

DATADESIGN AG

**DATADESIGN HBCI BANKING APPLICATION
COMPONENTS**

**FORMAT DER RDH
SICHERHEITSDATEI**

Ansprechpartner: Dipl.-Inform. (FH) Andreas Selle
DataDesign AG
Fürstenrieder Str. 267
80687 München
Tel: 089 / 74119 – 0
Fax: 089 / 74119 – 299

Datum: 30. April 1999
Version: 0.94

Kapitel:	Version:	DataDesign HBCI Banking Application Components
1	0.94	Format der RDH Sicherheitsdatei
Seite:	Stand:	Kapitel: Format der RDH Sicherheitsdatei
2	30. April 1999	Abschnitt: Motivation und Zielsetzung

I. FORMAT DER RDH SICHERHEITSDATEI

I.1 Motivation und Zielsetzung	3
I.2 Sicherheitsmedium Diskette.....	4
I.3 Aufbau der RDH Sicherheitsdatei	5
I.3.1 Dateikennung.....	5
I.3.2 Bankverbindungsdaten	6
I.3.3 Schlüsseldaten	7
I.3.4 Kundensystem-ID und Signatur-ID.....	7
I.4 Verschlüsselung mit Hilfe einer Paßphrase	10
I.4.1 Anforderungen an die Paßphrase	10
I.4.2 Verschlüsselung des privaten Exponenten	10
I.5 Abläufe	11
I.5.1 Anlegen einer neuen Sicherheitsdatei.....	11
I.5.2 Ersteinreichung.....	11
I.5.3 Schlüsseländerung	11
I.5.4 Schlüsselsperrung	12

DataDesign HBCI Banking Application Components Format der RDH Sicherheitsdatei	Version: 0.94	Kapitel: I
Kapitel: Format der RDH Sicherheitsdatei Abschnitt: Motivation und Zielsetzung	Stand: 30. April 1999	Seite: 3

I.1 Motivation und Zielsetzung

Der HBCI Standard sieht für das RDH Sicherheitsverfahren eine Speicherung der Kundenschlüssel in einer Sicherheitsdatei auf einer Diskette als Sicherheitsmedium vor. In HBCI wird jedoch weder das Format der Sicherheitsdatei noch die zugehörigen kryptographischen Verfahrensweisen festgelegt.

Eine genaue öffentliche Spezifikation der RDH Sicherheitsdatei ist erforderlich um Interoperabilität der Kundensysteme im Sinne von HBCI zu gewährleisten. Es ist der Wunsch des Kunden Homebanking nicht nur durch die von seinem Kreditinstitut zur Verfügung gestellten Software, sondern auch durch Standardsoftware wie beispielsweise Intuit Quicken oder Microsoft Money, nutzen zu können. Durch eine exakte Spezifikation der RDH Sicherheitsdatei werden die Voraussetzungen dafür geschaffen.

Die vorliegende Spezifikation beschreibt die RDH Sicherheitsdatei der DataDesign HBCI Banking Application Components (DDBAC).

Die allgemeine Veröffentlichung der verwendeten Sicherheitsmechanismen beeinträchtigen ihre Wirksamkeit nicht. Es gibt einen kryptologischen Grundsatz der besagt, daß Sicherheit nur durch einen geheimen Schlüssel und nicht durch Geheimhaltung eines Verfahrens erreicht werden kann. Dies gilt nicht nur für die in HBCI spezifizierten Sicherheitsmechanismen, sondern natürlich auch für das hier beschriebene Verfahren.

Kapitel: I	Version: 0.94	DataDesign HBCI Banking Application Components Format der RDH Sicherheitsdatei
Seite: 4	Stand: 30. April 1999	Kapitel: Format der RDH Sicherheitsdatei Abschnitt: Sicherheitsmedium Diskette

I.2 Sicherheitsmedium Diskette

Wird das rein softwarebasierte asymmetrische Sicherheitsverfahren RDH eingesetzt, so kann als Sicherheitsmedium eine RDH Sicherheitsdatei auf Diskette bzw. Festplatte dienen.



