



Spezifikation

FINTS - Format der RDH-2-Sicherheitsdiskette

Dokument-	
version:	1.0
Status:	Entwurf
Datum:	12.02.2004
Autoren:	R. Segebrecht
	M. Wendt

Versionsführung für Dokument **Spezifikation der FinTS RDH-2 Sicherheitsdiskette V1.0.doc**

Name	Datum	Doku-ment-version	Bemerkungen
R. Segebrecht M. Wendt	12.02.2004	1.0	Initialversion

Copyright
Dieses Dokument wurde von PPI Financial Systems GmbH erstellt und ist gegenüber Dritten urheberrechtlich geschützt. Alle Rechte, auch die der Übersetzung, des Nachdrucks oder der Vervielfältigung des gesamten Dokumentes oder Teilen daraus, bedürfen der Zustimmung von PPI.

Inhaltsverzeichnis

1	Einleitung	4
2	Begriffsvereinbarungen	5
2.1	Kryptografische Primitive	5
2.1.1	Padding nach RFC 1423	5
2.1.2	Passwort Restriktionen.....	5
2.1.3	Schlüsselableitung.....	5
2.1.4	MAC-Bildung	5
2.1.5	Verschlüsselung	6
2.2	Typdefinitionen	6
2.3	TLV-Beschreibung	6
3	Dateiinhalte	7
3.1	Dateiinhalte Feld 1: Kopf der Diskettenbeschreibung	8
3.2	Dateiinhalte Feld 2: Kreditinstitutionsverbindung	9
3.2.1	Kreditinstitutionsverbindungsdaten.....	10
3.2.2	Schlüsseleintrag	12
3.3	Dateiinhalte Feld 3: Öffentliche Schlüssel des Kreditinstitutes	13
3.4	Dateiinhalte Feld 4: Datum	14
3.5	Dateiinhalte Feld 5: MAC	14
	Literaturverzeichnis	15

1 Einleitung

Diese Spezifikation beschreibt das Format der RDH-Sicherheitsdiskette für das RDH-2-Verfahren gemäß ZKA: *FinTS 3.0 Spezifikation* ([3]).

2 Begriffvereinbarungen

In den folgenden Abschnitten werden die in diesem Dokument gültigen Begriffe erläutert.

2.1 Kryptografische Primitive

Die verwendeten kryptografischen Definitionen werden nachfolgend dargelegt.

2.1.1 Padding nach RFC 1423

Das Padding wird nach RFC 1423 vorgenommen (siehe [2]). Dies geschieht wie folgt:

- Es wird auf das nächste Vielfache von 8 aufgefüllt, wobei folgende Paddings in hexadezimaler Darstellung möglich sind: 01, 0202, 030303, 04040404, 0505050505, 060606060606, 07070707070707 und 0808080808080808.
- Es werden 1 bis 8 Bytes gepadded.

2.1.2 Passwort Restriktionen

Das Passwort muss folgende Bedingungen erfüllen:

- die Mindestlänge beträgt 8 Zeichen
- es enthält mindestens eines der folgenden Sonderzeichen: . > < () + - & ? * ; , % : " ' \ =

2.1.3 Schlüsselableitung

Es wird die Schlüsselableitungsfunktion $PBKDF2$ des Kryptografie Standards PKCS #5 v2.0 (siehe [1]) verwendet, wobei Folgendes gilt:

- Das Salt ist auf der Diskette im Kopf der Diskettenbeschreibung (siehe Abschnitt *Dateiinhalte Feld 1: Kopf der Diskettenbeschreibung*, Seite 8) abgelegt.
- Die Anzahl der Iterationen ist im Kopf der Diskettenbeschreibung abgelegt.
- Das Passwort (siehe vorigen Abschnitt) wird vom Benutzer erfragt.

Des Weiteren wird bei der Schlüsselableitungsfunktion $PBKDF2$ (siehe [1]) ein HMAC mit SHA-1 gebildet.

2.1.4 MAC-Bildung

Der Schlüssel für die MAC-Bildung hat eine Länge von 20 Bytes und wird mittels Schlüsselableitung (siehe vorigen Abschnitt) gebildet.

