16 unterstützt werden, muss die Secoder-Anwendungsfunktion entweder die physischen Eigenschaften des angeschlossenen Secoders kennen (per SecCmd „SECODER INFO“) oder nur jeweils 2 x 16 Zeichen ausgeben.

Auch für ein Kreditinstitut kann es aus organisatorischen Gründen wichtig sein, die physischen Eigenschaften des verwendeten Secoders zu kennen. Um einen Überblick über die wichtigsten Secoder-Eigenschaften wie z. B. die Displaygröße oder die eindeutige Reader-Id des Secoders zu erfahren wurde ein neuer Geschäftsvorfall (vgl. Abschnitt D.1 „Übermitteln / Anzeigen von Secoder-Informationen“) eingeführt, mit dem diese Daten zwischen Secoder-Anwendungsfunktion / FinTS-Instanz und Kreditinstitut ausgetauscht werden können.

A.2.2.1 Einführen einer Metasprache

Da es aus vielerlei Gründen nicht erwünscht ist, dass Online-Banking-Applikationen direkt Secoder-spezifische Kommandos aufbauen und übertragen, bietet sich die Schaffung einer Metasprache (vgl. Definiton „MetaData“ in Abschnitt “Begriffe“) an, die auf hohem Abstraktionsgrad all die Möglichkeiten abbildet, die ein Secoder darstellen kann. Hierzu gehört beispielsweise auch die Information, ob ein Element nur bestätigt oder vom Kunden eingetippt werden soll. Die MetaData entsprechen daher inhaltlich in etwa den Datensätzen DSx im Input von DATA CONFIRMATION.

Die Metasprache kapselt auch die beiden Secoder-Anwendungsaufrufe aut/aut zur Signaturbildung.

Bei FinTS V3.0 wird die Metasprache durch das neue BPD-Segment HIVISS abgebildet.

Die von der Online-Banking-Applikation erzeugten MetaData müssen von einer geeigneten Secoder-Anwendungsfunktion in die Secoder-spezifischen Kommandos umgesetzt werden. Dies kann zum einen ein Bestandteil eines FinTS-Kundenproduktes sein, im Bereich Internet-Banking z. B. aber auch eine Webserver-Applikation mit einem PlugIn oder Applet auf dem Kunden-PC (für die Ansteuerung des Secoders wird in jedem Fall eine aktive Komponente auf dem Kunden-PC benötigt).

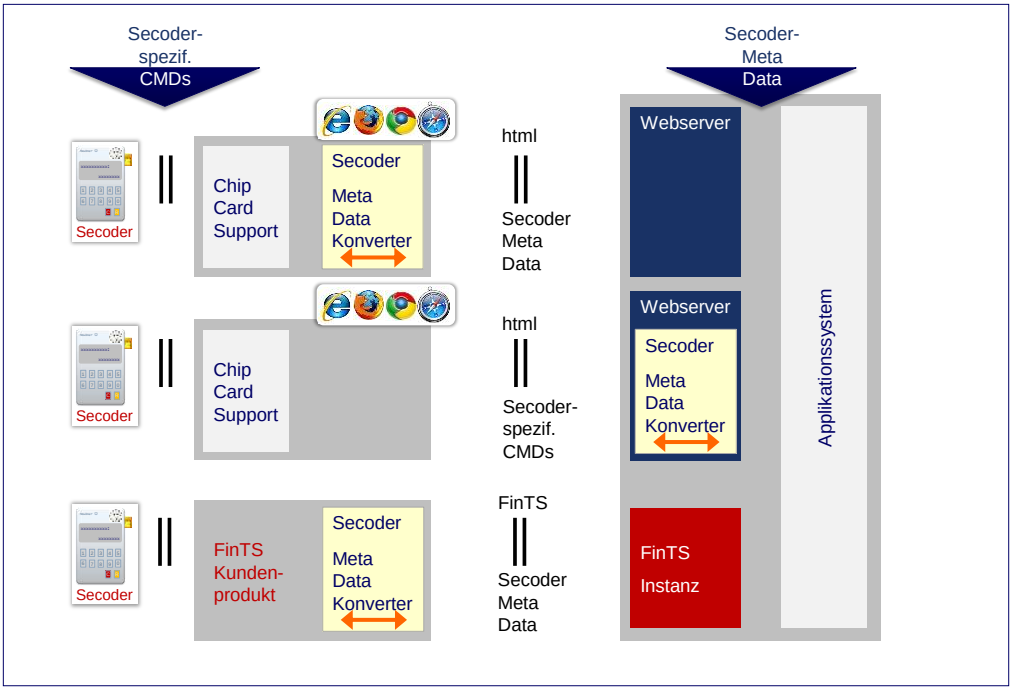


Abbildung 1: Mögliche logische Architekturen zur Integration des Secoders über eine Metadatenschnittstelle

B. VERFAHRENSBESCHREIBUNG

B.1 Allgemeines

Es gelten die in [Formals], [HBCI] und im übertragenen Sinn die in [Belegung] aufgeführten Formate und Belegungsrichtlinien.

Ergänzend bzw. abweichend hierzu gilt:

- Beim Einsatz von AZS-Verfahren kommt der neue Geschäftsvorfall HKAZS für die Abwicklung und die Parametersegmente HIAZSS und HIVISS für die Festlegungen hinzu. In den neuen BPD-Segmenten sind die spezifischen Informationen des Kreditinstituts für den Einsatz der unterstützten AZS-Verfahren bis hin zu einer elementweisen Definition der zu visualisierenden Daten enthalten.
- Der für den Kunden zugelassene Geschäftsvorfall HKAZS ist im Segment HIUPD mitzuteilen.

Für den Einsatz von AZS-Verfahren gelten zusätzlich die folgenden allgemeinen Festlegungen:

- 1 bis 99 unterschiedliche AZS-Verfahren pro Institut[1]
  1 bis 9 unterschiedliche AZS-Verfahren pro Benutzer
- Zur eindeutigen Bezeichnung von AZS-Verfahren wird das Element „Sicherheitsfunktion, kodiert“ um den zusätzlichen Nummernkreis 800 bis 899 erweitert.
  Auch die Verknüpfung von Code und Verfahren ist Teil dieser Spezifikation und wird in der BPD festgelegt.
- Mit dem Rückmeldungscode 3921 und Rückmeldeparametern (vgl. [RMCode]) werden dem Kunden in der Dialoginitialisierungsantwort die für ihn zugelassenen AZS-Verfahren mitgeteilt. Als Bezugssegment für das Rückmeldungssegment HIRMS wird HKVVB (Verarbeitungsvorbereitung) verwendet.

  Der Kunde übermittelt im Signaturkopf der Dialoginitialisierungsnachricht, mit welchem konkreten AZS-Verfahren er den Dialog führen will. Das konkrete AZS-Verfahren darf während des Dialogs nicht gewechselt werden (Näheres hierzu siehe Abschnitt C).
- Beim Einsatz von Mehrfach-Signaturen gilt ein konkretes AZS-Verfahren für den gesamten Dialog des jeweiligen Benutzers. Jeder Benutzer kann ein eigenes konkretes AZS-Verfahren verwenden, dieses darf im Kontext einer Mehrfach-Signatur-Einreichung jedoch nicht gewechselt werden.

[1] Aktuell ist nur ein AZS-Verfahren für fortgeschrittene Signaturen (S-Fkt=811) spezifiziert

Im Falle eines nicht zugelassenen Wechsels des AZS-Verfahrens muss das Kreditinstitut den Dialog mit Rückmeldungscode 9957 „Wechsel des AZS-Verfahrens bei Mehrfach-Signaturen nicht erlaubt“ beenden.

- Die Signierung von Kreditinstitutsnachrichten wird in AZV-Verfahren momentan nicht unterstützt.

B.2 AZS-Verfahren zur Secoder-Integration

Im Folgenden wird ausschließlich das derzeit spezifizierte AZS-Verfahren 811 betrachtet. Secoder-Kryptogramme (Definition vgl. Abschnitt „Begriffe“) ersetzen die ansonsten in FinTS verwendeten HBCI-Signaturen, d. h. der Hashwert über die zu signierenden Auftragsdaten wird in der FinTS-Instanz gebildet, an den Secoder übertragen, mit Visualisierungsdaten angereichert und dort signiert. In einem zweiten Aufruf des Secoders werden zur „Visualisation Authentication“ die Visualisierungsdaten zusammen mit einem definierten Visualisierungsbestätigungssignaturfüllwert (Konstante) signiert. Der Visualisierungsbestätigungssignaturfüllwert bewirkt, dass institutsseitig eindeutig festgestellt werden kann, dass der Secoder sich zum Zeitpunkt der Signaturbildung im Applikationsmodus befand, falls ein Secoder verwendet wurde (vgl. [Secoder]). Ein konkretes AZS-Verfahren bezeichnet in diesem Sinne also ein konkretes durch den Secoder durchzuführendes und dort definiertes, ein-schrittiges Signaturverfahren.

Die Arbeitsweise des Secoders sieht zur Signaturbildung vor, dass zusammen mit dem Hashwert über die Auftragsdaten eine Auswahl von Transaktionsdaten in Form von Secoder-spezifischen Kommandos (SecCmds) z. B. über die USB-Schnittstelle an den Secoder gesendet werden. Befindet sich der Secoder im Applikationsmodus – d. h. ist z. B. beim AZS-Verfahren 811 die Signatur-Applikation „aut“ selektiert – so werden die relevanten Daten im Secoder-Display angezeigt. Bestätigt der Kunde diese Daten mit Hilfe der Secoder-Tastatur, so gehen diese zusammen mit anderen Informationen in die Bildung der Signatur mit ein. Die Signatur selbst wird als Antwort an die aufrufende Secoder-Anwendungsfunktion im PC zurückgegeben und kann dort in das FinTS-Protokoll (DE „Signaturdaten“ in HKAZS) integriert werden.

B.2.1 AZS-Verfahren

Beim Ein-Schritt-Verfahren werden Daten und sämtliche Sicherheitselemente in einem Dialogschritt an das Kreditinstitut gesendet und von dort beantwortet. Alle Informationen zur Absicherung des Auftrags wie z. B. die am Secoder zu signierenden Auftragselemente sind lokal in der FinTS-Instanz über die BPD bekannt; es wird also kein zwei-schrittiges Challenge-Response-Verfahren wie bei chipTAN oder mobile TAN (vgl. [PINTAN]) benötigt.

Mit AZS-Verfahren werden keine Verfahren konkret spezifiziert – es erfolgt nur eine abstrakte Definition des Ablaufs, der über Parameter gesteuert wird. Der Ablauf selbst ist für alle AZS-Verfahren identisch. Die Parametrisierung eines konkreten AZS-Verfahrens erfolgt über die neuen Parametersegmente HIAZSS (Geschäftsvorfallparameter zu „Alternative ZKA-Sicherheitsverfahren“ HKAZS) und HIVISS (reines Parametersegment zur elementweisen Definition der Secoder-Visualisierungsdaten).

Bei Verwendung von Mehrfach-Signaturen wird innerhalb eines Ablaufs das AZS-Verfahren durch den Dialogführer des ersten (und ggf. einzigen) Dialogs für alle beteiligten Benutzer festgelegt.

Durch Verwendung des Parametersegmentes HIAZSS ist die abstrakte Beschreibung aller verfügbaren konkreten AZS-Verfahren in der BPD möglich, die über das Datenelement „Sicherheitsfunktion, kodiert“ referenziert werden (Details siehe Kapitel B.5“). Einem Benutzer können maximal 9 konkrete AZS-Verfahren zugeordnet werden. Bei der Verwendung von Mehrfach-Signaturen kann jeder beteiligte Benutzer ein eigenes konkretes AZS-Verfahren verwenden – die Verfahren können also innerhalb einer Nachricht unterschiedlich sein[2].

Mit dem Sicherheitsverfahren PIN/TAN wurden zwei Prozessvarianten eingeführt, welche bereits zu unterschiedlichen Belegungen der Elemente im Geschäftsvorfall HKTAN geführt haben. Entsprechend existieren dort auch zugehörige TAN-Prozesse, um die einzelnen Schritte zu kennzeichnen.

Im Rahmen der AZV-Verfahren ist im übertragenen Sinne nur die Prozessvariante 2 relevant. Analog zu den TAN-Prozessen bei PIN/TAN werden dort Signatur-Prozesse eingesetzt. Beim Ein-Schritt-AZS-Verfahren gemäß Sicherheitsfunktion 811 werden nur die Signatur-Prozesse 6 und 7 benötigt. Die Belegung der Elemente im Geschäftsvorfall HKAZS ist abhängig von der Charakteristik des konkret verwendeten AZS-Verfahrens (gekennzeichnet durch die Sicherheitsfunktion) und den zugehörigen Signatur-Prozessen.

AZS-Verfahren 811:

Bei dieser Variante wird der Auftrag inklusive aller Sicherheitsinformationen in einem Dialogschritt an das Kreditinstitut übertragen. Es existieren folgende Signatur-Prozesse.

[2] Da es im aktuellen Dialog nur einen Dialogführer geben kann, müssen die zulässigen konkreten AZS-Verfahren der weiteren Benutzer bereits vorab über separate Dialoge (und entsprechende Rückmeldecodes 3921) festgelegt worden sein.

- Signatur-Prozess=6:

  Dient der Einreichung von Auftrag und Secoder-Sicherheitsinformationen (VisDataSig und VisAuthSig) und wird durch das Kreditinstitut beantwortet. Kennzeichen von Signatur-Prozess=6 ist also, dass die Signaturbildung inklusive Secoder-Visualisierung geschieht. Wird nur bei der Auftragsverarbeitung und im Rahmen der Dialoginitialisierung benutzt, wenn diese Visualisierungsdaten enthält.

- Signatur-Prozess=7:

  Über diesen Signaturprozess wird signalisiert, dass es sich um keine Secodersignatur, sondern um eine RDH- bzw. RAH-Signatur ohne Secoder-Visualisierung handelt.

  Dies ist z. B. bei der Dialoginitialisierung ohne Visualisierungsdaten und bei Key-Management-Geschäftsvorfällen wie der Schlüsseleinreichung und –Änderung der Fall (vgl. Abschnitt C.1.3 „Key-Management bei AZS-Verfahren“).

  Dieser Signaturprozess kann auch bei einer späteren Verwendung von Instituts-signaturen genutzt werden. Diese werden immer durch Signatur-Prozess=7 gekennzeichnet, da Instituts-seitig keine (Secoder-)Visualisierung verwendet wird.

B.2.2 Abläufe für AZS-Verfahren

Die Abläufe zur Abwicklung der unterschiedlichen AZS-Verfahren unterscheiden sich je nach gewählter Variante. Konkret ist in der vorliegenden Version nur der folgende Ablauf unterstützt:

	S-Fkt.	Sicherheitsverfahren mit Secoder:
Ablauf 1:	811	Nutzung der Secoder-Applikation „aut“ mit Ein-Schritt-AZS-Verfahren ohne Institutssignatur

Die FinTS-Sicherheitssegmente Signaturkopf und Signaturabschluss können nicht für den Transport der Kryptogramme verwendet werden, da es sich um andere syntaktische Konstrukte handelt. Daher werden die FinTS Sicherheitssegmente teilweise mit speziellen Defaultwerten belegt (vgl. Abschnitt C.2).

Da die normale FinTS Dialoginitialisierung ebenfalls nicht für den Transport der kryptografischen Informationen verwendet werden kann, werden auch dort die Sicherheitssegmente teilweise mit Defaultwerten belegt, wenn im Rahmen der Dialoginitialisierung eine Visualisierung im Secoder erfolgen soll. Bei Sicherheitsfunktion 811 werden bei der Dialoginitialisierung ohne Visualisierung und der Schlüsseleinreichung oder –Änderung die üblichen HBCI RDH- bzw. RAH-Sicherheitsverfahren verwendet (vgl. Abschnitt C.1.3 „Key-Management bei AZS-Verfahren“).

Alle Abläufe sind bezogen auf die einzelnen Prozessschritte exakt in der beschriebenen Form umzusetzen; die Bildung von anderen Derivaten ist nicht zugelassen. Die Dialogendenachricht und die darauf folgende allgemeine Kreditinstitutsnachricht werden aus Gründen der Übersichtlichkeit in den Prozessen nicht dargestellt. Bei der Dialogendenachricht werden im Gegensatz zur normalen FinTS-Verarbeitung die AZS-Signaturen weggelassen, wie in Abschnitt Behandlung der Dialogendenachricht (HKEND) C.1.4 beschrieben.

Bei allen Abläufen wird davon ausgegangen, dass sich nur ein signaturpflichtiger FinTS-Auftrag in der Nachricht befindet. Dabei kann es sich auch um einen Sammelauftrag handeln.

In einem Dialog ist es grundsätzlich möglich aber nicht verpflichtend, dass mehrere in sich abgeschlossene Abläufe hintereinander durchgeführt werden.

Es gelten hierbei als Rahmenbedingungen die für den gesamten Dialog getroffenen Festlegungen, z. B. dass die Sicherheitsfunktion innerhalb eines Dialoges nicht gewechselt werden darf.

Bei der Verwendung von Mehrfach-Signaturen sind Aufträge, bei denen mindestens ein Secoder-Kryptogramm fehlerhaft ist, Kreditinstituts-seitig zu verwerfen.

B.2.2.1Sicherheitsverfahren mit Secoder

Beim FinTS-Sicherheitsverfahren HBCI unter Verwendung des Secoders werden die Prozesse des bestehenden Zwei-Schritt-TAN-Verfahrens sinngemäß übernommen und als AZS-Verfahren verfügbar gemacht. Bei Einsatz des Secoders existiert zurzeit nur eine Ausprägung der „Sicherheitsfunktion, kodiert“ mit S-Fkt=811 mit folgender Charakteristik:

S-Fkt	Segment	Bedeutung
811	Alternative ZKA Sicherungsverfahren: - HKAZS ergänzend zum Signaturabschluss, - HIAZSS und HIVISS Verfahrensparameter	Fortgeschrittene Elektronische Signatur („AUT-Signatur“) mit Secoder ohne Institutssignatur

B.2.2.1.1 Fortgeschrittene Elektronische Signatur mit Secoder (S-Fkt=811)

Bei diesem Szenario wird von einem ein-schrittigen Verfahren ausgegangen, d. h. die im Secoder zu visualisierenden Daten müssen dem Kundensystem über die BPD (HIAZSS und HIVISS) mitgeteilt werden.

Es wird davon ausgegangen, dass der Benutzer vor der ersten Verwendung dieses AZS-Verfahrens vollständig initialisiert ist, d. h. es muss über das Standard RDH- bzw. RAH-Verfahren (S-Fkt=1, Signaturübertragung im FinTS Signaturabschluss) ein Schlüsselaustausch erfolgt sein (vgl. hierzu auch Abschnitt C.1.3 „Key-Management bei AZS-Verfahren“).

Die Kreditinstitutsantwort enthält bei S-Fkt=811 keine Institutssignatur.

B.2.2.1.1.1 Dialoginitialisierung

Im Rahmen der FinTS-Dialoginitialisierungselemente werden nur die administrativen Informationen verwendet. Zusammen mit der Dialoginitialisierung wird ein Geschäftsvorfall HKAZS eingereicht, um die Kunden-seitige Authentikation durchzuführen.

Bei der Dialoginitialisierung werden wahlweise Daten im Secoder visualisiert und es erfolgt eine fortgeschrittene Signatur über die Anmeldedaten und ggf. Visualisierungsdaten mit der Secoder-Anwendung „aut“.

Abhängig vom Vorhandensein einer Visualisierung im Secoder wird der Signatur-Prozess 6 oder 7 genutzt. Im folgenden Prozess-Beispiel wird davon ausgegangen, dass im Rahmen der Dialoginitialisierung Daten im Secoder angezeigt und bestätigt werden sollen (Signatur-Prozess=6).