Aus Sicherheitsgründen sollte die RDH Sicherheitsdatei immer auf Diskette und nicht auf der Festplatte gehalten werden. Das Kundenprodukt sollte dies geeignet nahelegen.

Dieses Dokument definiert lediglich das Format einer RDH Sicherheitsdatei. Es wird davon ausgegangen, daß der Inhalt dieser Datei vom Kundensystem als transparente Bytefolge interpretiert werden kann. Das physikalische Format des Datenträgers bleibt offen.

Das Sicherheitsmedium Diskette wird vom ZKA lediglich als Übergangslösung bis zu einer breiten Verfügbarkeit von wesentlich sichereren RSA Chipkarten verstanden. In einer späteren HBCI Version wird deshalb eine einheitliche Chipkarte für das RDH Verfahren definiert werden.

DataDesign HBCI Banking Application Components Format der RDH Sicherheitsdatei	Version: 0.94	Kapitel: I
Kapitel: Abschnitt: Format der RDH Sicherheitsdatei Aufbau der RDH Sicherheitsdatei	Stand: 30. April 1999	Seite: 5

I.3 Aufbau der RDH Sicherheitsdatei

Die RDH Sicherheitsdatei enthält mit Ausnahme der Kundensystem-ID und der Signatur-ID alle Daten die ein Kundensystem für die Abwicklung eines normalen HBCI Dialogs erforderlich sind.

Die RDH Sicherheitsdatei sowie alle in ihr enthaltenen Datensätze und deren Felder haben eine feste Länge. Die in HBCI übliche Segmentstruktur wird nicht verwendet. Dies ermöglicht den direkten lesenden und schreibenden Zugriff auf einzelne Datenfelder.

Alphanumerische Feldinhalte ('aa') werden grundsätzlich ASCII-kodiert, linksbündig eingestellt und mit NUL-Zeichen (X'00') auf die vorgegebene Länge aufgefüllt.

Binärdaten werden als 'xx' angegeben. Ihr Format wird im einzelnen angegeben.

Als Dateinamenserweiterung wird ".rdh" vereinbart.

Die RDH Sicherheitsdatei ist immer genau 2348 Bytes lang und enthält acht direkt aufeinanderfolgende Datensätze wie folgt:

Offset	Länge	Inhalt	Referenz
0	8	Dateikennung	I.3.1
8	1150	Bankverbindungsdaten	I.3.2
1158	200	Schlüsseldaten: Signierschlüssel des Kunden	I.3.3
1358	200	Schlüsseldaten: Chiffrierschlüssel des Kunden	I.3.3
1558	200	Schlüsseldaten: Alternativer Signierschlüssel des Kunden	I.3.3
1758	200	Schlüsseldaten: Alternativer Chiffrierschlüssel des Kunden	I.3.3
1958	200	Schlüsseldaten: Signierschlüssel des Kreditinstituts	I.3.3
2158	200	Schlüsseldaten: Chiffrierschlüssel des Kreditinstituts	I.3.3
	2358	Länge insgesamt	

I.3.1 Dateikennung

Die ersten acht Byte der RDH Sicherheitsdatei enthalten einen Datensatz mit einer eindeutigen Dateikennung und einer Versionsnummer.

Offset	Länge	Inhalt	Erläuterung	Referenz
0	5	'44 44 52 44 48'	Kennzeichen der RDH Sicherheitsdatei. (ASCII "DDRDH")	
5	3	'aa aa aa'	Version des Dateiformats.	
	8	Länge insgesamt		

Das Kundensystem muß beim Öffnen einer existierenden RDH Sicherheitsdatei das Kennzeichen auf Gültigkeit und Kompatibilität mit der angegebenen Version überprüfen.

Die Version des Dateiformats ist analog der HBCI Versionsnummer (vgl. HBCI 2.1 II.6.2 Nr. 3) formatiert. Die durch dieses Dokument definierte RDH Sicherheitsdatei

Kapitel:	Version:	DataDesign HBCI Banking Application Components
I	0.94	Format der RDH Sicherheitsdatei
Seite:	Stand:	Kapitel: Format der RDH Sicherheitsdatei
6	30. April 1999	Abschnitt: Aufbau der RDH Sicherheitsdatei

trägt die Versionsnummer 2.1. Das Feld Version des Dateiformats muß also "210" enthalten.