Für die MAC-Bildung wird folgendes Authentifizierungsschema verwendet:

- HMAC-SHA-1 (siehe [1])

2.1.5 Verschlüsselung

Der Schlüssel DK (derived key) zur Verschlüsselung hat eine Länge von 24 Bytes und wird mittels Schlüsselableitung gebildet (siehe Abschnitt *Schlüsselableitung*, Seite 5). Für die einzelnen Verschlüsselungsschlüssel werden für *Key1* die Bytes 1-8, für *Key2* die Bytes 9-16 und für *Key3* die Bytes 17-24 verwendet.

Für die Verschlüsselung wird folgendes Verschlüsselungsschema verwendet:

- PKCS #5 (siehe [1]): DES-EDE3-CBC mit IV=0 und Padding nach RFC 1423, wobei für die Schlüsselableitung SHA-1 verwendet wird

2.2 Typdefinitionen

Sämtliche Längenangaben im Dokument erfolgen in Anzahl der belegten Bytes.

Folgende Typdefinitionen werden in dieser Spezifikation verwendet:

Format	Länge	Wertemenge	Beschreibung
Int	2		Little Endian Notation
Long	4		Little Endian Notation
Byte	Var	Binär	
Char	Var	ASCII-Zeichen	Inhalt linksbündig ausrichten und rechtsbündig mit Leerzeichen auffüllen
Keytype	1	X'00', X'01'	X'00' Signierschlüssel X'01' Chiffrierschlüssel

Bytefolgen, welche als Zahl interpretiert werden, sind grundsätzlich im Little Endian-Format abgelegt.

2.3 TLV-Beschreibung

Die im Weiteren benutzte Formatbeschreibung setzt auf dem TLV-Format (Tag-Length-Value) auf. Sie kann ggf. auf weitere Untergruppierungen verweisen. Das TLV-Format ist folgendermaßen aufgebaut:

Feld	Format	Länge	Anzahl	Inhalt	Beschreibung
1	Byte	2	1		Tag
2	Int	2	1	Länge von Feld 3	Länge
3	Byte	Var	1		Wert

3 Dateinhalt

Die Sicherheitsdiskette ist aus 5 Feldern aufgebaut, welche die Version der Diskette, die Kreditinstitutionsverbindungen des Kunden, die öffentlichen Schlüssel der Kreditinstitute, das letzte Änderungsdatum und den MAC enthalten.

Die Diskette wird durch den MAC und partielle Verschlüsselung gegen Missbrauch geschützt:

- Der Gesamtinhalt wird durch einen MAC gegen Verfälschung gesichert.
- Der geheime private Schlüssel eines Schlüsseleintrags wird durch Verschlüsselung gegen missbräuchlichen Zugriff geschützt.

Feld	Tag	Länge	Anzahl	Beschreibung
1	VN = X'564E'	26	1	Kopf der Diskettenbeschreibung (siehe Abschnitt <i>Dateinhalt Feld 1: Kopf der Diskettenbeschreibung</i> , Seite 8)
2	KV = X'4B56'	#	0..n	Kreditinstitutionsverbindung (siehe Abschnitt <i>Dateinhalt Feld 2: Kreditinstitutionsverbindung</i> , Seite 9)
3	ÖS = X'D653'	#	0..n	Öffentliche Schlüssel des Kreditinstituts (siehe Abschnitt <i>Dateinhalt Feld 3: Öffentliche Schlüssel des Kreditinstitutes</i> , Seite 13)
4	RD = X'5244'	14	1	Datum (siehe Abschnitt <i>Dateinhalt Feld 4: Datum</i> , Seite 14)
5	DM = X'444D'	20	1	MAC über die Felder 1 bis 4 (siehe Abschnitt <i>Dateinhalt Feld 5: MAC</i> , Seite 14)

Bemerkungen zur Belegung der Felder:

- Vor dem Arbeiten mit der Datei sollte diese geparkt werden und die Unversehrtheit durch Eingabe des Passwortes und anschließender Prüfung des MACs festgestellt werden. Danach können die Schlüssel genutzt werden.
- Es sind alle Felder lesbar mit Ausnahme der privaten geheimen Schlüssel (siehe Abschnitt *Geheimer Schlüssel*, Seite 12).
- Die Kreditinstitutionsverbindung wird durch weitere TLV-Records beschrieben.