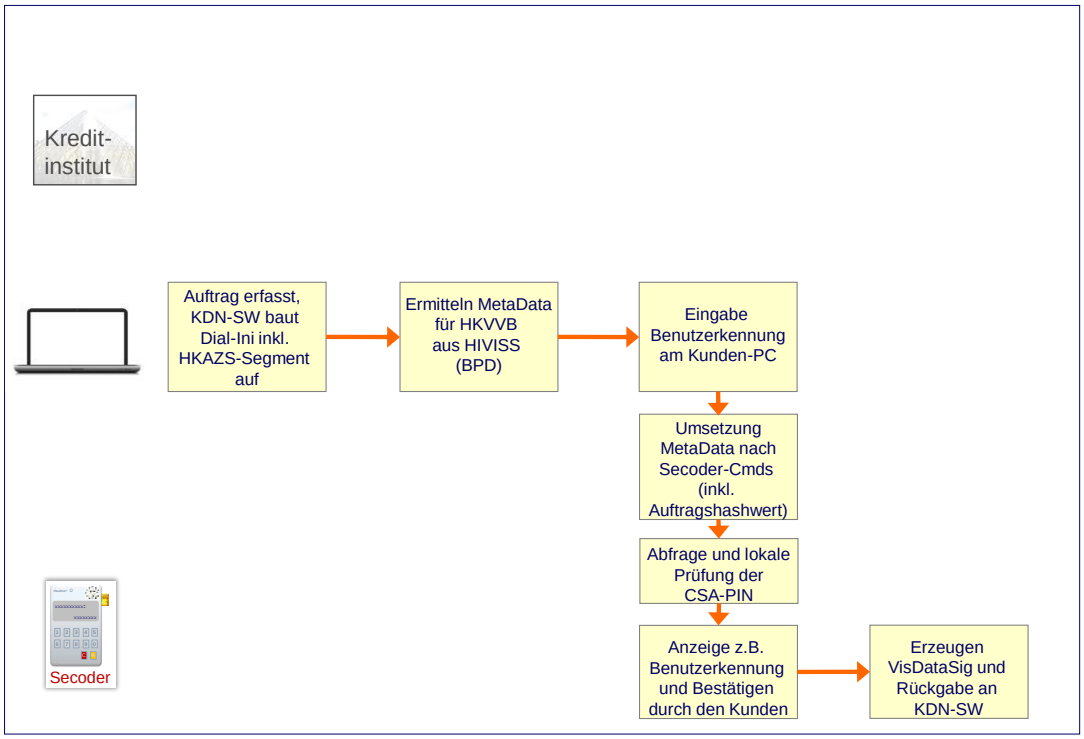


Abbildung 2: Dialoginitialisierung beim AZS-Verfahren S-Fkt=811 (1 von 2)

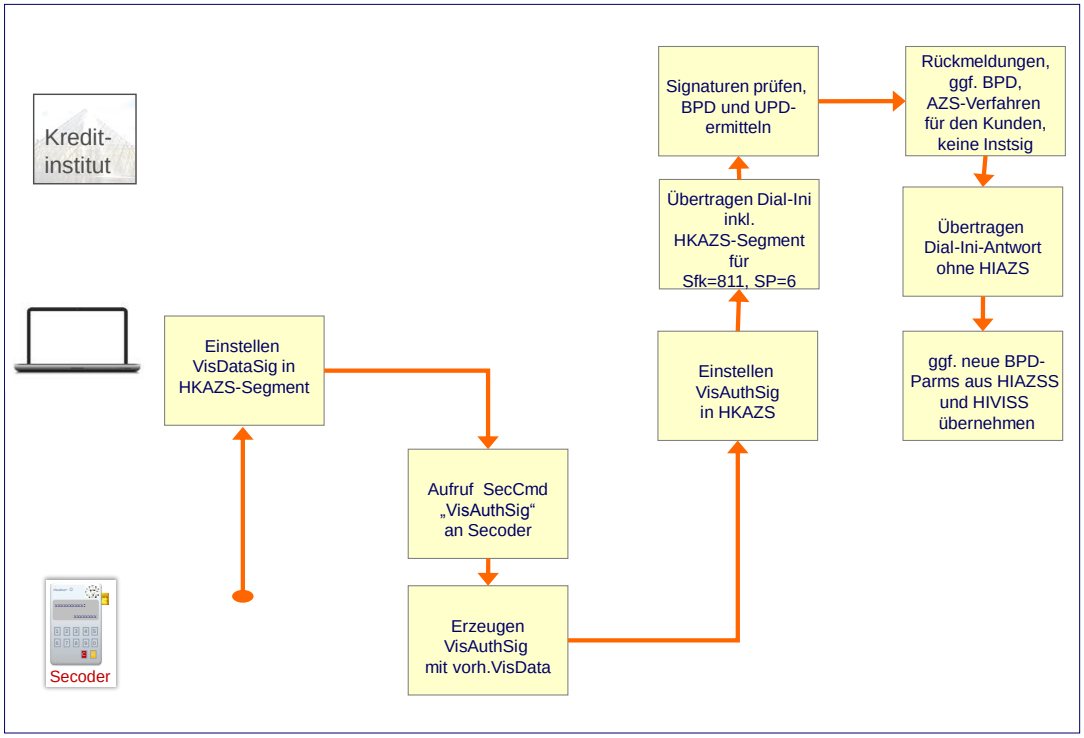


Abbildung 3: Dialoginitialisierung beim AZS-Verfahren S-Fkt=811 (2 von 2)

Der vollständige Ablauf sieht folgendermaßen aus:

Dialoginitialisierung bei Sicherheitsfunktion 811

Ausgangszustand:

- Der Kunde ist vollständig initialisiert und seine Karte ist frei geschaltet (über Zertifikate oder Ini-Brief-Austausch). Dieser Vorgang wird über das Standard RDH- bzw. RAH-Verfahren mit Sicherheitsfunktion=1 durchgeführt.
- Die Dialoginitialisierungsnachricht wird mit Defaultwerten gesendet; der Kunde hat dort durch entsprechende Belegung des DE „Sicherheitsfunktion, kodiert“=811 ein konkretes Ein-Schritt-Verfahren für den gesamten Dialog gewählt.
- Im Kundensystem ist eine aktuelle BPD gespeichert, aus der die Steuerungsinformationen zum Aufbau der MetaData ermittelt werden können. Ggf. muss die FinTS-Instanz die aktuelle BPD über einen anonymen Dialog ermitteln.

Schritt 1a HKIDN, HKAZS	→	Kundenauthentikationsdaten senden Das Kundensystem baut aus den Anmeldeinformationen des HKIDN (Benutzerkennung, KundenID, usw.) eine Struktur auf. Als Signaturalgorithmus wird ein geeignetes (BPD) Sicherheitsprofil selektiert. Über die Segmente Signaturkopf, Identifikation, Verarbeitungsvorbereitung und ggf. Anforderung öffentlicher Schlüssel wird entsprechend den Angaben im Sicherheitsprofil, das in den Verschlüsselungskopf eingestellt wird, der Auftragshashwert erzeugt. Falls die Länge des Hashwerts nicht der Blocklänge entspricht, wird dieser wird mit Nullen aufgefüllt. Es erfolgt ein Zugriff auf den Secoder im Default-Modus. Beim ersten Zugriff wird der Kunde aufgefordert, sein CSA-Passwort einzugeben. Der PIN Sicherheitszustand wird über den gesamten Ablauf hinweg gehalten, außer er wird durch eine Kundenaktion unterbrochen. Auch bei dem Wechsel zwischen Transparent- und Applikationsmodus im Secoder muss das CSA-Passwort nicht neu eingegeben werden. Anschließend wird der aktuelle Signaturzähler des Signier-Schlüssels ermittelt, inkrementiert und in das Datenelement Sicherheitsreferenznummer im Signaturkopf eingestellt. Auch die restlichen benötigten Werte wie z. B. CID, Schlüsselnummer, Schlüsselversion und ggf. Kunden-Zertifikat des Signierschlüssels werden auf diese Weise von der Karte gelesen und für die spätere Auftragsverarbeitung im Kundensystem gespeichert. Auf Basis der Informationen aus HIAZSS und HIVISS erzeugt das Kundensystem die benötigten SecCmds, um Visualisierungsinformationen und den Auftragshashwert zu übergeben und am Secoder im Applikationsmodus „aut“ eine VisData-Signatur mit Visualisierung der relevanten Daten aus HIVISS durchzuführen. Hierbei wird der VisData-Puffer sukzessive mit den Ergebnissen des Visualisierungsprozesses aufgefüllt. Nach der letzten Bestätigung des Kunden wird – beginnend mit dem Auftragshashwert, der als Initialwert zwischengespeichert wurde – in der Karte ein Hashwert über den VisData-Puffer gebildet. Das Ergebnis dieser Hashoperation wird an den Secoder zurückgegeben und ersetzt den Inhalt des VisData-Puffers.

		Über den erzeugten Hashwert wird dann eine VisData-Signatur mit dem Signierschlüssel gebildet und vom Secoder an die aufrufende Applikation zurückgegeben. Die Signatur wird dort in die bereits vorbereitete HKAZS-Struktur in das Datenelement Signaturdaten eingestellt. Das Kundenprodukt ruft den Secoder nochmals im Applikationsmodus „aut“ zur Visualisierungsbestätigung auf (VisAuthSig) und übergibt hierzu den in der BPD festgelegten Visualisierungsbestätigungssignaturfüllwert (Einzig möglicher Wert ist "SECODERSECODERSECODE"). Der noch im VisData-Puffer vorhandene Hashwert über die Secoder-Visualisierungsdaten wird an den ersten Stellen mit dem Visualisierungsbestätigungssignaturfüllwert überschrieben und über dieses Resultat eine VisAuth-Signatur gebildet. Die erzeugte VisAuth-Signatur wird an das Kundenprodukt zurückgegeben und dort in das Datenelement Visualisierungsbestätigungssignaturdaten des HKAZS eingefügt. Die fachlichen Segmente (HKIDN, HKVVB …) werden zusammen mit HKAZS mit der Belegung gemäß Signatur-Prozess=6 inklusive VisDataSig und VisAuthSig HBCI-verschlüsselt zum Institut übertragen.
Schritt 1b HIAZS	←	Authentikationsantwort senden Nach Überprüfung von VisDataSig und VisAuthSig wird eine Dialoginitialisierungsantwort aufgebaut. Da bei S-FKT=811 keine Institutssignatur verwendet wird, wird die Nachricht unsigniert aber HBCI-verschlüsselt an das Kundenprodukt übertragen. Das Kundensystem wertet nach der Entschlüsselung die Kreditinstitutsantwort aus und zeigt dem Kunden die Begrüßungsseite an.

B.2.2.1.1.2 Auftragseinreichung beim Ein-Schritt-AZS-Verfahren

Bei diesem Szenario werden die SecCmds von der aktiven Kundenkomponente auf Basis des zugrunde liegenden Auftrags offline erzeugt und an den Secoder übertragen. Der Kunde fügt – nach Kontrolle und Bestätigung des Datenextraktes im Secoder-Display - eine fortgeschrittene Signatur an. Diese wird zusammen mit dem Auftrag an das Institut gesendet.

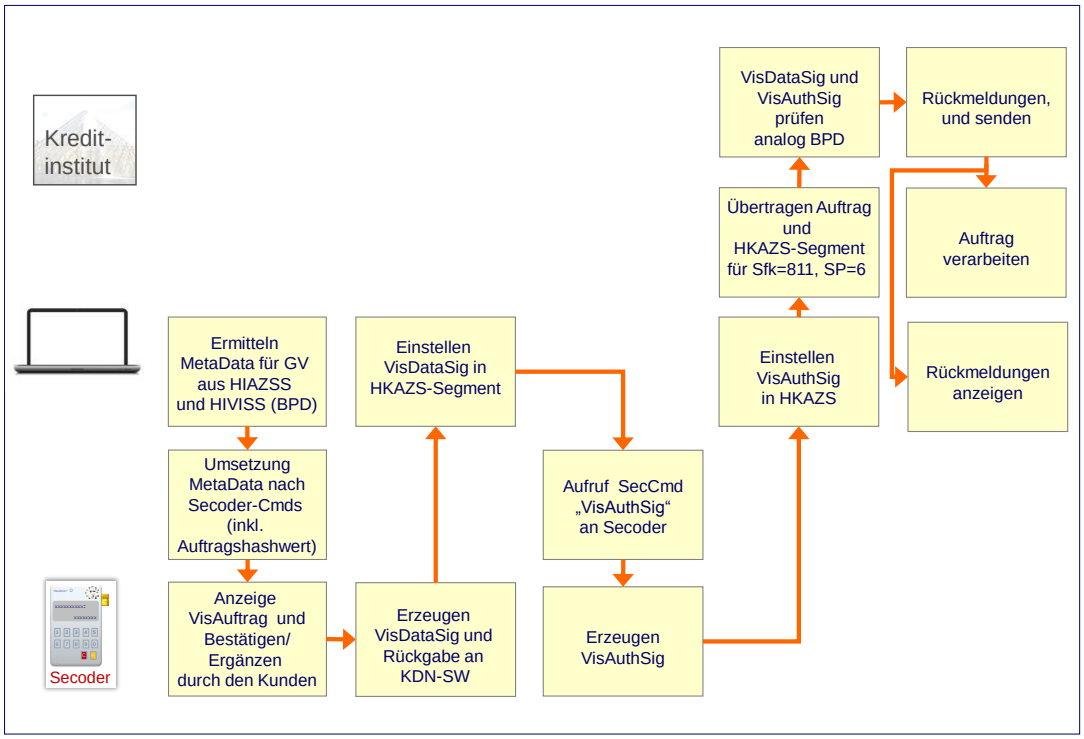


Abbildung 4: Fortgeschrittene Signatur mit Secoder, S-Fkt=811

Auftragseinreichung bei Sicherheitsfunktion=811 Ausgangszustand: • Die Dialoginitialisierung ist erfolgt; der Kunde hat dort durch entsprechende Belegung des DE „Sicherheitsfunktion, kodiert“=811 ein konkretes Zwei-Schritt-Verfahren für den gesamten Dialog gewählt.		
Schritt 1a HKUEB, HKAZS	→	Auftrag einreichen Im Kundenprodukt wird eine HKAZS-Struktur aufgebaut. Hierbei wird in das Datenelement Signaturreferenznummer im Signaturkopf der zwischengespeicherte und um 2 (VisDataSig und VisAuthSig) inkrementierte Signaturzähler eingestellt. Das Kundenprodukt erzeugt auf Basis des Auftrags und der BPD die benötigten SecCmds und sendet diese zusammen mit dem Auftragshashwert über Signaturkopf und fachliches Segment (z. B. HKUEB) an den Secoder im Applikationsmodus „aut“. Falls die Länge des Hashwerts nicht der Blocklänge entspricht, wird dieser wird mit Nullen aufgefüllt. Wie bei der Dialoginitialisierung beschrieben wird auch hier der VisData-Puffer sukzessive mit Informationen aus dem Secoder-Visualisierungsprozess gefüllt. Aus den Ergebnissen wird – mit dem zwischengespeicherten Hashwert über die Auftragsdaten startend – ein Hashwert über die Visualisierungsdaten gebildet, der den Inhalt des VisData-Puffers ersetzt. Über diesen Hashwert wird mit Hilfe des Signierschlüssels eine VisData-Signatur erzeugt. Die Kontrolle wird an das aufrufende Programm zurückgegeben

		und die VisData-Signatur in das DE Signaturdaten in HKAZS eingestellt. Das Kundenprodukt ruft den Secoder nochmals im Applikationsmodus „aut“ zur Visualisierungsbestätigung auf (VisAuthSig) und übergibt hierzu den in der BPD festgelegten Visualisierungsbestätigungssignaturfüllwert (Einzig möglicher Wert ist "SECODERSECODERSECODE"). Der noch im VisData-Puffer vorhandene Hashwert über die Secoder-Visualisierungsdaten wird an den ersten Stellen mit dem Visualisierungsbestätigungssignaturfüllwert überschrieben und über dieses Resultat eine VisAuth-Signatur gebildet. Die erzeugte VisAuth-Signatur wird an das Kundenprodukt zurückgegeben und dort in das Datenelement Visualisierungsbestätigungssignaturdaten des HKAZS eingefügt. Der fachliche Geschäftsvorfall (z. B. HKUEB) wird zusammen mit HKAZS mit der Belegung gemäß Signatur-Prozess=6 inklusive VisDataSig und VisAuthSig HBCI-verschlüsselt zum Institut übertragen.
Schritt 1b z. B. HIRMS zu HKUEB	←	Rückmeldungen senden Nach erfolgreicher Signaturprüfung kann der Auftrag verarbeitet werden. Es wird eine Auftragsantwort aufgebaut, die ggf. erzeugte Antwortsegmente sowie die Rückmeldungen zur Signatur-Prüfung und zum Auftrag selbst enthält Da bei S-FKT=811 keine Institutssignatur verwendet wird, wird die Nachricht unsigniert aber HBCI-verschlüsselt an das Kundenprodukt übertragen. Das Kundensystem wertet nach der Entschlüsselung die Kreditinstitutsantwort aus.

B.3 Geschäftsvorfall HKAZS für AZS-Verfahren

Dieser Geschäftsvorfall dient im Ein-Schritt-Verfahren als Ergänzung des FinTS-Segmentes Signaturabschluss eine fortgeschrittene Signatur zu transportieren.

Der Geschäftsvorfall HKAZS nimmt in FinTS eine Sonderrolle ein: HKAZS muss in BPD, UPD wie ein Geschäftsvorfall aufgeführt werden und besitzt mit HIAZSS auch Geschäftsvorfallparameter. Als Sonderbedingung wird HKAZS jedoch wie ein administratives Segment bei der Zählung im DE „Maximale Anzahl Aufträge“ pro Nachricht (vgl. [Formals], Kapitel D.6) nicht berücksichtigt.

Durch Existenz dieses Geschäftsvorfalls HKAZS in der BPD und UPD wird grundsätzlich festgelegt, ob das Kreditinstitut AZS-Verfahren unterstützt bzw. ob dies für den Kunden zugelassen ist.

Zusammen mit der Kreditinstitutsrückmeldung können abhängig vom verwendeten fachlichen Geschäftsvorfall auch Antwortsegmente zu diesem Auftrag übertragen werden.

B.3.1 Aufbau des Geschäftsvorfalls HKAZS

Der Geschäftsvorfall HKAZS wird verwendet, um beliebige digitale Signaturen von DK-Verfahren, die nicht das HBCI-Format verwenden, inklusive der zugehörigen Visualisierungsdaten transportieren zu können. Ausprägungen solcher Signaturverfahren können sein (aktuell ist nur S-Fkt=811 unterstützt):

- Fortgeschrittene Elektronische Signaturen mit Secoder (S-Fkt=811) Signaturbelegung nach HBCI

Sonderbelegungen der HBCI-Sicherheitssegmente für alle AZS-Verfahren sind in den Abschnitten B.5.2 und C enthalten.

Realisierung Bank: verpflichtend, wenn Geschäftsvorfälle mit Absicherung über alternative Signaturverfahren angeboten werden

Realisierung Kunde: verpflichtend, wenn Geschäftsvorfälle mit Absicherung über alternative Signaturverfahren unterstützt werden sollen

a) Kundenauftrag

◆ **Format**

Name: Alternative ZKA Sicherheitsverfahren

Typ: Segment

Segmentart: Geschäftsvorfall

Kennung: HKAZS

Bezugssegment: -

Segmentversion: 1

Sender: Kunde

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Segmentkopf	DEG			M	1	
2	Signatur-Prozess	DE	code	1	M	1	6, 7
3	Sicherheitsprofil	DEG			C	1	M: bei S-Fkt=811 N: sonst
4	Sicherheitskontrollreferenz	DE	an	..14	M	1	
5	SAK-Visualisierungsdaten	DE	bin	..2MB	C	1	M: nicht unterstützt N: sonst
6	Visualisierungsbestätigungssignaturdaten	DE	bin	..4KB	C	1	M: bei Signatur-Prozess=6 N: sonst
7	Signaturdaten	DE	bin	..4KB	C	1	M: bei Signatur-Prozess=6 und 7 N: sonst
8	Auftragsreferenz	DE	an	..35	C	1	M: nicht unterstützt N: sonst
9	Weitere Signatur folgt	DE	jn	1	C	1	M: nicht unterstützt N: sonst
10	Auftrag stornieren	DE	jn	1	C	1	O: nicht unterstützt N: sonst
11	SAK-Referenz	DE	num	..2	C	1	M: nicht unterstützt N: sonst
12	Aufsetzpunkt	DE	an	..35	O	1	

◆ **Belegungsrichtlinien**

Signaturdaten

Im Datenelement Signaturdaten befindet sich im Element „Signaturdaten“ die VisData-Signatur.

Ausnahme: Für Key-Management-Nachrichten oder Auftragsdaten ohne Visu-

alisierung wird eine RDH- bzw. RAH-Signatur mit Signatur-Prozess 7 verwendet (vgl. Abschnitt C.1.3).

b) Kreditinstitutsrückmeldung

- **Format**

Name: Alternative ZKA Sicherheitsverfahren Rückmeldung

Typ: Segment

Segmentart: Geschäftsvorfall

Kennung: HIAZS

Bezugssegment: HKAZS

Segmentversion: 1

Anzahl: 1

Sender: Kreditinstitut

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Segmentkopf	DEG			M	1	
2	Signatur-Prozess	DE	code	1	M	1	7
3	SAK-Visualisierungsdaten	DE	bin	..2MB	C	1	O: nicht unterstützt N: sonst
4	Signaturdaten	DE	bin	..4KB	C	1	M: nicht unterstützt N: sonst
5	Auftragsreferenz	DE	an	..35	C	1	M: nicht unterstützt N: sonst
6	Datum, Angebot gültig bis	DE	dat		C	1	O: nicht unterstützt N: sonst
7	Uhrzeit, Angebot gültig bis	DE	tim		C	1	O: nicht unterstützt N: sonst

- **Ausgewählte Beispiele für Rückmeldungscodes**

Code	Beispiel für Rückmeldungstext
0010	Auftrag entgegengenommen
3921	Zugelassene AZS-Verfahren für den Benutzer (+ Rückmeldungsparameter)
9210	Auftrag abgelehnt – Auftragsdaten inkonsistent. Eingereichter Auftrag gelöscht
9210	Auftrag abgelehnt – Kein eingereichter Auftrag gefunden
9210	Auftrag abgelehnt – Auftragsreferenz ist unbekannt
9380	Gewähltes AZS-Signatur-Verfahren nicht zulässig
9931	Sperrung des Kontos nach x Fehlversuchen
9941	Signatur ungültig

9350	Zertifikat abgelaufen
9351	Zertifikat gesperrt
9352	Zertifikatseigner unbekannt
9359	OCSP-Anfrage nicht beendet
9330	Schlüsseleigner gesperrt
9330	Schlüssel gesperrt
9340	Signatur fehlerhaft
9340	Sicherheitsprofil unbekannt
9360	Signatur fehlerhaft - BPD nicht mehr aktuell

c) Bankparameterdaten

- **Format**

Name: Alternative ZKA Sicherheitsverfahren, Parameter

Typ: Segment

Segmentart: Geschäftsvorfallparameter

Kennung: HIAZSS

Bezugssegment: HKVVB

Segmentversion: 1

Sender: Kreditinstitut

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Segmentkopf	DEG			M	1	
2	Maximale Anzahl Aufträge	DE	num	..3	M	1	
3	Anzahl Signaturen mindestens	DE	num	1	M	1	0, 1, 2, 3
4	Sicherheitsklasse	DE	code	1	M	1	0, 1, 2, 3, 4
5	Parameter Alternative ZKA Sicherheitsverfahren	DEG			M	1	

B.4 Erweiterung der Rückmeldungscodes

Bei Verwendung des AZS-Verfahrens können spezielle Rückmeldecodes vom Kreditinstitut zurückgemeldet werden, die rein AZS-spezifisch sind und u. U. nicht direkt mit dem zugehörigen Geschäftsvorfall in Verbindung stehen. Eine Beschreibung aller Rückmeldungscodes befindet sich in [RMCode].