I.3.2 Bankverbindungsdaten

Der Datensatz "Bankverbindungsdaten" ist 1150 Bytes lang und enthält die folgenden Felder:

Offset	Länge	Inhalt	Erläuterung	Referenz
0	1	'xx'	Status der Bankverbindung	siehe unten
1	3	'aa aa aa'	Länderkennzeichen des kontoführenden Instituts, z.B. "280"	HBCI 2.1 Kap. II.5.3.2 Nr. 1
4	30	'aa .. aa'	Kreditinstitutscode (Bankleitzahl) des kontoführenden Instituts	HBCI 2.1 Kap. II.5.3.2 Nr. 2
34	30	'aa .. aa'	Benutzerkennung	HBCI 2.1 Kap. VI.5.1.1 Nr. 2
64	30	'aa .. aa'	Persönliche Kunden-ID	HBCI 2.1 Kap. III.3.1.2 Nr. 3
94	30	'aa .. aa'	Benutzerkennung des Kreditinstituts	HBCI 2.1 Kap. VI.5.1.1 Nr. 2
124	2	'aa aa'	Kommunikationsdienst	HBCI 2.1 Kap. VIII.7 b) Nr. 1
126	512	'aa .. aa'	Kommunikationsadresse	HBCI 2.1 Kap. VIII.7 b) Nr. 2
638	512	'aa aa'	Kommunikationsadressenzusatz	HBCI 2.1 Kap. VIII.7 b) Nr. 3
	1150	Länge insgesamt		

Der Status der Bankverbindung kann folgende Werte annehmen:

Wert	Bedeutung
'00'	Dies ist der Initialwert der beim Anlegen einer neuen RDH Sicherheitsdatei geschrieben wird.
'01'	Alle erforderlichen Bankverbindungsdaten und Schlüssel sind vorhanden. Das Kundensystem führt jetzt eine erstmalige Übermittlung der Kundenschlüssel wird durch. Erst nach einer erfolgreichen Übermittlung der Kundenschlüssel wird der Status auf '02' gesetzt. Siehe: I.5.2 Ersteinreichung.
'02'	Die RDH Sicherheitsdatei ist komplett und einsatzbereit.
'03'	Dieser Status wird geschrieben bevor vom Kundensystem eine Sperrung eingeleitet wird. Nach erfolgreicher Sperrung werden die Kundenschlüssel gelöscht und der Status auf '00' gesetzt. Siehe: I.5.4 Schlüsselsperrung.
'04'	Dieser Status wird geschrieben bevor vom Kundensystem eine Schlüsseländerung eingeleitet wird. Nach erfolgreicher Schlüsseländerung wird der Status wieder auf '02' gesetzt. Siehe: I.5.3 Schlüsseländerung.

Durch Auswerten des Status der Bankverbindung beim öffnen einer RDH Sicherheitsdatei kann ein Kundensystem feststellen ob die Daten in einem konsistenten Zustand sind, oder ob eine begonnene Operation abgebrochen wurde.

Die persönliche Kunden-ID wird für administrative HBCI Dialoge sowie als Standardvorgabe für normale HBCI Dialoge verwendet. Wird das Feld leer gelassen, so muß vom Kundensystem die Benutzerkennung eingesetzt werden.

DataDesign HBCI Banking Application Components Format der RDH Sicherheitsdatei	Version: 0.94	Kapitel: I
Kapitel: Format der RDH Sicherheitsdatei Abschnitt: Aufbau der RDH Sicherheitsdatei	Stand: 30. April 1999	Seite: 7

Die Benutzererkennung des Kreditinstituts ist erforderlich damit das Kundensystem den Schlüsselnamen für die öffentlichen Schlüssel des Kreditinstituts korrekt bilden kann. Diese Benutzererkennung normalerweise bei der erstmaligen Anforderung der Kreditinstitutsschlüssel ermittelt.