- ❑ In jeder Kreditinstitutsverbindung existieren die für das Kreditinstitut erzeugten Schlüsseleinträge.
- ❑ Bei einem Schlüsselwechsel werden neue Schlüsseleinträge generiert. Sie gelten somit für genau diese Kreditinstitutsverbindung. Nach erfolgreicher Übertragung der neuen Schlüssel sind die alten Schlüsseleinträge zu löschen. Nach einem Wechsel des Signierschlüssels ist der Signaturzähler in den Kreditinstitutsverbindungsdaten zurückzusetzen.

3.1 Dateiinhalte Feld 1: Kopf der Diskettenbeschreibung

Feld	Format	Länge	Anzahl	Inhalt	Beschreibung
1	Int	2	1	1	Version der Diskettenbeschreibung
2	Byte	20	1		Salt
3	Long	4	1		Anzahl Iterationen

Bemerkungen zur Belegung der Felder:

- ❑ Die Version dieser Diskettenbeschreibung wird mit 1 festgelegt.
- ❑ Das Salt sollte kryptografisch einer möglichst breiten Verteilung genügen. Daher ist es sinnvoll, zur Erzeugung des Salt einen Zeitstempel zu verwenden, worüber mittels SHA-1 ein Hashwert gebildet wird (siehe [1]).
- ❑ Die Anzahl Iterationen sollte möglichst groß gewählt werden (siehe [1]).

3.2 Dateiinhalte Feld 2: Kreditinstitutionsverbindung

Eine Kreditinstitutionsverbindung besteht aus den Kreditinstitutionsverbindungsdaten, die dem Kunden vom Kreditinstitut mitgeteilt werden oder über Nachrichten ausgetauscht werden, und den Schlüsseleinträgen, die von der Kundenanwendung generiert und dem Kreditinstitut mitgeteilt werden.

Feld	Tag	Länge	Anzahl	Beschreibung
1	KD = X'4B44'	237	1	Kreditinstitutionsverbindungsdaten
2	SE = X'5345'	#	0, 2..4	Schlüsseleintrag

Bemerkungen zur Belegung der Felder:

- Die verschiedenen Kreditinstitutionsverbindungen eines Kunden werden durch Länderkennzeichen, Kreditinstitutionscode und Benutzerkennung referenziert. Diese finden sich in den Kreditinstitutionsverbindungsdaten.
- In den Schlüsseleinträgen sind alle Daten zu den Kundenschlüsseln abgelegt. Die Schlüsseleinträge müssen nicht vorhanden sein, um die Möglichkeit zu schaffen, sie erst beim Ablegen der Bankschlüssel in deren Länge anzulegen. Falls sie eingetragen werden, müssen minimal der Signier- und Chiffrierschlüssel vorhanden sein. Die maximale Anzahl der Einträge ist vier. Dann liegen zu den bereits übertragenen Signier- und Chiffrierschlüsseln jeweils noch nicht übertragene, neu generierte Signier- und Chiffrierschlüssel vor.
- Beim Generieren von neuen Schlüsseln erhöht sich die Anzahl der Schlüsseleinträge um die Anzahl der neu generierten Schlüssel.
- Neu generierte und durch eine Schlüsseländerung zu übertragende Schlüssel werden an der höheren Schlüsselversion im Schlüsseleintrag erkannt.

3.2.1 Kreditinstitutsverbindungsdaten

Länderkennzeichen, Kreditinstitutscode und Benutzerkennung stellen den Schlüssel für den Zugriff auf die Kreditinstitutsverbindungen dar. Jede Kreditinstitutsverbindung speichert eine Kommunikationsverbindung zum Kreditinstitut. Die Verbindung zu den öffentlichen Schlüsseln des Kreditinstituts wird über Länderkennzeichen und Kreditinstitutscode gebildet.