Es handelt sich hierbei um die folgenden Codes:

Erfolgsmeldungen

Code	Beispiel für Rückmeldungstext
0010	Auftrag entgegengenommen

Warnungen und Hinweise

Code	Beispiel für Rückmeldungstext
3918	Kompetenz nicht ausreichend – weitere Signatur erforderlich
3921	Zugelassene AZS-Verfahren für den Benutzer (+ Rückmeldungsparameter)
3950 -999	Individuell

Fehlermeldungen

Code	Beispiel für Rückmeldungstext
9210	Auftrag abgelehnt – Auftragsdaten inkonsistent. Eingereichter Auftrag gelöscht
9210	Auftrag abgelehnt – Kompetenz nicht ausreichend
9330	Schlüsseleigner gesperrt
9330	Schlüssel gesperrt
9340	Signatur fehlerhaft
9340	Sicherheitsprofil unbekannt

Code	Beispiel für Rückmeldungstext
9380	Gewähltes ZKA-Signatur-Verfahren nicht zulässig
9931	Sperrung des Kontos nach x Fehlversuchen
9931	Teilnehmersperre durchgeführt
9941	Signatur ungültig
9350	Zertifikat abgelaufen
9351	Zertifikat gesperrt
9352	Zertifikatseigner unbekannt
9953	Nur ein Signatur-pflichtiger Auftrag pro Nachricht erlaubt
9954	Mehrfach-Signaturen nicht erlaubt
9957	Wechsel des Signatur-Prozesses bei Mehrfach-Signaturen nicht erlaubt
9359	OCSP-Anfrage nicht beendet
9360	Signatur fehlerhaft – BPD nicht mehr aktuell

B.4.1 Beschreibung spezieller Rückmeldungen im AZS-Verfahren

Rückmeldungscode 3921: Zugelassene AZS-Verfahren für den Benutzer (+ Rückmeldungsparameter)

Der Rückmeldungscode 3921 dient dazu, dem Kundenprodukt im Rahmen der Dialoginitialisierungsantwort die für den Benutzer zugelassenen AZS -Verfahren mitzuteilen. Hierzu werden in den Rückmeldungsparametern P1 bis P10 entsprechend den zugelassenen Verfahren („800“ bis „899“) aus HIAZSS maximal zehn mögliche Ein- und Zwei-Schritt-Verfahren transportiert.

Das Kundenprodukt muss – unabhängig vom gewählten Verfahren in „Sicherheitsfunktion, kodiert“ – bei jeder Dialoginitialisierung die vom Institut mit dem Rückmeldungscode 3921 übermittelten Werte P1, ... , P10 prüfen, gegen gespeicherte Informationen vergleichen und diese ggf. aktualisieren.
Sollte das Kundenprodukt in der Dialoginitialisierungsnachricht ein Verfahren wählen, das für den Benutzer nicht bzw. nicht mehr zugelassen ist, so beendet das Kreditinstitut den Dialog mit Rückmeldungscode 9800 in Kombination mit Code 3921 und meldet die aktuell zugelassenen Verfahren in den Parametern P1 bis P10.

Rückmeldungscode 9360: Signatur fehlerhaft – BPD nicht mehr aktuell

Diese Rückmeldung bezieht sich speziell auf die Parametrisierung der Visualisierungsinformationen mit HIAZSS und HIVISS im Rahmen der Dialoginitialisierung. Es wird Instituts-seitig festgestellt, dass die Signaturprüfung fehlschlug, da das Kundensystem über keine aktuelle BPD verfügt. Die Kreditinstitutsnachricht wird analog den Informationen im Verschlüsselungskopf verschlüsselt und nicht signiert.

B.5 Erweiterung der Bank- und Userparameterdaten (BPD / UPD)

Für die Verwendung von AZS-Verfahren müssen dem Kundenprodukt weitere Daten im Rahmen der BPD- bzw. UPD-Segmentfolge übermittelt werden. So ist beispielsweise anzugeben, welche Geschäftsvorfälle über welches AZS-Verfahren abgesichert werden dürfen und wie die Visualisierung zu erfolgen hat. Hierfür existieren zwei zusätzliche Parametersegmente, welche die folgende Information transportieren:

HIAZSS	Parametersegment zu HKAZS In diesem Parametersegment werden grundsätzliche Mechanismen zu den AZS-Verfahren festgelegt: • Welche AZS-Verfahren werden unterstützt? • Welche Geschäftsvorfälle werden pro AZS-Verfahren unterstützt? Die vollständige Beschreibung von HIAZSS befindet sich in Abschnitt B.3.1.
HIVISS	Secoder-spezifische Visualisierungsinformationen nur Parametersegment; enthält pro Geschäftsvorfall / Segmentversion und Sicherheitsfunktion die Informationen, welche Daten am Secoder in welcher Weise zu visualisieren sind

Die Parametersegmente HIAZSS und HIVISS müssen nicht immer paarig auftreten; es sind folgende Konstellationen erlaubt:

GV enthalten in …	
HIAZSS: nein HIVISS: nein	Ist ein Geschäftsvorfall in keinem der beiden Parametersegmente enthalten. so findet weder eine Secoder-Visualisierung, noch eine kundenseitige Signatur statt. Diese Variante ist für das AZS-Verfahren mit Sicherheitsfunktion 811 nicht erlaubt.
HIAZSS: ja HIVISS: nein	Ist ein Geschäftsvorfall nur im Parametersegment HIAZSS enthalten. so findet keine Secoder-Visualisierung statt, es wird jedoch eine kundenseitige Signatur gebildet. Diese Variante ist für die AZS-Verfahren mit Sicherheitsfunktion 811 erlaubt. Bei Sicherheitsfunktion 811 wird im Transparent-Modus des Secoders eine RDH- bzw. RAH-Signatur (ohne Secodervisualisierung) gebildet und als Kundensignatur in das DE „Signaturdaten“ in HKAZS eingestellt. Das Sicherheitsniveau entspricht also dem des HBCI RDH-/RAH-Verfahrens. Es sind keine Interaktionen des Kunden am Secoder erforderlich.
HIAZSS: nein HIVISS: ja	nicht zulässig

HIAZSS: ja HIVISS: ja	Ist ein Geschäftsvorfall in beiden Parametersegmenten vorhanden, so wird auf Basis dieser Informationen eine Secodersignatur mit Visualisierung gebildet. Diese Variante ist für die AZS-Verfahren mit Sicherheitsfunktion 811 erlaubt und stellt dort den Normalfall eines am Secoder zu visualisierenden und zu signierenden Geschäftsvorfalls dar.

B.5.1 Secoder-spezifische Visualisierungsinformationen (HIVISS)

Die für die Definition von AZS-Verfahren notwendige BPD-/UPD-Erweiterung zur Visualisierung von Auftragsdaten am Secoder wird in Form eines speziellen Parametersegmentes realisiert, welches sich auf keinen echten Geschäftsvorfall bezieht, sondern Daten zu allen unterstützten Geschäftsvorfällen aufnehmen kann.

Das Spezialsegment HIVISS wird verwendet, um in die BPD-Segmentfolge Secoder-spezifische Visualisierungsdaten einzufügen. Aufgrund seines Aufbaus analog zu einem Segmentparametersegment wird es von Kundenprodukten, die keine AZS-Verfahren unterstützen, ignoriert, da es sich auf einen ihnen unbekannten Geschäftsvorfall zu beziehen scheint.

Die in HIAZSS aufgeführten Geschäftsvorfälle dürfen vom Kunden in über AZS-Verfahren abgesicherte Nachrichten eingestellt werden, sofern sie in den BPD und UPD als generell erlaubt hinterlegt sind. Alle übrigen Geschäftsvorfälle können mit AZS-Verfahren nicht verwendet werden.

Während HIAZSS die Segmentkennungen aller über AZS-Verfahren abgesicherten Geschäftsvorfälle enthält, treten in HIVISS nur die Geschäftsvorfälle auf, welche eine Visualisierung am Secoder benötigen. Reine Abholaufträge sind i. A. nicht in HIVISS enthalten.

B.5.1.1 Generelles Secoder-Visualisierungskonzept mit HIVISS

Das Visualisierungskonzept ist durch das Design des Secoders strikt vorgegeben. Im Speziellen gelten die Festlegungen der Secoder-Spezifikation [Secoder] und der „User Interface & Implementation Guide“ [Secoder Impl] inkl. der dort skizzierten Beispiele. Aus diesen Beispielen lassen sich für die Parametrisierung in der BPD drei grundsätzliche Anzeigedefinitionen ableiten (die Ausrichtung der Texte in der folgenden Abbildung sind exemplarisch).

Aufgrund der Komplexität der dezentralen Visualisierungsaufbereitung mittels HIAZSS und HIVISS wird dringend empfohlen, dass die Secoder-Anwendungsfunktion, welche für die Umsetzung in die Secoder-Kommandos zuständig ist, über eine Trace-Funktion verfügt, um die Fehlersuche bei abweichenden Signaturergebnissen zu erleichtern.

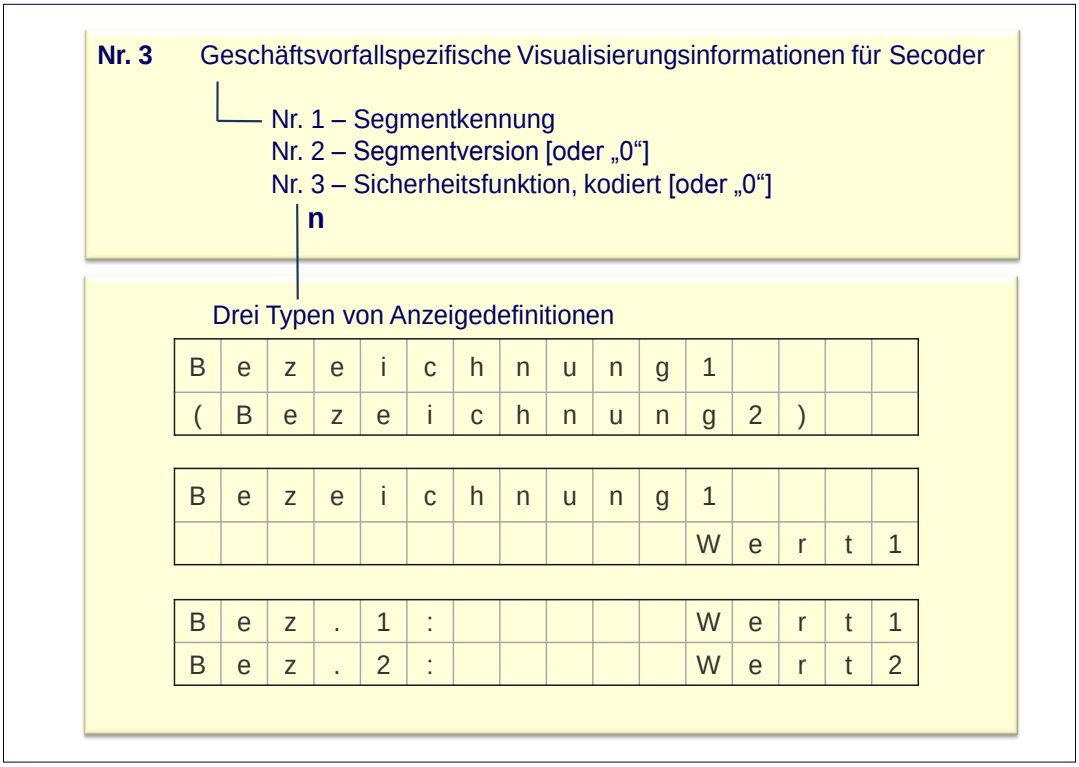


Abbildung 5: Struktur der geschäftsvorfallspezifischen Visualisierungsinformationen

In den folgenden Beispielen zur Visualisierung wird generell von einer heute marktüblichen Displaygröße von 2 x 16 Zeichen ausgegangen, wobei die Physik in der HIVISS-Definition keine Rolle spielt, sondern diese durch die Secoder-Anwendungsfunktion an die physischen Eigenschaften des verwendeten Secoders angepasst wird. Eine HIVISS-Anzeigedefinition umfasst eine Zeile, d. h. 1 x 16 Zeichen. Diese Einheit wird in der Secoder-Spezifikation als „Dataset (DS)“ bezeichnet. Somit beschreibt eine HIVISS-Anzeigedefinition genau ein Secoder Dataset.

Pro Geschäftsvorfall können – abhängig vom zur Verfügung stehenden VisData-Pufferbereich im Secoder von derzeit maximal 512 Byte theoretisch bis zu 255 solcher Datasets verwendet werden.

Die drei oben gezeigten typischen Anzeigedefinitionen haben folgende Bedeutung:

1. Die erste Anzeigedefinition dient der Darstellung von einer oder zwei textuellen Bezeichnungen, die linksbündig dargestellt werden, z. B. die Texte „Einzelüberweis.“ und „Inland“. Als Sonderfall dieser Anzeigedefinition kann die zweite Displayzeile auch leer bleiben.

2. Die zweite Anzeigedefinition kombiniert eine Bezeichnung (z. B. den Text „Ziel-Konto“) in Zeile 1 und einen zugehörigen Wert (z. B. die Kontonummer des Begünstigten aus einem DTA-Satz) in Zeile 2.
3. Mit der 3. Anzeigedefinition ist es möglich, je Zeile eine Bezeichnung und einen Wert darzustellen.

Auf Basis dieser 3 Anzeigedefinitionen ist es möglich, sieben grundsätzliche Szenarien abzubilden, die i. W. den Beispielen aus dem User Interface & Implementation Guide [Secoder Impl] entsprechen. Die Anzeigedefinitionen verfügen über folgenden Befehlsvorrat[4]:

Secoder-visualisierung Index	Index des jeweiligen Secodervisualisierungstextes, der in den „geschäftsvorfallspezifischen Visualisierungsinformationen“ referenziert wird. Default: Keiner
Display-Position	Ausrichtung des Dataset im Secoder-Display (Links, Rechts[5]). Diese Angabe ist nötig, da die Secoder-Texte in der BPD nicht gepadded werden. Default: Links
Länge (Secoder-Text)	Länge des Secoder-Textes Default: 0
Secoder-Text	Secoder-Text, entweder statisch durch die Anzeigedefinition selbst oder dynamisch aus den Auftragsdaten oder leer. Default: leer
Länge Secoder-Eingabedaten	Soll ein Wert nicht nur bestätigt, sondern komplett eingegeben oder ergänzt werden, wird hierdurch die Länge der geforderten Eingabedaten am Secoder vorgegeben. Default: 0
Ausrichtung und Format Secoder-Eingabedaten	Dieser Parameter beschreibt die Ausrichtung bei der Eingabe der Daten (Textmodus oder Taschenrechnermodus) und die Verwendung des Kommas. Default: Textmodus, falls Länge (Secoder-Text) > 0 Taschenrechnermodus, sonst
Secoder-Padding	Dieser Parameter beschreibt das Padding der Daten im Vis-Data-Puffer des Secoders. Default: E0 bei numerischen Daten EF bei alfanumerischen Daten

[4] Aus Gründen der besseren Lesbarkeit – vor allem in den Abbildungen – werden Begriffe wie „Secodervisualisierung Index“ auf „Index“ gekürzt. Die exakten Namen befinden sich im Data Dictionary unter „Secodervisualisierungstexte“.

[5] Die im Secoderkonzept theoretisch mögliche Ausprägung „Mitte“ wird in FinTS nicht benutzt.

Nähere Festlegungen zum konkreten Mapping zwischen HIVISS und Secoder-Kommandos befinden sich in Abschnitt B.5.1.2.

Die Secodervisualisierungstexte beschreiben die Anzeigedefinitionen für n Datasets, die sequentiell am Secoder angezeigt und ergänzt bzw. bestätigt werden, bevor ein Secoder-Kryptogramm gebildet wird.

Die Secodervisualisierung MetaData („Geschäftsvorfallspezifische Visualisierungsinformationen“) enthalten die Definitionen pro Geschäftsvorfall.

Pro Geschäftsvorfall wird über Indizes auf die entsprechenden Secodervisualisierungstexte referenziert, wobei berücksichtigt werden muss, ob eine Abfrage oder Bestätigung ein- oder zweizeilig dargestellt wird.

Da in den Secodervisualisierungstexten im Element Secoder-Text auch Variable – dargestellt durch den Platzhalter „#“ – enthalten sind, enthalten die Geschäftsvorfallspezifischen Visualisierungsinformationen die jeweiligen Werte hierfür (z. B. konkrete Kontonummer aus FinTS, DTA, SEPA …).

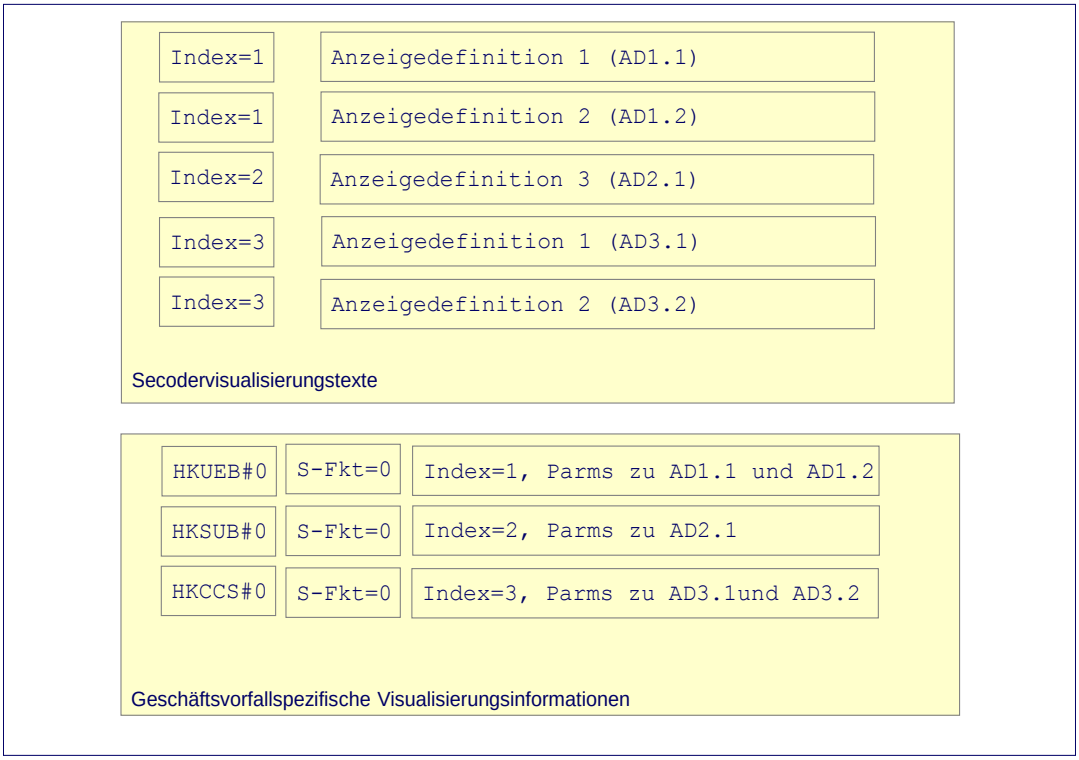


Abbildung 6: Zusammenhang zwischen Secoder MetaData und Secodervisualisierungstexten

B.5.1.1.1 Element „Ausrichtung und Format von Secoder-Eingabedaten“

Die Ausrichtung der Secoder-Eingabedaten kann auf zwei Arten erfolgen:

a) Textmodus

Im Textmodus beginnt die Eingabe direkt hinter dem letzten Zeichen des Secoder-Textes, d. h. der Cursor blinkt an der ersten Stelle rechts vom Secoder-Text. Ist der Secoder-Text leer, blinkt der Cursor im verwendeten Beispiel an Stelle 16 ganz rechts.

Jedes eingegebene Zeichen schiebt den Cursor um eine Position nach rechts, bis die vorgegebene Anzahl von Eingabezeichen erreicht ist. Der Secoder-Text und die bereits eingegebenen Zeichen werden nicht bewegt.

Der Textmodus stellt die Default-Belegung dar, wenn ein Secoder-Text vorhanden ist.

b) Taschenrechnermodus

Im Taschenrechnermodus blinkt der Cursor grundsätzlich im verwendeten Beispiel an Position 16 der Zeile ganz rechts, auch wenn ein Secoder-Text vorhanden ist. Jedes eingegebene Zeichen schiebt die bereits vorhandenen Zeichen (ggf. Secoder-Text und Eingabezeichen) um eine Position nach links.

Der Taschenrechnermodus stellt die Default-Belegung dar, wenn kein Secoder-Text vorhanden ist.

B.5.1.1.2 Repräsentative Beispiele zum Secoder-Visualisierungskonzept

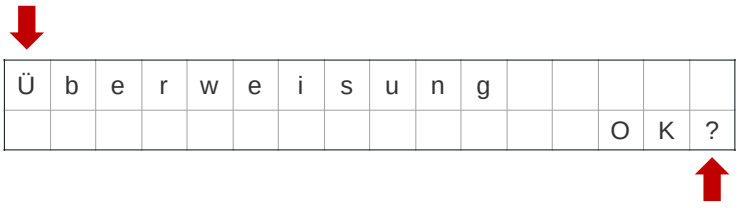
Die folgenden Kapitel beschreiben die Parametrisierung anhand von sieben repräsentativen Konstellationen, wie sie auch im User Interface & Implementation Guide [Secoder_Impl] verwendet werden. Das Mapping der FinTS-Definitionen in die SecCmds ist mit der dort beschriebenen Syntax der Secoder Data Confirmation vorzunehmen.

Bei der Sicherheitsfunktionen 811 kann nur eine Bestätigung von Daten, kein Auffüllen erfolgen.

Bei den Beispielen wird wie in [Secoder_Impl] von einer Displaygröße von 2 x 16 Zeichen ausgegangen.

a) Szenario 1: Bestätigung statischer Werte, 2-zeilig

Szenario 1 dient zur Darstellung statischer Werte im Display des Secoders. Die Anzeigedefinition lässt die Belegung beider Displayzeilen mit je einem maximal 16-stelligen Textstring zu. Die Bezeichnungen werden ohne Leerzeichen eingestellt. Über die Parametrisierung im Element „Display-Position“ wird eine entsprechende Ausrichtung erreicht. In den Beispielen werden wo immer möglich die Defaultwerte verwendet, um eine möglichst kompakte BPD-Modellierung zu erzielen.