Feld	Format	Länge	Anzahl	Inhalt	Beschreibung
1	Char	3	1		Länderkennzeichen
2	Char	30	1		Kreditinstitutscode
3	Char	60	1		Kreditinstitutsname
4	Char	30	1		Benutzerkennung
5	Char	30	1		Kunden-ID
6	Char	30	1		Kundensystem-ID
7	Byte	1	1		Kommunikationsdienst
8	Char	50	1		Kommunikationsadresse
9	Int	2	1		Signaturzähler
10	Byte	1	1		Schlüsselstatus

Bemerkungen zur Tabelle:

- Der Kreditinstitutsname kann vom Benutzer frei gewählt werden.
- Die Benutzerkennung und Kunden-ID (optional) werden dem Benutzer vom Kreditinstitut zugewiesen.
- Die Kundensystem-ID wird im Zuge der Erstinitialisierung übermittelt. Initial ist sie mit dem Zeichen 0 zu belegen.
- Die Kommunikationsdaten werden dem Benutzer vom Kreditinstitut mitgeteilt.
- Der Signaturzähler gibt den nächsten gültigen Wert für die Signatur-ID mit dem Signaturschlüsseleintrag der niedrigeren Schlüsselnummer (d.h. des aktiven Schlüssels) an. Nach einem Wechsel des Signaturschlüssels (respektive Schlüsseleintragswechsel) wird dieser auf 1 zurückgesetzt.
- Der Schlüsselstatus enthält 8 Flags mit folgender Bedeutung:

Flag	Inhalt	Bedeutung
Bit 1	Ja=1, Nein=0	erstmalige Übermittlung der Kundenschlüssel notwendig
Bit 2	Ja=1, Nein=0	Institutsrechner erwartet Signaturen nach ISO9796 mit AnnexA
Bit 3	Ja=1, Nein=0	Institutsschlüssel validiert

Flag	Inhalt	Bedeutung
Bit 4	Ja=1, Nein=0	ausstehende Übermittlung des neuen öffentlichen Chiffrierschlüssels des Kunden bei Schlüsseländerung
Bit 5	Ja=1, Nein=0	ausstehende Übermittlung des neuen öffentlichen Signierschlüssels des Kunden bei Schlüsseländerung
Bit 6	Ja=1, Nein=0	Schlüsselsperre mit Erfolg durchgeführt (Information, da terminierte Sperrung erst in der Zukunft wirksam werden kann)
Bit 7	Ja=1, Nein=0	Leistungsprobleme bei Übermittlung neuer Schlüssel
Bit 8	0	RFU (reserved for future use)

3.2.2 Schlüsseleintrag

Feld	Format	Länge	Anzahl	Inhalt	Beschreibung
1	Byte	1	1	x'02'	RFU (reserved for future use)
2	Keytype	1	1		Schlüsseltyp
3	Int	2	1	2	Schlüsselnummer
4	Int	2	1		Schlüsselversion
5	Int	2	1		Länge öffentlicher Exponent
6	Byte	#	1	Little Endian	öffentlicher Exponent
7	Int	2	1		Länge öffentlicher Modulus
8	Byte	#	1	Little Endian	öffentlicher Modulus
9	Int	2	1		Länge des verschlüsselten geheimen Schlüssels
10	Byte	#	1	Little Endian	Geheimer Schlüssel (verschlüsselt) (siehe Abschnitt <i>Geheimer Schlüssel</i> , Seite 12)

Bemerkungen zur Belegung der Felder:

- Die Schlüsselnummer ist mit 2 festgelegt.
- Die Schlüsselversion wird beim Generieren neuer Schlüssel um eins hochgezählt.
- Der geheime Schlüssel wird gemäß Verschlüsselung (siehe Abschnitt *Verschlüsselung*, Seite 6) mit dem Schlüssel DK verschlüsselt.
- Die Länge des Feldes 6 wird durch den Inhalt von Feld 5 festgelegt.
- Die Länge des Feldes 8 wird durch den Inhalt von Feld 7 festgelegt.
- Die Länge des Feldes 10 wird durch den Inhalt von Feld 9 festgelegt.