Aufbau der beiden zugehörigen Visualisierungstexte:

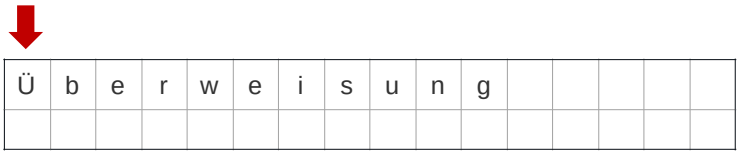
Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
1	**Def**	**Def**	**„Überweisung“**	**Def**	**Def**	**Def**

Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
1	**R**	**Def**	**„OK?“**	**Def**	**Def**	**Def**

Die Bezeichnungen werden ohne Leerzeichen eingestellt. Über die Parametrisierung im Element „Display-Position“ wird eine entsprechende Ausrichtung erreicht. In den Beispielen werden wo immer möglich die Defaultwerte verwendet, um eine möglichst kompakte BPD-Modellierung zu erzielen.

b) Szenario 2: Bestätigung statischer Werte, 1-zeilig

Szenario zwei beschreibt die Darstellung eines 1-zeiligen Textes im Secoder-Display. Der Kunde geht in diesem Fall ohne weitere Benutzerführung davon aus, dass er diesen Text bestätigen muss, bevor die Verarbeitung im Secoder fortgesetzt werden kann.



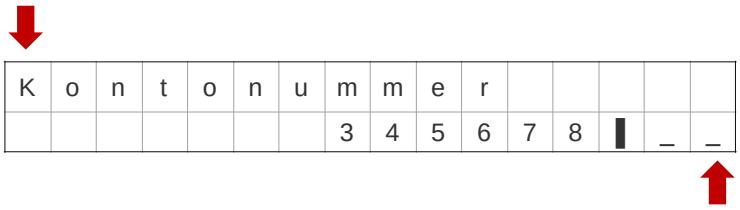
Aufbau des zugehörigen Visualisierungstextes:

Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
2	**Def**	**Def**	**„Überweisung“**	**Def**	**Def**	**Def**

Zu beachten ist, dass in FinTS nur die Secoderoption durch Wegfall des Dataset für die zweite Zeile unterstützt wird, nicht die Möglichkeit ein Dataset mit Länge 0 zu verwenden (dies würde zu einem unterschiedlichen Kryptogramm führen).

c) Szenario 3: Auffüllen mit Ziffern, 2-zeilig

In Szenario 3 wird in Zeile 1 ein statischer Text angezeigt. Zeile 2 enthält einen Teil der Kontonummer (des Begünstigten), die letzten 3 Stellen muss der Kunde ergänzen, bevor er den gesamten Ausdruck betätigen kann.



Aufbau der beiden zugehörigen Visualisierungstexte:

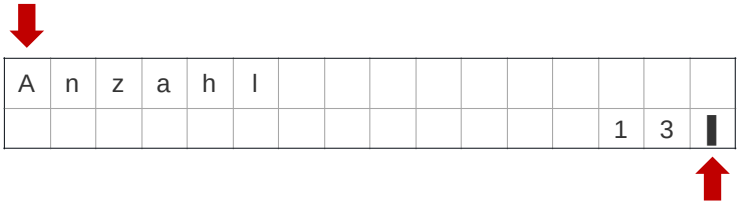
Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
3	**Def**	**Def**	**„Kontonummer“**	**Def**	**Def**	**Def**

Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
3	**R**	**6**	#	**3**	**Def**	**Def**

In Zeile 1 wird der statische Text aus dem Element „Secoder-Text“ angezeigt. Zeile 2 besteht aus einem dynamischen Text, was durch den Platzhalter „#“ im Element „Secoder-Text“ dargestellt ist. Der konkrete Wert für den Platzhalter wird durch die „Geschäftsvorfallspezifische Visualisierungsinformation“ festgelegt.

d) Szenario 4: Eingabe von Ziffern, 2-zeilig

Szenario 4 ist dadurch gekennzeichnet, dass der Wert in Zeile 2 komplett vom Kunden eingetippt werden muss.



Aufbau der beiden zugehörigen Visualisierungstexte:

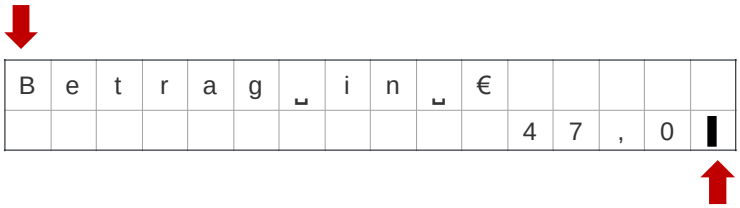
Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
4	**Def**	**Def**	**„Anzahl“**	**Def**	**Def**	**Def**

Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
4	**R**	**Def**	**Def**	**3**	**Def**	**Def**

In Zeile 2 wird die geforderte Länge der Eingabedaten angegeben. Alle anderen Werte entsprechen der Default-Belegung.

e) Szenario 5: Eingabe von Komma-Zahlen, 2-zeilig

Szenario 5 verwendet als Eingabe eine Kommadarstellung.



Aufbau der beiden zugehörigen Visualisierungstexte:

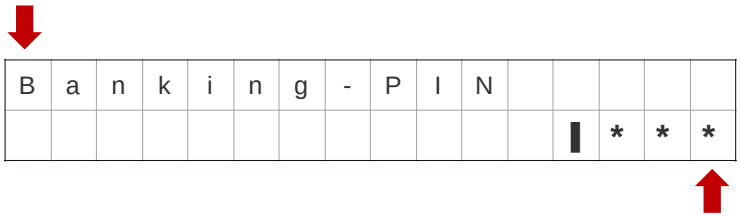
Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
5	**Def**	**Def**	**„Betrag in €“**	**Def**	**Def**	**Def**

Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
5	**R**	**Def**	**Def**	**5**	**05**	**Def**

In Zeile 2 wird zusätzlich zur geforderten Datenlänge 5 (inklusive Komma) das Formatkennzeichen „05“ für die Darstellung von 2 Kommastellen am Secoder verwendet.

f) Szenario 6: Verdeckte Eingabe, 2-zeilig

In Szenario 6 wird die Verwendung der verdeckten Eingabe angewendet. Die Eingabe der Banking-PIN ist mit Sicherheitsfunktion 811 auf diese Weise nicht unterstützt, da die Applikation „aut“ nicht über eine zentrale Anwendungs-PIN, sondern über die CSA-PIN geschützt ist, die lokal geprüft wird.



Aufbau der beiden zugehörigen Visualisierungstexte:

Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
6	**Def**	**Def**	**„Banking-PIN“**	**Def**	**Def**	**Def**

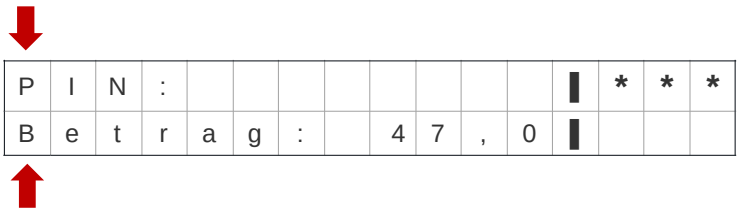
Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
6	**R**	**Def**	**Def**	**4**	**04**	**Def**

Die verdeckte Eingabe wird durch das Kennzeichen „04“ für die Darstellung am Secoder erreicht.

g) Szenario 7: Auffüllen mit Ziffern in einer Zeile

Szenario 7 stellt eine kompakte Darstellung der Eingabe von zwei Werten inklusive Beschreibung in je einem Dataset dar.

Hinweis: Diese Darstellung kann aus Sicht der Benutzerfreundlichkeit ungünstig wirken, wenn Secoder-Produkte mit unterschiedlichen Displaygrößen, z. B. 2 x 32, in identischer Weise angesteuert werden sollen.



Aufbau der beiden zugehörigen Visualisierungstexte:

Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
7	**Def**	**12**	**„PIN:“**	**4**	**04**	**Def**

Idx	Display-Pos	Länge (Secoder-Text)	Secoder-Text	Länge Sec-Eingabe-Daten	Ausrichtung und Fmt. Sec-Eingabedaten	Secoder-Padding
7	**Def**	**8**	**„Betrag:“**	**5**	**05**	**Def**

Die Verwendung von Secoder-Text und Eingabe in einer Zeile wird durch die Kombination eines statischen Secoder-Textes „Betrag:“, der durch die Längenangabe „8“ mit Leerzeichen ergänzt wird, mit einer 5-stelligen Eingabe erreicht.

Restriktion:
Als Einschränkung bei diesem Szenario gilt, dass keine Kombination eines statischen Secoder-Textes (Im Beispiel „Betrag:“) mit einem dynamischen Textfragment (z. B. aus der DTA-Defintion), das dann noch manuell zu ergänzen wäre, möglich ist. Hierfür muss eine zweizeilige Darstellung pro Parameter gewählt werden.

B.5.1.2 Struktur des Parametersegmentes HIVISS

Das Parametersegment HIVISS hat einen zweiteiligen Aufbau, der im Folgenden detailliert beschrieben ist:

- Abschnitt Nr. 1: Secoder Visualisierungstexte und Secoder MetaData
- Abschnitt Nr. 2: Geschäftsvorfallspezifische Visualisierungsinformationen

Die generelle Struktur von HIVISS wird aus folgendem Diagramm ersichtlich:

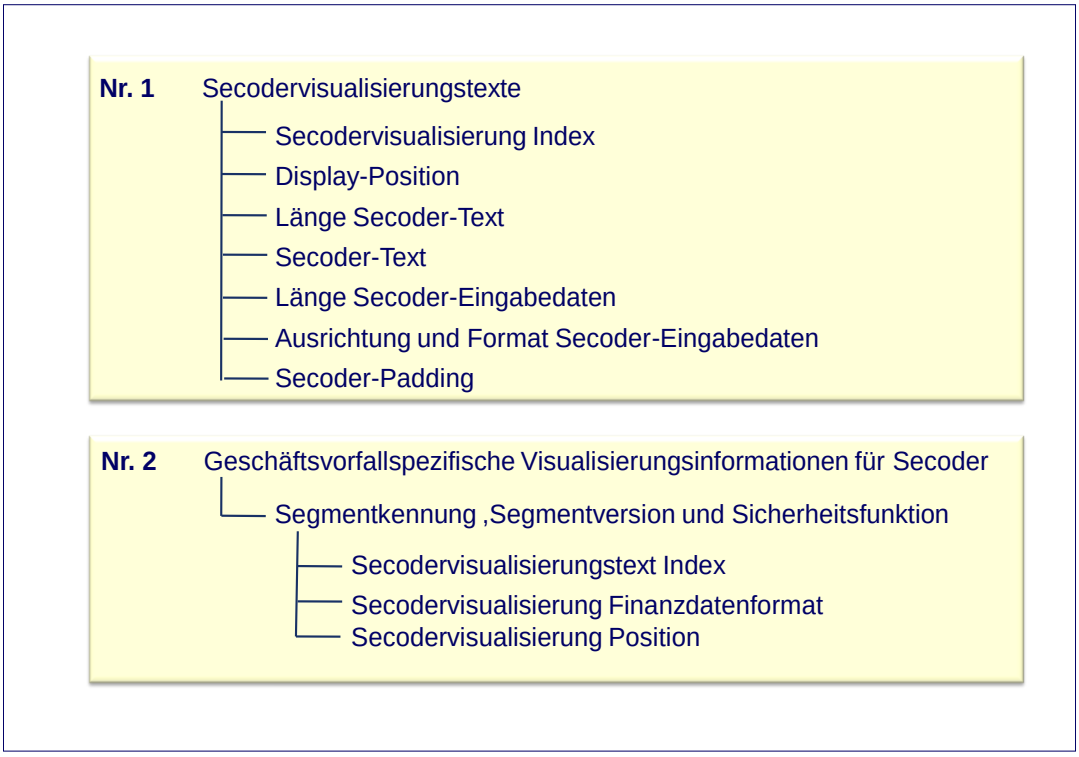


Abbildung 7: Aufbau des Parametersegmentes HIVISS

B.5.1.3Tabelle der Secodervisualisierungstexte

Im Display des Secoders können zunächst beliebige Texte angezeigt werden. Um jedoch Redundanzen zu vermeiden und die HIVISS möglichst kompakt zu halten, werden die verwendeten Texte und die Secoder MetaData in einer Tabelle jeweils bestehend aus Index, zugehörigem Text und MetaData zusammengefasst:

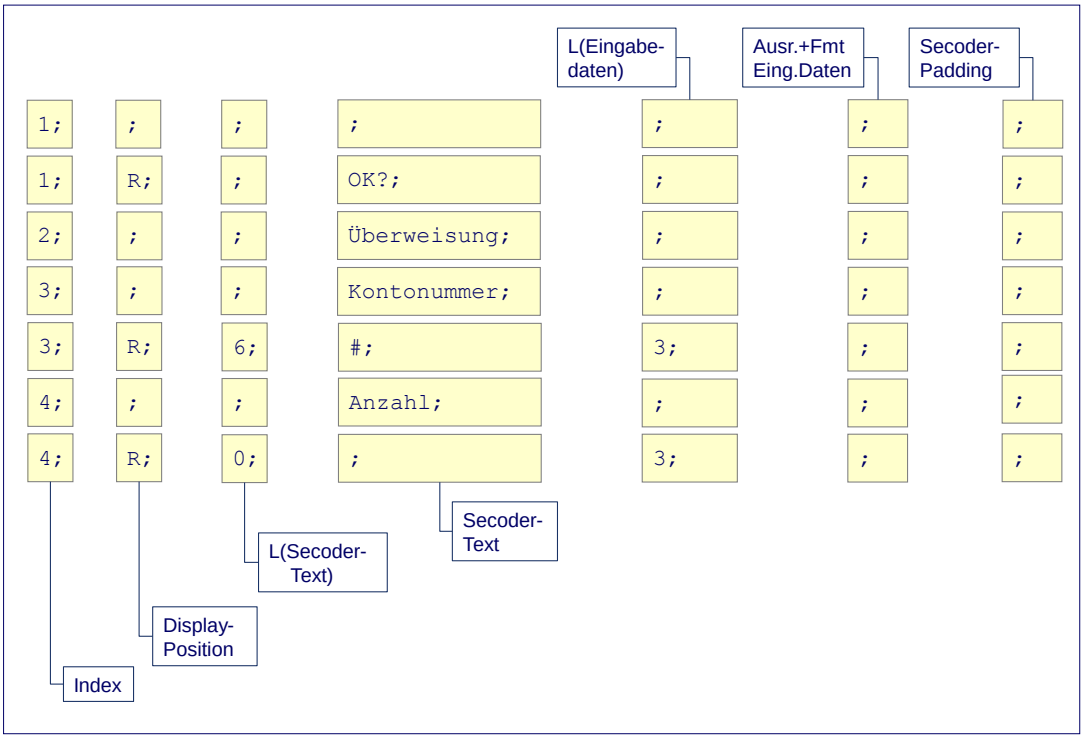


Abbildung 8: Tabelle der Secodervisualisierungstexte und Secoder MetaData in HIVISS

Die einzelnen Secodervisualisierungstexte können so über die Angabe des Index in den geschäftsvorfallspezifischen Visualisierungsdaten referenziert werden. Es können nur Texte dargestellt werden, die auch in der Tabelle der Secodervisualisierungstexte enthalten sind.

Die Secoder MetaData sind vom Funktionsumfang so gestaltet, dass sie 1:1 in SecCmds umgesetzt werden können, wie sie im Secoder User Interface & Implementation Guide [Secoder Impl] beschrieben sind.

Im Einzelnen sind folgende Werte möglich:

Secoder-visualisierung Index	Index des jeweiligen Secodervisualisierungstextes, der in den „geschäftsvorfallspezifischen Visualisierungsinformationen“ referenziert wird. Da im angenommenen Beispiel ein Dataset 1 x 16 Zeichen umfasst, werden 2 Anzeigedefinitionen benötigt, um z. B. 2 x 16 Zeichen für die Darstellung von Bezeichnung und Wert eines Auftragsbestandteils am Secoder darzustellen. Hierbei weisen beide Anzeigedefinitionen denselben Index auf. Die Reihenfolge der Anzeigedefinitionen in der Tabelle der Secodervisualisierungstexte entspricht

	der Reihenfolge der Anzeige dieser Texte am Secoder. Default: Keiner
Display-Position	Ausrichtung des Dataset im Display. Diese Angabe ist nötig, da die Secoder-Texte in der BPD nicht gepadded werden. Mögliche Werte sind: L: Links R: Rechts Die Secoder-Anwendungsfunktion kann über die Einstellung von DS_X.L-VIS und der realen Displaygröße die gewünschte Ausrichtung einstellen. Default: Links
Länge (Secoder-Text)	Folgende Möglichkeiten existieren: 1. Statischer Text , z. B. „Ziel-Konto:“ Hier ergibt sich die Länge implizit aus der Länge des Secoder-Textes und muss nicht angegeben werden (Default). 2. Dynamischer Wert, z. B. „12345678“ Die Daten werden dynamisch aus dem Auftrag (z. B. DTA) übernommen und werden in der angegebenen Länge verwendet. In diesem Fall muss ein konkreter Wert für die Länge angegeben werden. Default: 0
Secoder-Text	Für diesen Parameter existieren drei Ausprägungen: 1. Statischer Text die Bezeichnung („Label“) des zu bestätigenden bzw. zu ergänzenden Wertes, z. B. „Ziel-Konto:“. 2. Fester Anteil des dynamischen Werts Der zu bestätigende Wert selbst in Form des Platzhalters # bzw. dessen fester Anteil, z. B. „12345678“ bzw. „12345___“. 3. Leer Es ist weder ein statischer Text noch ein fester Anteil eines dynamischen Wertes angegeben, d. h. der Kunde muss den Wert komplett eingeben. Statische Secoder-Texte stehen fest und ohne Padding in der BPD. Dynamische Werte (komplett oder anteilig) werden durch den Platzhalter „#“ repräsentiert. Pro Anzeigedefinition, die aus bis zu zwei Datasets mit identischem Index bestehen kann, kann maximal ein Platzhalter „#“ definiert werden. Default: leer
Länge Secoder-Eingabedaten	Soll ein Wert nicht nur bestätigt, sondern komplett eingegeben oder ergänzt werden, wird hierdurch die Länge der geforderten Eingabedaten am Secoder vorgegeben. Hinweis: Das Ergänzen von Daten kann bei der Sicherheitsfunktion 811 nicht genutzt werden; es kann nur eine Bestäti-

<table>
<tr><td></td><td>gung von Daten erfolgen.
Default: 0</td></tr>
<tr><td>Ausrichtung und Format Secoder-Eingabedaten</td><td>Dieser Parameter beschreibt die Ausrichtung bei der Eingabe der Daten und die Verwendung des Kommas. Er entspricht dem Secoder-Parameter DS_x.DCF (Details hierzu siehe unter [Secoder_Impl]) und hat folgende Ausprägungen:
<table>
<tr><td>00</td><td>Eingabeposition ist ab der ersten Stelle hinter dem Secoder-Text bzw. rechts, falls kein Secoder-Text vorhanden.</td></tr>
<tr><td>01</td><td>Die Eingabe startet rechts (Taschenrechnermodus).</td></tr>
<tr><td>04</td><td>Wie 00, jedoch werden die Eingaben verdeckt dargestellt. Dieses Format ist bei der Sicherheitsfunktion 811 nicht zugelassen.</td></tr>
<tr><td>03
05
07</td><td>Wie 01, jedoch mit 1, 2 oder 3 Kommastellen.</td></tr>
</table>
Default:
00, falls Länge (Secoder-Text) > 0
01, sonst</td></tr>
<tr><td>Secoder-Padding</td><td>Dieser Parameter beschreibt das Padding der Daten im Vis-Data-Puffer des Secoders und entspricht dem Secoder-Parameter DS_x.VCI (Details hierzu siehe unter [Secoder_Impl]) und hat folgende Ausprägungen:
<table>
<tr><td>D0</td><td>Numerische Daten (BCD) mit 0-Padding</td></tr>
<tr><td>E0</td><td>Numerische Daten (BCD) mit F-Padding</td></tr>
<tr><td>DF</td><td>Alfanumerische Daten (ISO 646) mit 0-Padding</td></tr>
<tr><td>EF</td><td>Alfanumerische Daten (ISO 646) mit F-Padding</td></tr>
</table>
Default:
E0 bei numerischen Daten
EF bei alfanumerischen Daten</td></tr>
</table>

Es bestehen die folgenden Zusammenhänge zwischen den MetaData-Defintionen und dem Aufbau des Secoder Data Confirmation Kommandos.

BPD-Parameter	Secoder-Kommando
Index	keine Entsprechung
Display-Position	keine Entsprechung, wird anhand $DS_x.L_{VIS}$ und physischer Displaygröße umgesetzt
Länge (Secoder-Text)	$DS_x.L_{DB}$
Secoder-Text	$Dataset_x.Datablock_x (DS_x.DB_x)$
Länge Secoder-Eingabedaten	$DS_x.L_{DB}$ + L (Secoder-Eingabedaten) = $DS_x.L_{VIS}$
Ausrichtung und Format Secoder-Eingabedaten	DS_x.DCF
Ziffern/Buchstaben	DCF = ´00´ / ´01´
Kommastellen K1 bis K3	DCF = ´03´/ ´05´/ ´07´
Asterisk	DCF = ´04´
Secoder-Padding	DS_x.VCI

Abbildung 9: Analogien zwischen MetaData und Secoder Data Confirmation

B.5.1.4 Geschäftsvorfallspezifische Visualisierungsinformationen für Secoder

Die Mechanismen zur Steuerung des Visualisierungsvorgangs im Rahmen des Parametersegmentes HIVISS werden durch die folgende Abbildung verdeutlicht. Dabei sind die Werte für Segmentversion und Sicherheitsfunktion mit „0“ als Default definiert, um die Anzahl der Einträge zu minimieren. Werden hierbei explizite Werte benutzt, um unterschiedliche Visualisierungen zu erreichen (z. B. Segmentversion #1 und #2 und/oder S-Fkt= 811), so muss pro Ausprägung eine eigene Definition vorhanden sein.

Hinweise:

1. Alle nicht explizit modellierten Segmentversionen bzw. Sicherheitsfunktionen werden analog dem Default-Eintrag behandelt. Beim aktuellen Stand der Spezifikation mit nur einem AZS-Verfahren ist dies gleichbedeutend mit dem Default-Eintrag.
2. Für jede in HIAZSS angegebene Sicherheitsfunktion muss eine Definition (Default oder Explizit) in HIVISS vorhanden sein.

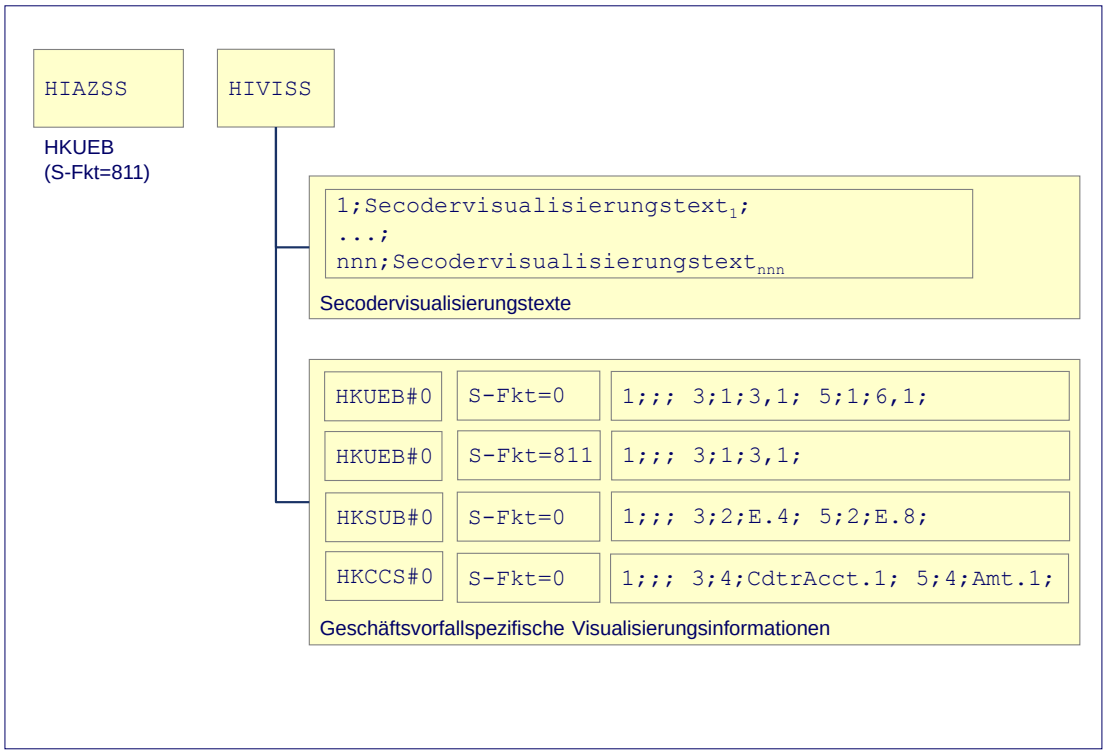


Abbildung 10: Definition der Secoder MetaData pro Geschäftsvorfall

Das Parametersegment enthält im Abschnitt der Geschäftsvorfallspezifischen Visualisierungsinformationen pro Anzeigendefinition einen Eintrag mit folgendem Aufbau:

Index	Index auf den / die darzustellenden Visualisierungstext(e) in der Tabelle der Secodervisualisierungstexte. Die dort enthaltenen Texte (1 oder 2 Datasets) mit identischem Index werden in der dort definierten Reihenfolge abgearbeitet.
Finanzdatenformat	Kennzeichnung der Art des Finanzdatenformates wie z. B. FinTS, DTA oder SEPA; dies ist Voraussetzung für die Positionierung innerhalb eines Finanzdatenformates, um die zu visualisierenden Daten zu lokalisieren. Außer FinTS sind dort alle vorkommenden Finanzdatenformate definiert.
Position	Je nach Finanzdatenformat wird die Position des zu visualisierenden Elementes angegeben, z. B. FinTS: 3,1 → 3. DEG, 1. Datenelement DTA: E.4 → 4. Element im E-Satz SEPA: CdtrAcct.1 → 1. Vorkommen des Tag „CdtrAcct“ Je Anzeigedefinition kann maximal ein Positionsparameter festgelegt werden. Dieser ersetzt dann den Platzhalter „#“ an der entsprechenden Stelle des jeweiligen Secodervisualisierungstextes.

Die einzelnen Einträge sind durch Semikolon getrennt, da sie syntaktisch in FinTS V3.0 nicht mehr darstellbar sind. Ein DE „Secodervisualisierung MetaData“ kann theoretisch bis zu 255 solcher Secoder-Visualisierungselemente aufnehmen; die Begrenzung erfolgt nur durch den Speicherplatz im Secoder.

Da alle drei Informationen zwingend belegt werden müssen, können die einzelnen Secoder-Visualisierungselemente leicht identifiziert werden.

B.5.1.5Positionierung bei der Secodervisualisierung

Abhängig vom jeweiligen Finanzdatenformat existieren unterschiedliche Adressierungsmöglichkeiten für die einzelnen Elemente im Format. Es handelt sich dabei im Regelfall um die Auswertung der Kundennachrichten, welche ein oder mehrere Finanzdatenformate enthalten können, d. h. ein Kundenprodukt kann – ‚offline‘ – vor der Einreichung des Auftrags auf Basis der BPD die Secodervisualisierungsdaten ermitteln. Gleiches gilt für Werte aus dem Segment HKIDN, welche im Rahmen der Dialoginitialisierung visualisiert werden können.

Enthält die Kundennachricht mehrere Aufträge – in Form mehrerer FinTS-Elemente oder Sammelaufträge – so muss beim Aufbau der BPD darauf geachtet werden, dass ein Visualisierungselement ausgewählt wird, das sich auf alle Aufträge bezieht wie z. B. Summenwerte oder ein bestimmtes Auftreten eines Wertes im Auftrag. Iterationen von Visualisierungselementen über enthaltene Einzelaufträge – also die Anzeige mehrerer Instanzen eines Wertes im Secoder – sind nicht vorgesehen.

Da durch die Definition im DE „Secodervisualisierung Finanzdatenformat“ eine Festlegung nur für ein Secoder-Visualisierungselement gemacht wird, können die Visualisierungsdaten für einen Auftrag auch aus unterschiedlichen Finanzdatenformaten bestehen, z. B. aus ggf. vorangestellten FinTS-Datenelementen und einem DTA-Format.

B.5.1.5.1 Positionierung bei FinTS-Formaten

Bei FinTS-Formaten findet die Positionierung folgendermaßen statt:

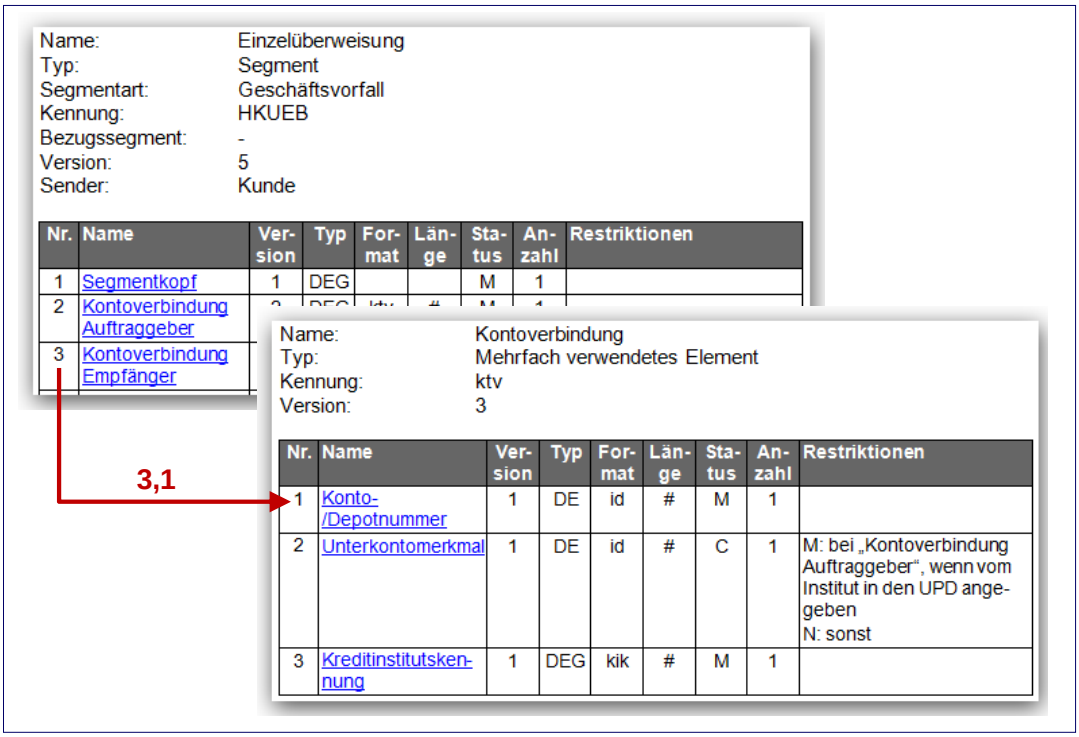


Abbildung 11: Adressierung der Secodervisualisierungsdaten bei FinTS-Formaten

Entsprechend der definierten Protokollhierarchie wird ein Datenelement durch seine Position im Segment beschrieben. Ausschlaggebend ist dabei die Nummerierung, die in der Geschäftsvorfallbeschreibung angegeben ist.

Ein Datenelement auf Segmentebene wird durch die einstufige Adressierung

`<Position des DE im Segment>`

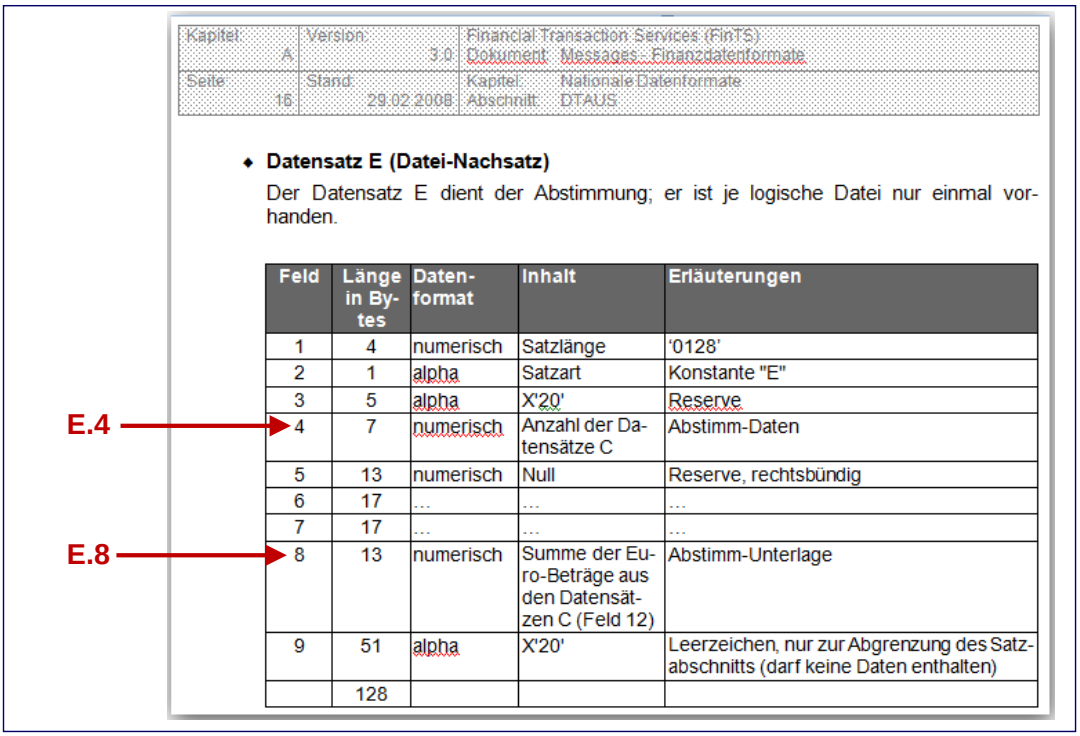
beschrieben (z. B. ‚4‘ für ‚Name Empfänger‘ beim Segment Einzelüberweisung).

Ein Gruppendatenelement in einer Datenelementgruppe wird durch die zweistufige Adressierung

`<Position der DEG im Segment>,<Position des GD in der DEG>`

beschrieben (z. B. ‚3,1‘ für ‚Konto-/Depotnummer Empfänger‘ in der DEG ‚Kontoverbindung Empfänger‘ beim Segment ‚Einzelüberweisung‘).

B.5.1.5.2 Positionierung bei DTA



◆ **Datensatz E (Datei-Nachsatz)**

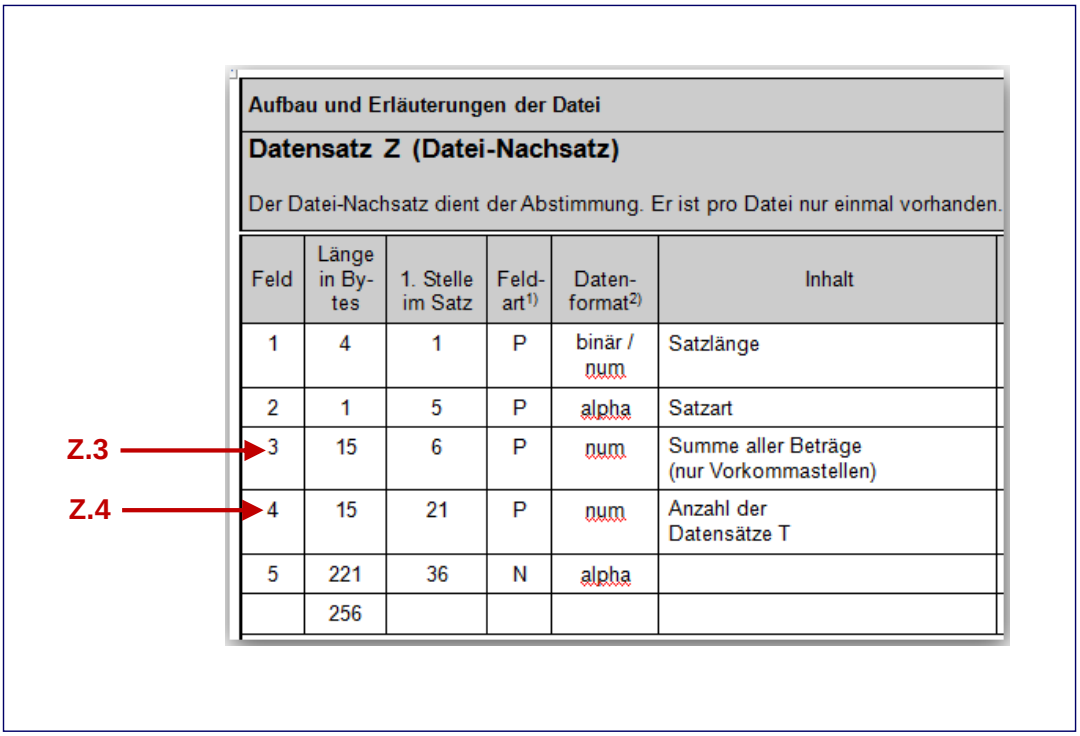
Der Datensatz E dient der Abstimmung; er ist je logische Datei nur einmal vorhanden.

Feld	Länge in Bytes	Datenformat	Inhalt	Erläuterungen
1	4	numerisch	Satzlänge	'0128'
2	1	alpha	Satzart	Konstante "E"
3	5	alpha	X'20'	Reserve
4	7	numerisch	Anzahl der Datensätze C	Abstimm-Daten
5	13	numerisch	Null	Reserve, rechtsbündig
6	17	...	...	...
7	17	...	...	...
8	13	numerisch	Summe der Euro-Beträge aus den Datensätzen C (Feld 12)	Abstimm-Unterlage
9	51	alpha	X'20'	Leerzeichen, nur zur Abgrenzung des Satzabschnitts (darf keine Daten enthalten)
	128			

Abbildung 12: Adressierung der Secodervisualisierungsdaten bei DTA-Formaten

Beim Format „DTA“ wird durch die erste Stelle der Datensatz innerhalb des DTA-Formates festgelegt; die zweite Stelle bezeichnet das Feld innerhalb des DTA-Datensatzes. Es erfolgt also grundsätzlich eine zweistufige Adressierung.

B.5.1.5.3 Positionierung bei DTAZV



Aufbau und Erläuterungen der Datei

Datensatz Z (Datei-Nachsatz)

Der Datei-Nachsatz dient der Abstimmung. Er ist pro Datei nur einmal vorhanden.

Feld	Länge in Bytes	1. Stelle im Satz	Feldart[1)]	Datenformat[2)]	Inhalt
1	4	1	P	binär / num	Satzlänge
2	1	5	P	alpha	Satzart
3	15	6	P	num	Summe aller Beträge (nur Vorkommastellen)
4	15	21	P	num	Anzahl der Datensätze T
5	221	36	N	alpha	
	256				

Abbildung 13: Adressierung der Secodervisualisierungsdaten bei DTAZV-Formaten

Beim Format „DTAZV“ wird durch die erste Stelle der Datensatz innerhalb des DTAZV-Formates festgelegt; die zweite Stelle bezeichnet das Feld innerhalb des DTAZV-Datensatzes. Es erfolgt also grundsätzlich eine zweistufige Adressierung.

B.5.1.5.4 Positionierung bei SEPA XML Formaten

Bei den SEPA-Formaten wird zur Bezeichnung der im Secoder-Display zu visualisierenden Daten das entsprechende Tag verwendet, gefolgt von einem Index i, der das i-te Vorkommen dieses Tags im SEPA-Format bezeichnet, wie die folgende Abbildung zeigt:

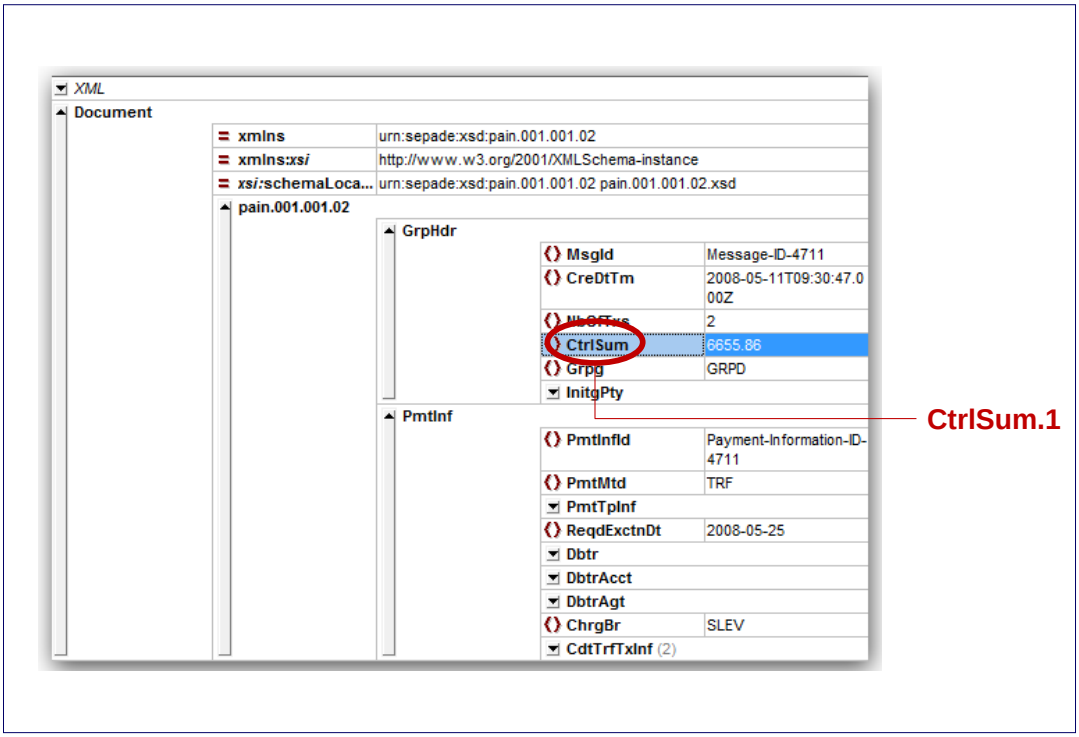


Abbildung 14: Adressierung der Secodervisualisierungsdaten bei SEPA-Formaten

In diesem Fall würde das erste vorkommende Tag `CtrlSum` als Position angegeben. Die Angabe erfolgt case sensitive.

Hinweis: für die Visualisierung der SEPA-Formate können auch die Informationen aus dem Datenelementen des FinTS-Geschäftsvorfalls (außerhalb des transparenten SEPA-Formats) wie z. B. das Datenelement `Wert` aus der DEG `Summenfeld` über die Kennzeichnung `3,1` visualisiert werden.

B.5.1.6 Parametersegment HIVISS

Realisierung Bank: verpflichtend, falls Geschäftsvorfälle mit AZS-Absicherung angeboten werden

Realisierung Kunde: optional

◆ **Format**

Name: Secoder-spezifische Visualisierungsinformationen

Typ: Segment

Segmentart: Geschäftsvorfall

Kennung: HIVISS

Bezugssegment: HKVVB

Segmentversion: 1

Sender: Kreditinstitut

Format: Geschäftsvorfall mit Parametern

◆ **Erläuterungen**

Name: Parameter Secoder-spezifische Visualisierungsinformationen

Typ: Datenelementgruppe

Status: M

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Segmentkopf	DEG			M	1	
2	Maximale Anzahl Aufträge	DE	num	..3	M	1	
3	Anzahl Signaturen mindestens	DE	num	1	M	1	0, 1, 2, 3
4	Sicherheitsklasse	DE	code	1	M	1	0, 1, 2, 3, 4
5	Parameter Secoder-spezifische Visualisierungsinformationen	DEG			M	1	

◆ **Belegungsrichtlinien**

In „Parameter Secoder-spezifische Visualisierungsinformationen“, dort „Geschäftsvorfallspezifische Visualisierungsinformationen für Secoder“

Segmentversion

Als „Segmentversion“ kann in HIVISS eine der definierten Segmentversionen des jeweiligen Segmentes oder der Wert „0“ auftreten. „0“ bedeutet, dass die Struktur „Geschäftsvorfallspezifische Visualisierungsinformationen für Secoder“ als Default für alle Segmentversionen des Segmentes gilt, für die keine explizite Parameter-Struktur vorhanden ist.