Geheimer Schlüssel

Der geheime Schlüssel ist Bestandteil eines Schlüsseleintrages. Er wird verschlüsselt abgelegt. Zuvor hat er folgenden Aufbau:

Feld	Format	Länge	Anzahl	Inhalt	Beschreibung
1	Int	2	1		Länge des Modulus n
2	Byte	#	1	Little Endian	Modulus $n=p \cdot q$
3	Int	2	1		Länge der Primzahl p
4	Byte	#	1	Little Endian	Primzahl p

Feld	Format	Länge	Anzahl	Inhalt	Beschreibung
5	Int	2	1		Länge der Primzahl q
6	Byte	#	1	Little Endian	Primzahl q
7	Int	2	1		Länge von $d \bmod (p-1)$
8	Byte	#	1	Little Endian	$d \bmod (p-1)$
9	Int	2	1		Länge von $d \bmod (q-1)$
10	Byte	#	1	Little Endian	$d \bmod (q-1)$
11	Int	2	1		Länge von A_p
12	Byte	#	1	Little Endian	A_p
13	Int	2	1		Länge von A_q
14	Byte	#	1	Little Endian	A_q

Bemerkungen zur Belegung der Felder:

- Die Länge der Felder 2, 4, 6, 8, 10, 12 und 14 wird jeweils durch den Wert des Feldes mitgeteilt, das davor steht.
- A_p beinhaltet den Wert $q^{p-1} \bmod (n)$.
- A_q beinhaltet den Wert $n+1-A_p$.

3.3 Dateinhalt Feld 3: Öffentliche Schlüssel des Kreditinstitutes

Die öffentlichen Schlüssel der Kreditinstitute werden über Länderkennzeichen und Kreditinstitutscode referenziert.

Feld	Format	Länge	Anzahl	Inhalt	Beschreibung
1	Char	3	1		Länderkennzeichen
2	Char	30	1		Kreditinstitutscode
3	Byte	1	1	x'02'	RFU (reserved for future use)
4	Keytype	1	1		Schlüsseltyp
5	Int	2	1		Schlüsselversion
6	Int	2	1		Schlüsselnummer
7	Char	30	1		Schlüsselname
8	Int	2	1		Länge öffentlicher Modulus
9	Byte	#	1	Little Endian	Öffentlicher Modulus
10	Int	2	1		Länge öffentlicher Exponent

Feld	Format	Länge	Anzahl	Inhalt	Beschreibung
11	Byte	#	1	Little Endian	Öffentlicher Exponent

Bemerkungen zur Belegung der Felder:

- Die Schlüsselversion und Schlüsselnummer werden dem Kunden vom Kreditinstitut geliefert.
- Schlüsselname ist der Name des Schlüssels, der vom Kreditinstitut vergeben wurde.
- Die Länge des Feldes 9 wird durch den Inhalt des Feldes 8 festgelegt
- Die Länge des Feldes 11 wird durch den Inhalt des Feldes 10 festgelegt.

3.4 Dateiinhalte Feld 4: Datum

Feld	Format	Länge	Anzahl	Inhalt	Beschreibung
1	Char	14	1		Datum letzter Änderung/Erstellung

Bemerkung zur Belegung der Felder:

- Das Datum liegt im Format JJJJMMTTTHHMMSS vor.

3.5 Dateiinhalte Feld 5: MAC

Feld	Format	Länge	Anzahl	Inhalt	Beschreibung
1	Byte	20	1		MAC über den Disketteninhalt

Bemerkung zur Belegung der Felder:

- Der MAC wird gemäß MAC-Bildung (siehe Abschnitt *MAC-Bildung*, Seite 5) gerechnet.

Literaturverzeichnis

- [1] RSA Laboratories: PKCS #5 v2.0: Password-Based Cryptography Standard vom 25.3.1999
(<http://www.rsasecurity.com/rsalabs/pkcs/pkcs-5/>)

- [2] D. Balenson: RFC 1423 - Privacy Enhancement for Internet Electronic Mail: Part III: Algorithms, Modes, and Identifiers (Februar 1993)
(<http://www.faqs.org/rfcs/rfc1423.html>)

- [3] ZKA: FinTS 3.0 Spezifikation vom 22.4.2003
(http://www.hbci.de/siz_hbci.nsf/ZKAPages/Spezifikation30)