Sicherheitsfunktion, kodiert

Als „Sicherheitsfunktion, kodiert“ kann in HIVISS einer der für AZS-Verfahren definierten Werte (derzeit nur 811) oder der Wert „0“ auftreten. „0“ bedeutet, dass die Struktur „Geschäftsvorfallspezifische Visualisierungsinformationen für Secoder“ als Default für all die AZS-Verfahren gilt, für die keine explizite Definition vorhanden ist.

B.5.2 Spezielle Festlegungen für die Dialoginitialisierung beim AZS-Verfahren

Im Rahmen der Dialoginitialisierung werden folgende Informationen ausgetauscht:

Zugelassene AZS-Verfahren für den Benutzer

In der Dialoginitialisierungsantwort wird dem Kunden im Rahmen der Rückmeldungen zu Segmenten (HIRMS) über den Rückmeldungscode 3921 und entsprechende Rückmeldungsparameter mitgeteilt, welche konkreten AZS-Verfahren für ihn zugelassen sind. Dabei wird pro Rückmeldeparameter (P1 bis P10) ein Verfahrenskennzeichen übermittelt. Die Kodierung erfolgt analog der Belegung des DE „Sicherheitsfunktion, kodiert“ im Parametersegment HIVISS, also ist derzeit nur der Wert „811“ möglich.

Das Kreditinstitut muss organisatorisch sicherstellen, dass der Kunde über eine geeignete Version eines Kundenproduktes verfügt, das die Rückmeldeparameter entsprechend interpretieren kann. In jedem Falle sollte der Kunde durch einen verständlichen Rückmeldetext darauf hingewiesen werden, dass er ggf. ein aktualisiertes Kundenprodukt benötigt.

Sollte der Kunde vertraglich an die Nutzung eines der AZS-Verfahren gebunden sein und verwendet er ein Kundenprodukt, welches dieses AZS-Verfahren nicht unterstützt, so ist der Dialog zu beenden. Über den Rückmeldungscode 9955 „AZS-Verfahren nicht zugelassen“ und einen geeigneten Rückmeldungstext muss der Kunde eindeutig über die Ursache dieser Dialogbeendigung informiert werden. Der Rückmeldungstext muss auch berücksichtigen, dass die Anfrage des Kundenproduktes z. B. mit DE „Sicherheitsfunktion, kodiert“ = „999“ in diesem Fall nur erfolgt, um die unterstützten Sicherheits-Verfahren für den Benutzer zu ermitteln. Diese müssen über den Rückmeldungscode 3921 „Zugelassene AZS-Verfahren für den Benutzer“ (oder den entsprechenden Rückmeldungscode 3921 in Kombination mit Code 9800 im Fehlerfall) mitgeteilt werden. Bei der Sicherheitsfunktion 811 gilt dies ab dem ersten Dialog nach der Schlüsseleinreichung (mit S-Fkt=1).

Sollte das Kundenprodukt AZS-Verfahren unterstützen und noch keine Verfahrensparameter mit Angabe der für den aktuellen Benutzer unterstützten Verfahren verfügen, so muss es einen Dialog eröffnen, um über die Rückmeldeparameter in Kenntnis der erlaubten Verfahren zu gelangen. Hierbei ist für das DE „Sicherheitsfunktion, kodiert“ der Wert „999“ für Ein-Schritt-TAN-Verfahren zu verwenden.

Gewähltes AZS-Verfahren des Kunden

Ein Kunde kann aus den für Ihn zugelassenen AZS-Verfahren eines für den aktiven Dialog auswählen. Das entsprechende Verfahrenskennzeichen wird in das DE „Si-

cherheitsfunktion, kodiert“ im Signaturkopf der Dialoginitialisierungsnachricht eingestellt. Die Kodierung erfolgt analog der Belegung des DE „Sicherheitsfunktion, kodiert“ im Parametersegment HIVISS, also ist derzeit nur der Wert „811“ möglich. Das gewählte AZS-Verfahren muss für den Benutzer erlaubt sein (BPD, Rückmeldung 3921 bei Dialoginitialisierung).

C. DIALOGSPEZIFIKATION FÜR AZS-VERFAHREN

C.1 Allgemeines

Der Einsatz von AZS-Verfahren erfordert den Transport von sicherheitsrelevanten Daten mit Hilfe des Geschäftsvorfalls „Alternative ZKA-Sicherheitsverfahren (HKAZS)“. Hiermit lassen sich alle auftragsbezogenen Sicherheitsinformationen komplett transportieren und die HBCI-Signatursegmente werden nur soweit nötig mit konkreten Werten gefüllt und ansonsten mit FinTS-Füllwerten bestückt übertragen. Insbesondere wird bei AZS-Verfahren das DE „Validierungsresultat“ im Signaturabschluss nicht belegt, da die Signatur selbst im DE „Signaturdaten“ in HKAZS bzw. HIAZS übertragen wird. AZS-Verfahren (derzeit nur S-Fkt=811) können mit FinTS V3.0 eingesetzt werden[1]. In HBCI V2.2 ist das AZS-Verfahren gemäß Sicherheitsfunktion 811 nicht unterstützt.

Zusätzlich besteht jedoch die Option, bereits im Zuge der Dialoginitialisierung visualisierte Daten übertragen zu können, was dazu führt, dass optional bereits zu diesem Zeitpunkt HKAZS-Segmente zu integrieren sind. Da eine Modifikation des FinTS-Protokolls auf ein Minimum beschränkt werden soll, werden die im Folgenden beschriebenen Anpassungen gezielt nur für die AZS-Verfahren eingesetzt. Diese sind am Nummernkreis 800 bis 899 für die „Sicherheitsfunktion, kodiert“ eindeutig zu erkennen. Für alle anderen Sicherheitsfunktionen bleibt die bestehende FinTS V3.0 Protokollstruktur erhalten.

Beim AZS-Verfahren gemäß Sicherheitsfunktion 811 kann das Segment HIAZS in der Kreditinstitutsnachricht wegfallen, wenn es ausschließlich FinTS-Füllwerte enthält.

C.1.1 Verschlüsselung des Dialoges

Grundsätzlich sind bei AZS-Verfahren sowohl alle Kunden- als auch alle Kreditinstitutsnachrichten eines Dialoges mit HBCI zu verschlüsseln. Von dieser Regel ausgenommen sind die folgenden Dialogarten:

- Anonymer Zugang
- Schlüsselsperrung durch den Kunden
- Kommunikationszugang anfordern

Im Unterschied zu der standardmäßigen Belegung in FinTS wird bei AZS-Verfahren im Datenelement „Sicherheitsfunktion, kodiert“ der Wert für das AZS-Verfahren, also 811 eingestellt. Hierdurch wird indirekt auch das eigentliche Verschlüsselungsverfahren „HBCI“ festgelegt.

Die sonstigen Protokolleigenschaften zum Verschlüsseln von Daten sind den entsprechenden Vorgaben des Sicherheitsverfahrens HBCI (HBCI-Verschlüsselung) zu entnehmen.

---

[1] Ab FinTS V4.1 ist die Secoderunterstützung direkt in das FinTS-Protokoll integriert und die Geschäftsvorfälle HKAZS und HIVISS sind somit obsolet.

C.1.2 Institutssignaturen bei AZS-Verfahren

Beim AZS-Verfahren 811 werden keine Institutssignaturen eingesetzt.

C.1.3 Key-Management bei AZS-Verfahren

Der Geschäftsvorfall HKAZS wird nur zur Auftragsverarbeitung verwendet, d. h. dass beim AZS-Verfahren 811 davon ausgegangen wird, dass das Key- bzw. Zertifikatsmanagement mit anderen syntaktischen Mitteln erfolgt.

Daher muss die erstmalige Schlüsseleinreichung nach dem Standard RDH- bzw. RAH-Verfahren erfolgen. Zum Einreichen der Schlüssel muss also zunächst ein separater Dialog mit Sicherheitsfunktion=1 durchgeführt werden. Gleiches gilt für die Schlüsselsperre. Änderungen der Kundenschlüssel könnten bei den kartenbasierten AZS-Verfahren nicht auftreten.

Eine Ausnahme stellt das Anfordern der öffentlichen Schlüssel des Instituts dar. Der Austausch erfolgt via HKISA / HIISA im Rahmen des AZS-Verfahrens.

C.1.4 Behandlung der Dialogendenachricht (HKEND)

Die Dialogendenachricht (HKEND) wird grundsätzlich nicht signiert. Dies gilt für die Kunden- und die Institutsseite. Dadurch kann das entsprechende HxAZS, sowie die Segmente Signaturkopf (HNSHK) und Signaturabschluss (HNSHA) entfallen. Die Verschlüsselung ist von dieser Sonderregelung nicht betroffen.

C.2 Besondere Belegungsrichtlinien für AZS-Verfahren

In den im Folgenden dargestellten Strukturen sollten Datenelemente mit Status „O“, grundsätzlich leer gelassen werden.

Für einige Datenelemente gelten bei AZS-Verfahren besondere Belegungsrichtlinien, die von den allgemeinen in [HBCI] aufgeführten Richtlinien abweichen. Diese sind nachfolgend aufgeführt.

C.2.1 Segment Sicherheitsverfahren, DEG Unterstützte Sicherheitsverfahren

Das Segment „Sicherheitsverfahren (HISHV)“ ist Teil der Segmentfolge „Bankparameterdaten“.

Um die Kompatibilität mit den bestehenden Sicherheitsverfahren HBCI und PIN/TAN sicherzustellen, konnte der mögliche Wertebereich innerhalb von HISHV-Segmenten nicht um einen weiteren Wert für AZS-Verfahren erweitert werden. Clients können diesem Segment somit nicht entnehmen, ob AZS-Verfahren unterstützt werden oder nicht. Dies muss am Vorkommen des HIVISS-Segments festgemacht werden. Ist ein solches Segment vorhanden, werden AZS-Verfahren unterstützt, andernfalls nicht.

C.2.2 DEG „Sicherheitsprofil“

Die DEG „Sicherheitsprofil“ wird sowohl im Signaturkopf als auch in HKAZS verwendet und identisch belegt.

Sicherheitsverfahren, Code

„RDH“ bzw. „RAH“ bei allen Nachrichten

Version des Sicherheitsverfahrens

„r“ : bei allen Nachrichten mit Sicherheitsfunktion 811,
wobei „r“ dem der Version des Sicherheitsprofils entspricht,
dessen Algorithmen bei HBCI-Verschlüsselung, Hashwert-Bildung usw. verwendet werden.
Gültige Sicherheitsprofile: RAH-7, RAH-9, RDH-3, RDH-5 – RDH-9

C.2.3 DEG „Schlüsselname“

Es gelten die HBCI-Konventionen.

Schlüsselnummer

FinTS-Füllwert, z. B. „0“

Schlüsselversion

FinTS-Füllwert, z. B. „0“

C.2.4 DEG „Sicherheitsidentifikation, Details“

CID

Dieses Feld ist bei allen AZS-Verfahren mit der CID zu belegen, da es sich bei AZS-Verfahren ausschließlich um Chipkarten-Verfahren handelt.

Identifizierung der Partei

Dieses Feld muss eine gültige, zuvor vom Banksystem angeforderte Kundensystem-ID enthalten (analog zum RSA-Verfahren). Dies gilt auch für Zweit- und Drittsignaturen.

C.2.5 Segment „Signaturkopf“

Zertifikat

Dieses Feld wird bei den zertifikatsbasierten Verfahren RAH-7 und RDH-7 belegt.

C.2.6 DEG „Hashalgorithmus“

Es gelten die HBCI-Konventionen.

Wert des Hashalgorithmusparameters

Dieses Feld darf nicht belegt werden.

C.2.7 DEG „Signaturalgorithmus“

Es gelten die HBCI-Konventionen.

Signaturalgorithmus, kodiert

FinTS-Füllwert, z. B. „10“

Operationsmodus, kodiert

FinTS-Füllwert, z. B. „16“

C.2.8 Segment „Signaturabschluss“

Es ist der Signaturabschluss gemäß [HBCI] in Segmentversion 2 zu verwenden.

Validierungsresultat

Dieses Feld darf nicht belegt werden, da die Signatur selbst im DE „Signaturdaten“ in HKAZS bzw. HIAZS übertragen wird.

C.2.9 Segment „Verschlüsselungskopf“

Sicherheitsfunktion, kodiert

Im Unterschied zu der standardmäßigen Belegung in FinTS wird bei AZS-Verfahren im Datenelement „Sicherheitsfunktion, kodiert“ der Wert für das AZS-Verfahren, also 811 eingestellt.

Zertifikat

Dieses Feld darf nicht belegt werden.

C.2.10 DEG „Verschlüsselungsalgorithmus“

Es gelten die HBCI-Konventionen.

Wert des Algorithmusparameters, Schlüssel

FinTS-Füllwert, z. B. X’00 00 00 00 00 00 00 00’

Bezeichner für Algorithmusparameter, Schlüssel

FinTS-Füllwert, z. B. „5“

Wert des Algorithmusparameters, IV

Belegung nicht zulässig.

C.2.11 Segment „Verschlüsselte Daten“

Daten, verschlüsselt

Enthält die verschlüsselten Daten.

C.3 Nachrichtenaufbau für AZS-Verfahren

In Ergänzung zum bisherigen Segmentaufbau wird in der FinTS-Kundennachricht vor dem Signaturabschlusssegment ein Segment HKAZS eingefügt, das die alternativen Sicherungselemente transportiert. Somit kann das HKAZS-Segment auch als Erweiterung des Signaturabschlusssegmentes gesehen werden.

C.3.1 Kundennachricht bei der Dialoginitialisierung

Die folgende Abbildung zeigt den Aufbau der Kundennachricht bei der Dialoginitialisierung mit Visualisierungsdaten (Signatur-Prozess=6). Sie enthält zusätzlich ein HKAZS-Segment zur Übermittlung der Signatur-und Visualisierungsinformationen. In die Signaturbildung fließen die Segmente Signaturkopf (HNSHK), Identifikation (HKIDN), Verarbeitungsvorbereitung (HKVVB) und ggf. die Anforderung öffentlicher Schlüssel (HKISA) ein.

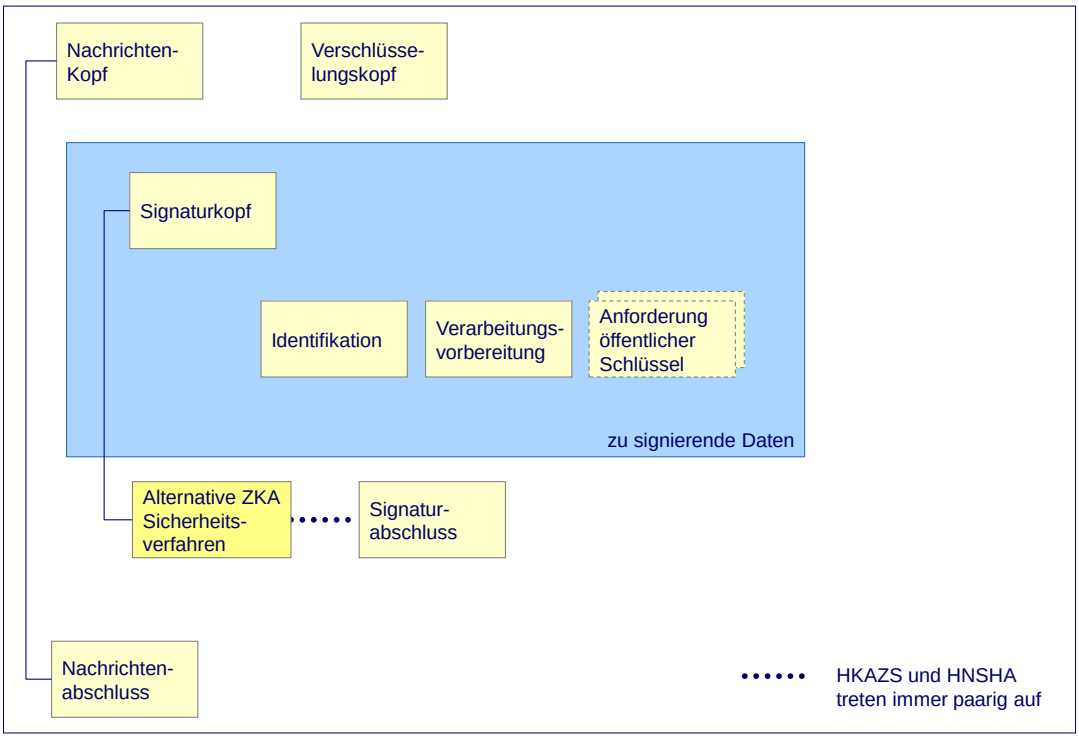


Abbildung 15: Segmentaufbau der Dialoginitialisierungsnachricht (Kunde) bei AZS-Verfahren

Die folgende Tabelle zeigt die Belegungsmöglichkeiten für das AZS-Verfahren 811.

Segmentname	S-Fkt=811
HNHBK	X
HNVSK	811
HNSHK	X
HKIDN	X
HKVVB	X
HKISA	X
HKAZS	X
HNSHA	X
HNHBS	X

X: Segment wird gemäß FinTS-Protokoll ohne Sonderbelegung verwendet.

C.3.1.1 Nachrichtenformat

Realisierung Bank: verpflichtend
Realisierung Kunde: verpflichtend

- **Beschreibung**

Da der Kunde die Dialogsprache erst in dieser Nachricht mitteilt, muss zur Bildung der Dialoginitialisierungsnachricht der mit der Standardsprache des Kreditinstituts festgelegte Zeichensatz herangezogen werden. Dieser ist dem Feld „Standardsprache“ des Kommunikationszugangs zu entnehmen. Die Antwort des Kreditinstituts erfolgt dann in der vom Kunden gewünschten Sprache (Zeichensatz).

- **Format**

Name: Dialoginitialisierung

Typ: Nachricht

Version: 5

Sender: Kunde

Nr.	Name	Typ	Kennung	Status	Anzahl	Anmerkungen
1	Nachrichtenkopf	SEG	HNHBK	M	1	
2	Signaturkopf	SEG	HNSHK	M	1	s. [HBCI], Kap. B.5.1
3	Identifikation	SEG	HKIDN	M	1	
4	Verarbeitungsvorbereitung	SEG	HKVVB	M	1	
5	Anforderung eines öffentlichen Schlüssels	SEG	HKISA	C	3	O: bei RDH/RAH, S-Fkt=811 N: sonst
6	Alternative ZKA Sicherungsverfahren	SEG	HKAZS	C	1	M: S-Fkt=800 .. 899 N: sonst
7	Signaturabschluss	SEG	HNSHA	M	1	s. [HBCI], Kap. B.5.2
8	Nachrichtenabschluss	SEG	HNHBS	M	1	

C.3.2 Kreditinstitutsnachricht bei der Dialoginitialisierung

Die folgende Abbildung zeigt den generellen Aufbau der Kreditinstitutsnachricht bei der Dialoginitialisierung. In die Signaturbildung fließen die Segmente Signaturkopf

(HNSHK), Rückmeldungen (HIRMG und HIRMS), und ggf. BPD und UPD, die Übermittlung öffentlicher Schlüssel (HIISA) und Kreditinstitutsmeldungen (HIKIM) ein.

Beim AZS-Verfahren 811 entfällt das Segment HIAZS.

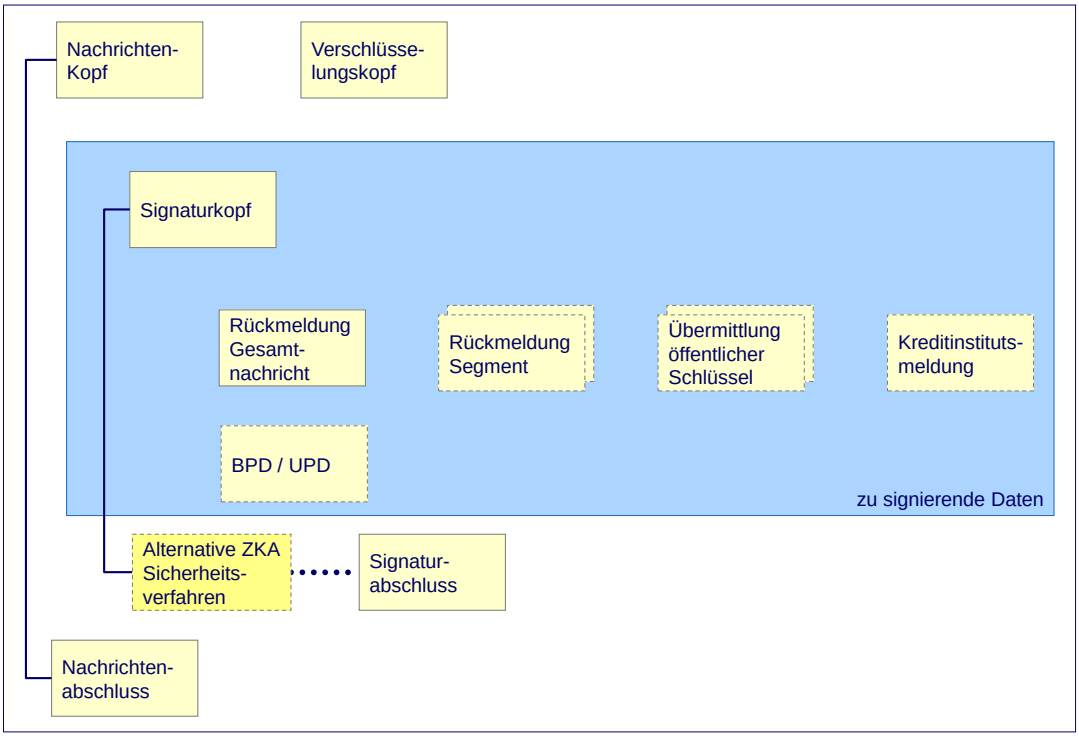


Abbildung 16: Segmentaufbau der Dialoginitialisierungsnachricht (Kreditinstitut) bei AZS-Verfahren

Die folgende Tabelle zeigt die Belegungsmöglichkeiten für das AZS-Verfahren 811.

Segment-name	S-Fkt=811
HNHBK	X
HNSHK	X
HIRMG	X
HIRMS	X
„BPD“	X
„UPD“	X
HIISA	X
HIKIM	X
HIAZS	nicht erlaubt
HNSHA	FinTS-FW
HNHBS	X

X: Segment wird gemäß FinTS-Protokoll ohne Sonderbelegung verwendet.

C.3.2.1 Nachrichtenformat

Realisierung Bank: verpflichtend
Realisierung Kunde: verpflichtend

- **Format**

Name: Antwort auf Dialoginitialisierung

Typ: Nachricht

Version: 5

Sender: Kreditinstitut

Nr.	Name	Typ	Kennung	Status	Anzahl	Anmerkungen
1	Nachrichtenkopf	SEG	HNHBK	M	1	
2	Signaturkopf	SEG	HNSHK	O	1	s. [HBCI], Kap. B.5.1
3	Rückmeldungen zur Gesamtnachricht	SEG	HIRMG	M	1	
4	Rückmeldungen zu Segmenten	SEG	HIRMS	O	n	
5	Bankparameterdaten	SF	#	O	1	
6	Userparameterdaten	SF	#	O	1	
7	Übermittlung eines öffentlichen Schlüssels	SEG	HIISA	C	3	O: bei RDH/RAH, S-Fkt=811 N: sonst
8	Kreditinstitutsmeldung	SEG	HIKIM	O	n	
9	Alternative ZKA Sicherungsverfahren	SEG	HIAZS	C	1	M: S-Fkt=800 .. 899 N: sonst
10	Signaturabschluss	SEG	HNSHA	O	1	s. [HBCI], Kap. B.5.2
11	Nachrichtenabschluss	SEG	HNHBS	M	1	

C.3.3 Kundenauftragsnachricht

Die folgende Abbildung zeigt den Aufbau der Kundennachricht bei der Auftragsverarbeitung. Sie enthält zusätzlich ein HKAZS-Segment zur Übermittlung der Signatur- und Visualisierungsinformationen. In die Signaturbildung fließt der Signaturkopf (HNSHK) und ein Auftragssegment ein.

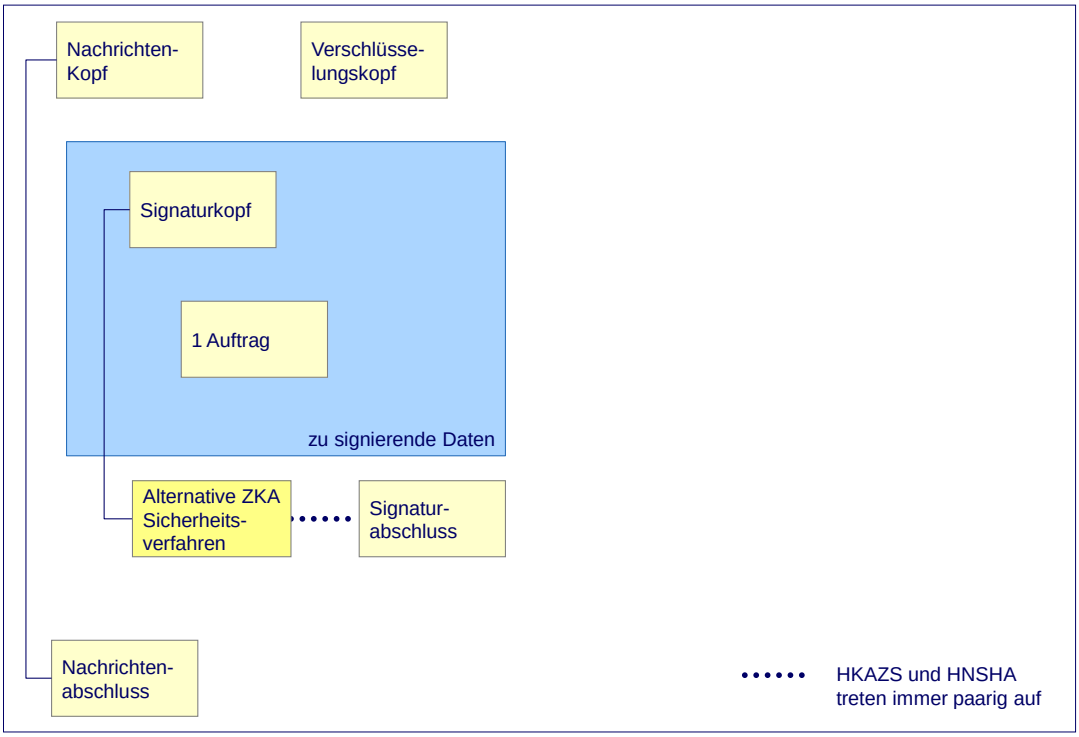


Abbildung 17: Segmentaufbau der Auftragsnachricht (Kunde) bei AZS-Verfahren

C.3.4 Kreditinstitutsauftragsnachricht

Die folgende Abbildung zeigt den generellen Aufbau der Kreditinstitutsnachricht bei der Auftragsverarbeitung. In die Signaturbildung fließen die Segmente Signaturkopf (HNSHK), ggf. Auftragsantwortsegmente und die Rückmeldungssegmente (HIRMG und HIRMS) ein.

Beim AZS-Verfahren 811 entfällt das Segment HIAZS.

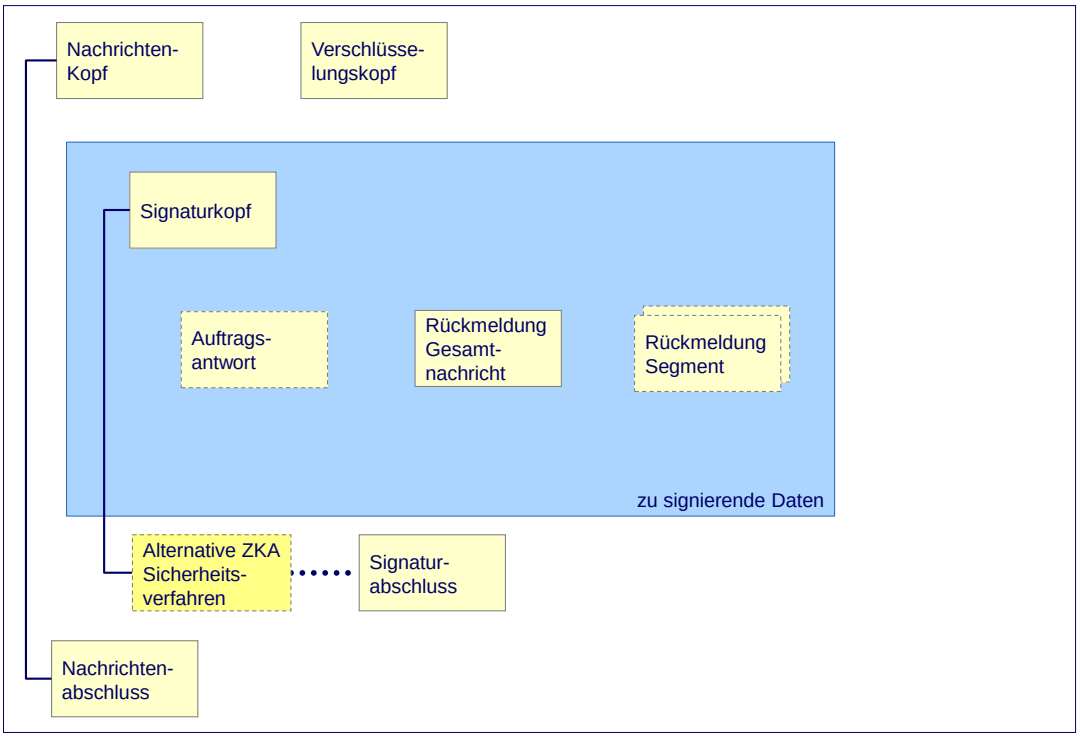


Abbildung 18: Segmentaufbau der Auftragsnachricht (Kreditinstitut) bei AZS-Verfahren

C.3.5 Kundennachricht bei Mehrfachsignaturen

Beim AZS-Verfahren 811 ist es möglich, mehrere Signaturen im Rahmen der Auftragseinreichung mitzuschicken, wie dies beim FinTS-Ein-Schritt-Verfahren grundsätzlich möglich ist. In Kombination mit den zugehörigen HKAZS-Segmenten ergibt sich folgender Segmentaufbau:

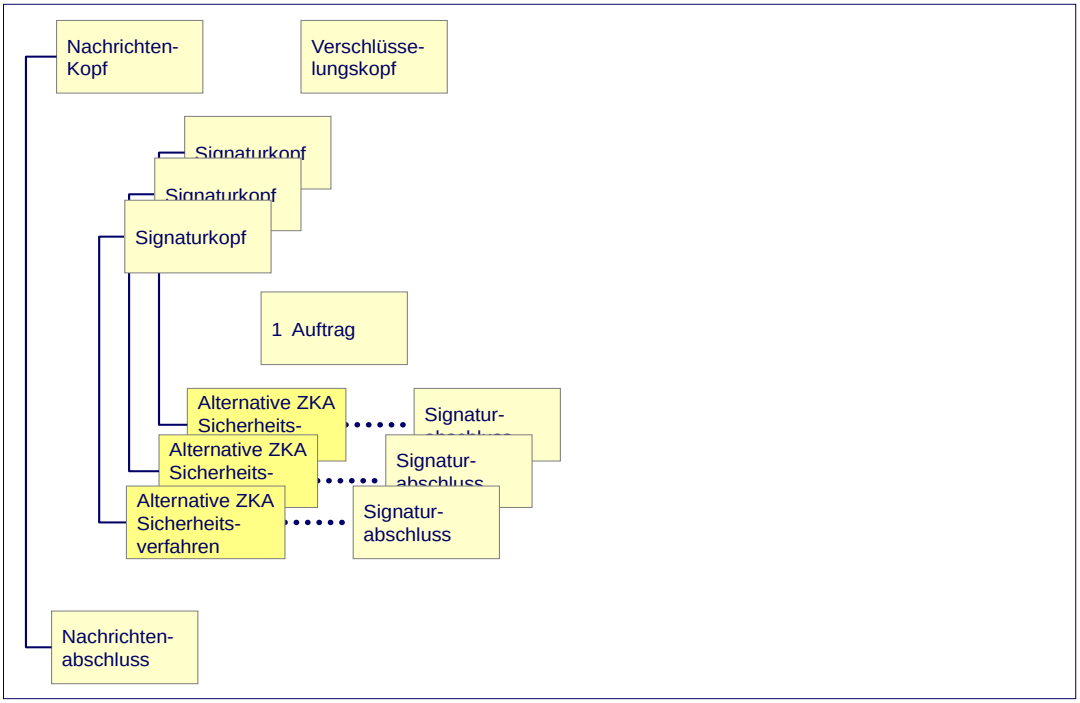


Abbildung 19: Kundennachricht bei Mehrfachsignaturen

Mehrfachsignaturen werden beim AZS-Verfahren 811 wie beim Sicherheitsverfahren HBCI von außen nach innen verarbeitet.

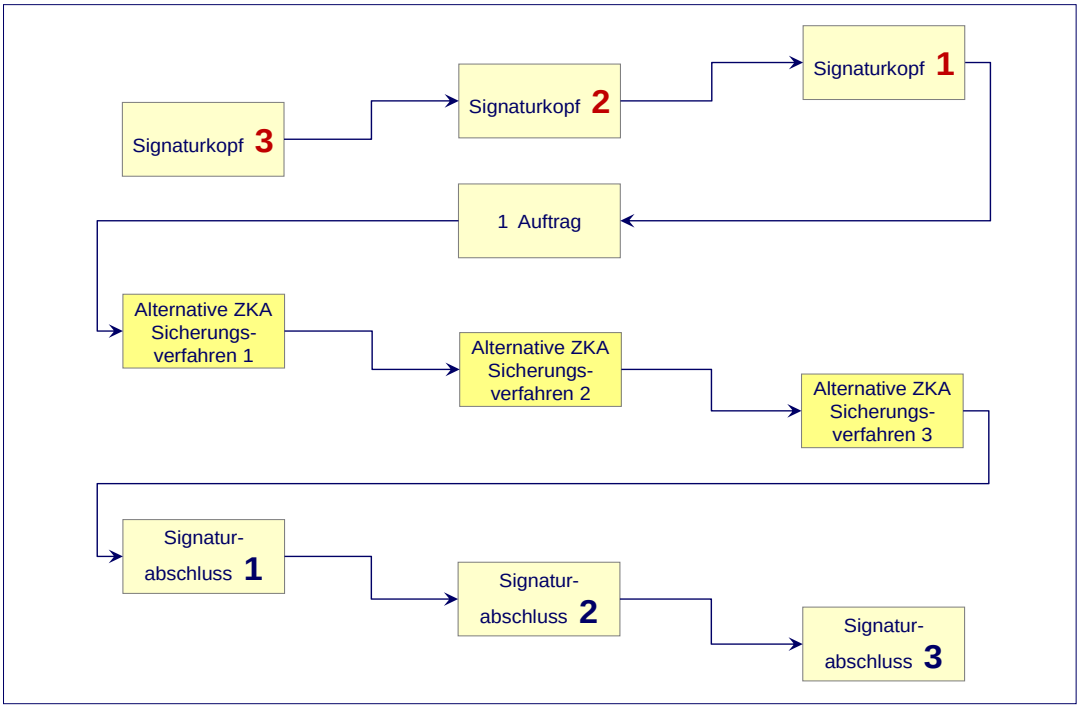


Abbildung 20: Reihenfolge der Sicherheitssegmente bei Mehrfachsignaturen

Die Referenzierung zwischen den Segmenten geschieht in den drei Segmenten Signaturkopf, Signaturabschluss und HKAZS mit Hilfe der Sicherheitskontrollreferenz.

Bei Mehrfachsignaturen wird das Datenelement „Bereich der Sicherheitsapplikation, kodiert“ mit dem Wert 1 (= SHM) belegt, d. h. die Sicherheitselemente der anderen Signierenden werden nicht mit signiert und die Reihenfolge der Signaturen ist daher nicht von Belang.

Alle anderen Belegungen wie z. B. die „Rolle des Sicherheitslieferanten, kodiert“ sind identisch zu den Aussagen bzgl. Mehrfachsignaturen bei Sicherheitsverfahren HBCI.

D. SECODER-MANAGEMENT

Der im Folgenden beschriebene Geschäftsvorfall HKHSI ist identisch im FinTS-Band [PIN/TAN] erläutert. Dort befindet sich auch die Spezifikation aller vorkommenden Elementversionen bestimmter Datenelemente; hier sind nur die Secoder-spezifischen Elemente dokumentiert.

D.1 Übermitteln / Anzeigen von Secoder-Informationen

Dieser Geschäftsvorfall dient dazu, Informationen über die Eigenschaften eines TAN-Generators (HHD) oder Secoders vom Kundenprodukt an das Kreditinstitut zu senden. Das Kreditinstitut kann mit diesen Daten zum Einen seine eigene Bestandsverwaltung pflegen, aber auch entsprechende Informationen, die sich aus den übertragenen Daten ergeben, zurück melden.

So kann z. B. ein Kunde die eindeutige Reader-ID seines TAN-Generators ermitteln (per HotKey oder durch die Challenge-Klasse 09 seines HHD – vgl. [HHD]) und diese an das Kreditinstitut übermitteln. Durch Interpretation der Reader-ID kann das Institut z. B. Hersteller, Gerätetyp und Version der Firmware ermitteln und in der Kreditinstitutsantwort an den Kunden übertragen.

Realisierung Bank: optional

Realisierung Kunde: optional

a) Kundenauftrag

Format

Name: HHD/Secoder-Informationen übermitteln
Typ: Segment
Segmentart: Geschäftsvorfall
Kennung: HKHSI
Bezugssegment: -
Segmentversion: 1
Sender: Kunde

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Segmentkopf	DEG			M	1	
2	TAN-Medium-Klasse	DE	code	1	M	1	G, S
3	Reader-ID	DE	id	#	C	1	M: bei DE „TAN-Medium-Klasse“ = „G“ und DE „Reader-ID erforderlich“ = „J“ O: bei DE „TAN-Medium-Klasse“ = „G“ und DE „Reader-ID erforderlich“ = „N“ N: sonst
4	Verfahrensbestätigung	DE	jn	#	C	1	M: bei DE „Verfahrensbestätigung erforderlich“ = „J“ (BPD) O: sonst

Belegungsrichtlinien

TAN-Medium-Klasse[2]

Als TAN-Medium-Klasse kann entweder „G“ für TAN-Generator bzw. HHD oder „S“ für Secoder angegeben werden.

Reader-ID

Die Reader-ID kann belegt werden, wenn diese institutsseitig nicht bekannt ist und abgeglichen bzw. erfasst werden soll. Durch den BPD-Parameter „Reader-ID erforderlich“ kann gesteuert werden, ob die Angabe der Reader-ID zwingend für die Ausführung des Geschäftsvorfalls erforderlich ist.

[2] Es ist die Elementversion 2 der TAN-Medium-Klasse zu verwenden.

b) Kreditinstitutsrückmeldung

Erläuterungen

Es wird ein Datensegment zurückgemeldet.

Format

Name: HHD/Secoder Informationen rückmelden
Typ: Segment
Segmentart: Geschäftsvorfall
Kennung: HIHSI
Bezugssegment: HKHSI
Segmentversion: 1
Sender: Kreditinstitut

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Segmentkopf	DEG			M	1	
2	Reader-ID	DE	id	#	O	1	
3	Gerätehersteller	DE	an	..64	O	1	
4	Geräteklasse	DE	an	..64	O	1	
5	Gerätebezeichnung	DE	an	..64	O	1	
6	Geräteversion	DE	an	..64	O	1	

Ausgewählte Beispiele für Rückmeldungscodes

Code	Beispiel für Rückmeldungstext
0020	Auftrag verarbeitet

c) Bankparameterdaten

Format

Name: HHD/Secoder Informationen Parameter
Typ: Segment
Segmentart: Geschäftsvorfall
Kennung: HIHSIS
Bezugssegment: HKVVB
Segmentversion: 1
Sender: Kreditinstitut

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Segmentkopf	DEG			M	1	
2	Maximale Anzahl Aufträge	DE	num	..3	M	1	
3	Anzahl Signaturen mindestens	DE	num	1	M	1	0, 1, 2, 3
4	Sicherheitsklasse	DE	code	1	M	1	0, 1, 2, 3, 4
5	Parameter HHD/Secoder Informationen	DEG			M	1	

E. DATA-DICTIONARY

A

Anzahl Signaturen mindestens	
	Mindestanzahl der Signaturen, die für einen Geschäftsvorfall als erforderlich definiert ist. Vom Kreditinstitut wird immer die Minimalanforderung an einen Geschäftsvorfall mitgeteilt, d. h. '0', wenn der Geschäftsvorfall auch über den anonymen Zugang angeboten wird, ansonsten mindestens '1', da Aufträge von Kunden immer signiert werden müssen. Die für Kunden jeweils genaue Angabe der Signaturanzahl ergibt sich in den UPD aus dem DE „Anzahl benötigter Signaturen“. Dabei muss die in den UPD angegebene Signaturanzahl größer oder gleich der in den BPD angegebenen Anzahl sein. Für Institute, die keine UPD unterstützen, bedeutet dies, dass der Eintrag '0' in den BPD nur für Nichtkunden gilt und für Kunden als 'mindestens 1' zu interpretieren ist. Der Wert gilt für alle Signaturverfahren. Typ: DE Format: num Länge: ..1 Version: 1

B

Bezugssegment	
	Sofern sich ein Kreditinstitutssegment auf ein bestimmtes Kundensegment bezieht (z. B. Antwortrückmeldung auf einen Kundenauftrag) hat das Kreditinstitut die Segmentnummer des Segments der Kundennachricht einzustellen, auf das sich das aktuelle Segment bezieht (s. DE „Segmentnummer“). In Zusammenhang mit den Angaben zur Bezugsnachricht aus dem Nachrichtenkopf ist hierdurch eine eindeutige Referenz auf das Segment einer Kundennachricht möglich. Falls die Angabe eines Bezugssegments erforderlich ist, ist dieses bei der Formatbeschreibung eines Kreditinstitutssegments angegeben. Typ: DE Format: num Länge: ..3 Version: 1

D

Dialog-ID	
	Die Dialog-ID dient der eindeutigen Zuordnung einer Nachricht zu einem HBCI-Dialog. Die erste Kundennachricht (Dialoginitialisierung) enthält als Dialog-ID den Wert 0. In der ersten Antwortnachricht wird vom Kreditinstitut eine Dialog-ID vorge-

geben, die für alle nachfolgenden Nachrichten dieses Dialogs einzustellen ist. Es ist Aufgabe des Kreditinstituts, dafür zu sorgen, dass diese Dialog-ID dialogübergreifend und systemweit eindeutig ist.

Typ: DE

Format: id

Länge: #

Version: 1

G

Geschäftsvorfallspezifische Informationen zum alternativen Sicherheitsverfahren

Ein DE dieses Typs enthält für genau einen Geschäftsvorfall relevante Informationen zum gewählten alternativen Sicherheitsverfahren. Ist für einen Geschäftsvorfall ein zugehöriges DE hinterlegt, kann das Kundenprodukt diesen Geschäftsvorfall über das alternative Sicherheitsverfahren absichern, andernfalls ist dies nicht erlaubt.

Hierdurch wird nicht festgelegt, ob und wie oft ein Geschäftsvorfall zu signieren ist. Dies wird weiterhin über die BPD und UPD angegeben. Werden in BPD und UPD keine Signaturen gefordert, können diese selbst dann weggelassen werden, wenn für den betreffenden Geschäftsvorfall laut diesem DE eine Signatur erforderlich ist.

Da die einzelnen Bestandteile dieses Datenelements aus syntaktischen Gründen nicht mehr modelliert werden können, werden diese mittels „Semikolon“ („;“ – 0x3B nach FinTS-Zeichensatz) getrennt.

Im Feld „Segmentkennung“ ist die Kennung des Auftragssegments des Geschäftsvorfalls anzugeben, auf den sich die Informationen zum alternativen Sicherheitsverfahren beziehen.

Beim Einsatz der AZS-Verfahren müssen für die in HIVISS enthaltenen Segmentkennungen und Segmentversionen auch entsprechende Einträge in HIAZSS enthalten sein.

Beim AZS-Verfahren 811 werden für Segmentkennungen und Segmentversionen die in HIAZSS aber nicht in HIVISS enthalten sind und für die somit keine Secoder-Visualisierung erfolgen soll, Signaturen nach Signatur-Prozess=7 erzeugt.

Die Elemente Nr. 4 und 5 sind mit FinTS-Füllwerten belegt und werden in der nächsten Segmentversion entfernt.

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Segmentkennung	DE	an	..6	M	1	
2	Segmentversion	DE	num	..3	M	1	
3	SAK-Visualisierungsformat	DE	code	1	M	1	0
4	Zeit- und Dialogbezug beim alternativen Signaturverfahren	DE	code	1	M	1	1
5	Sicherheitsfunktion, kodiert	DE	code	3	O	..99	

Typ: DE

Format: an

Länge: ..512

Version: 1

Geschäftsvorfallspezifische Visualisierungsinformationen für Secoder

In diesem Datenelement wird eine Sequenz von Secoder-MetaData-Elementen beschrieben, d. h. bezogen auf einen Geschäftsvorfall ein Anzeigetext und die zugehörigen Daten aus dem Auftrag. Je Geschäftsvorfall können so viele Secodervisualisierungstexte definiert werden, wie der Secoder erlaubt.

Da die einzelnen Bestandteile eines Secoder-Visualisierungselementes aus syntaktischen Gründen nicht mehr modelliert werden können, werden diese mittels „Semikolon“ („;“ – 0x3B nach FinTS-Zeichensatz) getrennt. Die Struktur beginnt somit mit der Segmentkennung und der Segmentversion des Geschäftsvorfalls und der Sicherheitsfunktion, gefolgt von n Secoder-Visualisierungselementen, die rekursiv dargestellt werden.

Die Informationen Secodervisualisierung Finanzdatenformat und –Position müssen immer paarweise auftreten und deren Anzahl muss den Platzhaltern „#“ in den Secodervisualisierungstexten entsprechen.

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Segmentkennung	DE	an	..6	M	1	
2	Segmentversion	DE	num	..3	M	1	0, Seg-Vers.
3	Sicherheitsfunktion, kodiert	DE	code	..3	M	1	0, 811
3	Secodervisualisierungstext Index	DE	num	..3	M	1	
4	Secodervisualisierung Finanzdatenformat	DE	code	1	K	1	
5	Secodervisualisierung Position	DE	an	..256	K	1	

Typ: DE

Format: an

Länge: ..16KB

Version: 1

M

Maximale Anzahl Aufträge

Höchstens zulässige Anzahl an Segmenten der jeweiligen Auftragsart je Kundennachricht. Übersteigt die Anzahl der vom Kunden übermittelten Segmente pro Auftragsart die zugelassene Maximalanzahl, so wird die gesamte Nachricht abgelehnt.

Typ: DE

Format: num

Länge: ..3

Version: 1

P

Parameter Alternative ZKA Sicherheitsverfahren

Für die Kennzeichnung der alternativen ZKA Sicherheitsverfahren auf Basis von HKAZS stehen definierte Werte als „Sicherheitsfunktion, kodiert“ im Wertebereich von 800 bis 899 zur Verfügung.

Die Elemente Nr. 1 und 2 sind mit FinTS-Füllwerten belegt und werden in der nächsten Segmentversion entfernt.

Die Elemente Nr. 3 und 4 werden beim AZS-Verfahren 811 nicht genutzt und können weggelassen werden.

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Visualisierungsbestätigungssignaturfüllwert	DE	code	1	M	1	1
2	Auftragsstorno erlaubt	DE	jn	#	M	1	N
3	Zulässige Signaturanwendungskomponenten	DE	an	..1KB	O	1	nicht unterstützt
4	Unterstützte EMV AC Versionen	DE	num	1	O	..9	nicht unterstützt
5	Geschäftsvorfallspezifische Informationen zum alternativen Sicherheitsverfahren	DE	an	..512	M	n	

Typ: DEG

Format:

Länge:

Version: 1

Parameter HHD-/Secoder-Informationen

Auftragsspezifische Bankparameterdaten für den Geschäftsvorfall „HHD-/Secoder-Informationen übermitteln“. Der Parameter „Verfahrensbestätigung erforderlich“ wird beim Secoder nicht verwendet und ist mit „N“ als FinTS Füllwert zu belegen.

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Reader-ID erforderlich	DE	jn	#	M	1	

2	Verfahrensbestätigung erforderlich	DE	jn	#	M	1	N

Typ: DEG
Format:
Länge:
Version: 1

Parameter Secoder-spezifische Visualisierungsinformationen

Mit dieser DEG können Visualisierungstexte und eine geschäftsvorfallabhängige Darstellung festgelegt werden.

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Secodervisualisierungstexte	DE	an	..16KB	M	1	
2	Geschäftsvorfallspezifische Visualisierungsinformationen für Secoder	DE	an	..4KB	M	n	

Typ: DEG
Format:
Länge:
Version: 1

R

Reader-ID

Eindeutige Identifikationsnummer eines HHD bzw. eines Secoders.

Typ: DE
Format: id
Länge: #
Version: 1

Reader-ID erforderlich

Über diesen Parameter wird festgelegt, ob die Übertragung der Reader-ID zwingend erforderlich ist oder optional erfolgen kann. So kann ein Kreditinstitut die Übertragung der Reader-ID verlangen, wenn keine zentralen Bestände zur Verfügung stehen oder die Reader-ID für eine zentrale Verwaltung erfasst werden soll.

Typ: DE
Format: jn
Länge: #
Version: 1

S

Secodervisualisierungstexte

Das DE Secodervisualisierungstexte beinhaltet eine Tabelle aller möglichen Anzeigendefintionen für das Secoderdisplay. Die Länge der einzelnen Anzeigedefintionen muss sich an der Zeilengröße der zugelassenen Secoderprodukte am Markt orientieren. Momentan sind dies 16 Zeichen, es sind jedoch bis zu 32 Zeichen möglich. Ein einzelner Eintrag der Tabelle besteht aus einem maximal dreistellig numerischen Index und zugehörigen Secoder MetaData (vgl. Abschnitt „Begriffe“) zur Ansteuerung des Secoders durch eine geeignete aktive Komponente.

Index und Text sowie die einzelnen Index-Text-Kombinationen werden durch „Semikolon“ („;“ – 0x3B nach FinTS-Zeichensatz) getrennt. Die Texte selbst dürfen das Zeichen ‚Semikolon‘ nicht enthalten. Die Kodierung erfolgt auf Basis der MetaData im FinTS-Zeichensatz. Die Konvertierung in den ISO 646 DE Zeichensatz des Secoders wird durch die Secoder-Anwendungsfunktion durchgeführt. Der Zeichenvorrat wird allerdings durch die Festlegungen in der Secoder-Spezifikation [Secoder] bestimmt, nicht durch den FinTS-Zeichenvorrat.

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Secodervisualisierung Index	DE	num	..3	M	1	
2	Displayposition	DE	code	1	O	1	L, R
3	Länge Secoder-Text	DE	num	..2	O	1	
4	Secoder-Text	DE	an	..32	O	1	
5	Länge Secoder Eingabedaten	DE	num	..2	O	1	0
6	Ausrichtung und Format Secoder Eingabedaten	DE	dig	2	O	1	00, 01, 03, 04, 05, 07
7	Secoder Padding	DE	an	2	O	1	D0, DF, E0, EF

Das Element „Länge Secoder Eingabedaten“ muss beim AZS-Verfahren 811 mit 0 belegt werden, da kein Eingeben / Ergänzen von Daten erlaubt ist.

Die detailliierte Beschreibung der Secoder MetaData befindet sich in Abschnitt B.5.1.1. Eine weitere Untergliederung der Elemente im Data Dictionary erfolgt nicht.

Typ: DE

Format: an

Länge: ..16384

Version: 1

Secodervisualisierungstext Index	
	Index auf die Tabelle der Secodervisualisierungstexte. Ein Index muss auf einen gültigen Tabelleneintrag verweisen. Typ: DE Format: num Länge: ..3 Version: 1

Secodervisualisierung Finanzdatenformat	
	Die Adressierung der Secodervisualisierungsdaten innerhalb eines Geschäftsvorfalls oder Finanzdatenformats ist abhängig von dessen Typ. Folgende Datenformate sind für die Secodervisualisierung vorgesehen: Codierung: 1: FinTS 2: DTA 3: DTAZV 4: SEPA Typ: DE Format: code Länge: 1 Version: 1

Secodervisualisierung Position

Je nach Datenformat gibt es unterschiedliche Möglichkeiten der Adressierung der Secodervisualisierungsdaten im Auftrag:

bei FinTS: <DE#> oder

< DEG#>,< DE# in der DEG>

bei DTA <Datensatz>.<Feld im Datensatz>

bei DTAZV: <Datensatz>.<Feld im Datensatz>

bei SEPA: <Tagname>.<Index>

Bzgl. der Nummern sind die entsprechenden Definitionen der Geschäftsvorfälle bzw. Finanzdatenformate heranzuziehen.

Typ: DE

Format: an

Länge: ..256

Version: 1

Sicherheitsfunktion, kodiert

Kodierte Information über die Sicherheitsfunktion, die auf die Nachricht angewendet wird.

Bis HBCI 2.2:

dient der Unterscheidung zwischen DDV und RDH, wobei die 1 das RDH-Verfahren kennzeichnet und 2 das DDV-Verfahren.

FinTS V3.0 – Sicherheitsverfahren HBCI:

Die Sicherheitsfunktion hat ab FinTS 3.0 lediglich informatorischen Wert, da die eigentliche Steuerung über die Sicherheitsprofile und –klassen erfolgt.

FinTS V3.0 – Sicherheitsverfahren PIN/TAN:

Codierung der verwendeten Sicherheits- und Verschlüsselungsfunktionen

FinTS V3.0 – Alternative ZKA Sicherheitsverfahren:

Dient der Kennzeichnung des jeweiligen Verfahrens in Verbindung mit dem Geschäftsvorfall HKAZS

Codierung:

Code	Segment	Bedeutung
1	Sicherheitsverfahren HBCI: - Signaturkopf	Non-Repudiation of Origin, für RDH/RAH (NRO)
2	Sicherheitsverfahren HBCI: - Signaturkopf	Message Origin Authentication, für RDH/RAH und DDV (AUT)
4	Sicherheitsverfahren HBCI:	Encryption, Verschlüsselung und evtl. Komprimie-

	- Verschlüsselungskopf	rung (ENC)
811	Alternative ZKA Sicherheitsverfahren: - Signaturkopf bei HKAZS, - HIAZSS Verfahrensparameter	Fortgeschrittene Elektronische Signatur („AUT-Signatur“) mit Secoder ohne Institutssignatur
900	Sicherheitsverfahren PIN/TAN: - Signaturkopf bei HKTAN, - HITANS Verfahrensparameter Zwei-Schritt-Verfahren	1. konkretes Zwei-Schritt-TAN-Verfahren
901	Sicherheitsverfahren PIN/TAN: - Signaturkopf bei HKTAN, - HITANS Verfahrensparameter Zwei-Schritt-Verfahren	2. konkretes Zwei-Schritt-Verfahren
. . .		
996	Sicherheitsverfahren PIN/TAN: - Signaturkopf bei HKTAN, - HITANS Verfahrensparameter Zwei-Schritt-Verfahren	97. konkretes Zwei-Schritt-Verfahren
997	Sicherheitsverfahren PIN/TAN: - Signaturkopf bei HKTAN, - HITANS Verfahrensparameter Zwei-Schritt-Verfahren	98. konkretes Zwei-Schritt-Verfahren
998	Sicherheitsverfahren PIN/TAN: - Verschlüsselungskopf	Daten im Klartext (nur in Verbindung mit SSL erlaubt)
999	Signaturkopf	Klassisches Ein-Schritt-Verfahren

Typ: DE

Format: Code

Länge: ..3

Version: 2

Sicherheitskontrollreferenz

Referenzinformation, mit der die Verbindung zwischen Signaturkopf, dazu gehörigem Signaturabschluss und HKAZS hergestellt werden kann. Die Sicherheitskontrollreferenz im Signaturkopf muss mit der entsprechenden Information im Signaturabschluss und in HKAZS übereinstimmen.

Typ: DE

Format: an

Länge: ..14

Version: 1

Sicherheitsprofil

Verfahren zur Absicherung der Transaktionen, das zwischen Kunde und Kreditinstitut vereinbart wurde. Das Sicherheitsprofil wird anhand der Kombination der beiden Elemente „Sicherheitsverfahren“ und „Version“ bestimmt (z. B. RDH-3, DDV-1, PIN-1, EMV-2). Für das Sicherheitsverfahren PINTAN ist als Code der Wert PIN und als Version der Wert 1 einzustellen.

Nr.	Name	Typ	Format	Länge	Status	Anzahl	Restriktionen
1	Sicherheitsverfahren, Code	DE	code	3	M	1	DDV, RAH, RDH, PIN
2	Version des Sicherheitsverfahrens	DE	num	..3	M	1	1, 2, 3, 4, 5, 6, 7, 8, 9, 10

Typ: DEG

Format:

Länge:

Version: 1

Sicherheitsverfahren, Code

Code des unterstützten Signatur- bzw. Verschlüsselungsalgorithmus. Weitere Informationen zu den Verfahren sind Kapitel B.1 [HBCI] zu entnehmen.

Codierung:.

DDV: DES-DES-Verfahren

RAH: RSA-AES-Hybridverfahren, AZS-Verfahren mit S-Fkt=811

RDH: RSA-DES-Hybridverfahren, AZS-Verfahren mit S-Fkt=811

PIN: PIN/TAN-Verfahren

Typ: DE

Format: code

Länge: 3

Version: 3

Signaturdaten

Ergebnis der Signaturbildung bei AZS-Verfahren, unabhängig von der Art der erzeugten Signatur.

Folgende Inhalte sind für das DE Signaturdaten möglich:

S-Fkt.	Inhalt des DE „Signaturdaten“
811	Fortgeschrittene Signatur, wie sie mit Hilfe der Secoder-Applikation „aut“ erzeugt wurde (VisDataSig) oder HBCI RAH- bzw. RDH-Signatur bei Signatur-Prozess=7

Typ: DE

Format: bin

Länge: ..4096

Version: 1

Signatur-Prozess	
	Bei der Verwendung der Elektronischen Signatur werden die notwendigen Prozess-Schritte mittels des Geschäftsvorfalls HKAZS durchgeführt. Der Signatur-Prozess wird wie folgt kodiert: Signatur-Prozess=6: Kennzeichnet die Auftragseinreichung inkl. aller Sicherheitsinformationen mit beim AZS-Verfahren S-Fkt=811, soweit es sich um Secoder-Signaturen inklusive Secoder-Visualisierung handelt. Signatur-Prozess=7: Kennzeichnet Signaturen mit dem AZS-Verfahren S-Fkt=811, bei denen keine Secoder-Visualisierung erfolgt, also das native RAH- / RDH-Verfahren genutzt wird. Typ: DE Format: Code Länge: 1 Version: 1

T

TAN-Medium-Klasse	
	dient der Klassifizierung der möglichen TAN-Medien. Bei Geschäftsvorfällen zum Management der TAN-Medien kann aus diesen nach folgender Codierung selektiert werden. Codierung: L: Liste G: TAN-Generator M: Mobiltelefon mit mobileTAN S: Secoder Typ: DE Format: code Länge: 1 Version: 2

V

Version des Sicherheitsverfahrens
Version des unterstützten Sicherheitsverfahrens (s. „Sicherheitsverfahren, Code“). In Kombination mit den HBCI-Sicherheitsverfahren RAH bzw. RDH sind die folgenden Versionen gültig:

Version	Signaturverfahren	Schlüssellänge (bit)	Hashverfahren	Schlüsselart*
1	ISO 9796-1	708-768	RIPEMD-160	S, V
2	DIN, ISO 9796-2	1024-2048	RIPEMD-160	S, V
3	DIN, ISO 9796-2 PKCS#1 V1.5	1024-2048	RIPEMD-160 SHA-1	D, S, V
4	PKCS#1 V1.5	1024-2048	SHA-1	D, S, V
5	PKCS#1 V1.5	1024-2048	SHA-1	S,V
6	PKCS#1 V1.5	1536-4096	SHA-256	D, S, V
7	PKCS#1 PSS	1536-4096	SHA-256	D, S, V
8	PKCS#1 V1.5	1536-4096	SHA-256	S, V
9	PKCS#1 PSS	1536-4096	SHA-256	S, V
10	PKCS#1 PSS	1536-4096	SHA-256	S, V

In Kombination mit dem Sicherheitsverfahren DDV sind die folgenden Versionen gültig (nicht bei AZS-Verfahren zugelassen!):

Version	Signaturverfahren	Schlüssellänge (bit)	Hashverfahren	Schlüsselart*
1	MAC	128	RIPEMD-160	S, V
2	MAC	128	SHA-256	S, V

Bei Verwendung des Sicherheitsverfahrens PIN/TAN sind die folgenden Versionen gültig (nicht bei AZS-Verfahren zugelassen!):

Version	Signaturverfahren	Schlüssellänge (bit)	Hashverfahren	Schlüsselart*
1	PINTAN			
2	PINTAN			

Bei Verwendung der chipkartenbasierten AZS-Verfahren können bei der S-Fkt=811 in Verbindung mit RAH / RDH folgenden Versionen auftreten:

Version	Signaturverfahren	Schlüssellänge (bit)	Hashverfahren	Schlüsselart*
3	DIN, ISO 9796-2 PKCS#1 V1.5	1024-2048	RIPEMD-160 SHA-1	D, S, V
5	PKCS#1 V1.5	1024-2048	SHA-1	S,V
6	PKCS#1 V1.5	1536-4096	SHA-256	D, S, V
7	PKCS#1 PSS	1536-4096	SHA-256	D, S, V
8	PKCS#1 V1.5	1536-4096	SHA-256	S, V
9	PKCS#1 PSS	1536-4096	SHA-256	S, V

* s. Element „Schlüsselart“

Andere als die genannten Profile sind nicht zulässig.

Typ: DE

Format: num

Länge: ..3

Version: 3

Visualisierungsbestätigungssignaturfüllwert	
	Die Secoder-Spezifikation lässt zwei unterschiedliche Arten von Fülldaten bei der Bildung einer Visualisierungssignatur zu. In FinTS wird nur der konstante String mit Code=1 unterstützt. Hinweis: dieses DE wird in der nächsten Segmentversion entfernt werden. Codierung: 1: String "SECODERSECODERSECODE" ~~2: Eindeutige Reader ID, mit "SECODERSECODERSECODE" aufgefüllt auf 20 bytes~~ Typ: DE Format: code Länge: 1 Version: 1

Visualisierungsbestätigungssignaturdaten	
	Ergebnis der Signaturbildung über die zwischengespeicherten visualisierten Daten beim Secoder in einem extra Schritt. Dadurch wird sichergestellt, dass ein Secoder sich zum Zeitpunkt der Signaturbildung im Applikationsmodus befand. Die Visualisierungsbestätigungssignatur wird mit der Applikation „aut" gebildet. Typ: DE Format: bin Länge: ..4096 Version: 1

F. ANLAGEN

F.1 Übersicht der Segmente

Nr.	Segmentname	Kennung	Sender[3]	Version
1	Alternative ZKA-Sicherungsverfahren	HKAZS	K	1
2	Alternative ZKA-Sicherungsverfahren Parameter	HIAZSS	I	1
3	Alternative ZKA-Sicherungsverfahren Rückmeldung	HIAZS	I	1
4	Secoderspezifische Visualisierungsinformationen	HIVISS	I	1

F.2 Übersicht Nachrichtenaufbau

Segment	Nachricht					
	Dialoginitialisierung		Auftragsnachricht		Dialogbeendigung	
	Kunde	Kredit-	Kunde	Kredit-	Kunde	Kredit-
	N6	N2	N15	N14	N8	N14
Nachricht	1	1	0-n	0-n	1	1
HNHBK	1	1	1	1	1	1
HNVSK	1	1	1	1	1	1
HNVSD	1	1	1	1	1	1
HNSHK	1	0-1	1-3	0-1	1	-
HIRMG	-	1	-	1	-	1
HIRMS	-	0-m	-	0-m	-	0-m
HKIDN	1	-	-	-	-	-
HKVVB	1	-	-	-	-	-
HKISA	0-2	-	-	-	-	-
HKSYN	-	-	-	-	-	-
HIBPA	-	0-1	-	-	-	-
HIKOM	-	0-1	-	-	-	-
HISHV	-	0-1	-	-	-	-
HIKPV	-	0-1	-	-	-	-
HIUEBS	-	0-n	-	-	-	-
...[4]	-	0-n	-	-	-	-
HIAZSS	-	0-1	-	-	-	-
HIVISS	-	0-1	-	-	-	-
HIUPA	-	0-1	-	-	-	-
HIUPD	-	0-n	-	-	-	-
HIISA	-	0-2	-	-	-	-
HISYN	-	-	-	-	-	-
HIKIM	-	0-n	-	-	-	-

[3] K: Kunde, I: Kreditinstitut

[4] Hier sind für die weiteren unterstützten Geschäftsvorfälle die entsprechenden Parameter-Segmente einzustellen.

Segment	Nachricht					
	Dialoginitialisierung		Auftragsnachricht		Dialogbeendigung	
	Kunde	Kredit-	Kunde	Kredit-	Kunde	Kredit-
	N6	N2	N15	N14	N8	N14
HKSAL[5]	-	-	1	-	-	-
HISAL	-	-	-	0-n	-	-
...	-	-	-	-	-	-
HKAZS	1	-	0-3	-	-	-
HIAZS	-	0-1	-	0-1	-	-
HKPRO	-	-	0-1	-	-	-
HIPRO	-	-	-	0-n	-	-
HKEND	-	-	-	-	1	-
HNSHA	1	0-1	1-3	0-1	1	-
HNHBS	1	1	1	1	1	1

[5] Exemplarisch wird hier der Geschäftsvorfall „Saldenabfrage“ angenommen.