I.3.3 Schlüsseldaten

Die RDH Sicherheitsdatei enthält sechs Datensätze mit den Schlüsseldaten des Benutzers und des Kreditinstituts. Ein Schlüsseldatensatz ist 198 Bytes lang und ist wie folgt aufgebaut:

Offset	Länge	Inhalt	Erläuterung	Referenz
0	1	'xx'	Status des Schlüssels	siehe unten
1	1	'aa'	Schlüsselart: "S" (X'53') für Signierschlüssel, "V" (X'56') für Chiffrierschlüssel	HBCI 2.1 Kap. VI.5.1.1 Nr. 3
2	3	'aa aa aa'	Schlüsselnummer in ASCII, z. B. "1" (X'31 00 00')	HBCI 2.1 Kap. VI.5.1.1 Nr. 4
5	3	'aa aa aa'	Schlüsselversion in ASCII, z. B. "1" (X'31 00 00')	HBCI 2.1 Kap. VI.5.1.1 Nr. 5
8	96	'xx .. xx'	Öffentlicher Modulus in "big endian" Reihenfolge	siehe unten
104	96	'xx .. xx'	Verschlüsselter privater Exponent	siehe unten
	200	Länge insgesamt		

Änderungen an einem Schlüsseldatensatz sollen immer atomar vorgenommen werden, d.h. das Kundensystem muß immer den gesamten Datensatz in einer Operation schreiben.

Das Statusbyte kann folgende Werte annehmen:

Wert	Bedeutung
'00'	Der Schlüsseldatensatz ist unbenutzt. Der Inhalt aller anderen Felder des Datensatzes ist nicht definiert und sollte beim Schreiben mit binär '00' belegt werden.
'07'	Dieser Status muß vom Kundensystem geschrieben werden wenn ein neuer Schlüssel eingetragen wird. Die Felder Schlüsselnummer und Schlüsselversion müssen vom Kundensystem entsprechend belegt werden.
'10'	Der Schlüssel ist aktiv. Dieser Status darf nie von den alternativen Kundenschlüsseln angenommen werden.

Der öffentliche Modulus muß binär in "big endian" Reihenfolge (d.h. die höherwertigeren Bytes kommen vor den niederwertigeren Bytes) mit führenden Nullbytes eingetragen werden.


Die Speicherung des verschlüsselten privaten Exponents ist unter "I.4.2 Verschlüsselung des privaten Exponenten" definiert. Bei den Datensätzen für die öffentlichen Schlüssel des Kreditinstituts bleibt der private Exponent unbekannt. Das entsprechende Datenfeld muß mit binär null belegt werden.

I.3.4 Kundensystem-ID und Signatur-ID

Die im RDH Verfahren obligatorische Kundensystem-ID sowie die Signatur-ID werden nicht in der RDH Sicherheitsdatei gespeichert sondern müssen direkt auf

Kapitel: I	Version: 0.94	DataDesign HBCI Banking Application Components Format der RDH Sicherheitsdatei
Seite: 8	Stand: 30. April 1999	Kapitel: Format der RDH Sicherheitsdatei Abschnitt: Aufbau der RDH Sicherheitsdatei

dem Kundensystem, z. B. auf der Festplatte, abgelegt werden. Zum Verständnis hier ein paar Zitate aus HBCI 2.1:



HBCI 2.1 III.3.1.2 Nr. 4 Kundensystem-ID

Die Kundensystem-ID ist beim RDH-Verfahren erforderlich. Sie dient hier der eindeutigen Kennzeichnung des Kundensystems und sichert in Kombination mit der Signatur-ID die Validität (Eindeutigkeit) der Signatur. [...]

[...] Bevor ein Benutzer bei Verwendung des RDH-Verfahrens von einem neuen Kundensystem Aufträge erteilen kann, hat er im Wege einer Synchronisierung (Kap. III.8) eine Kundensystem-ID für dieses System anzufordern. Diese ID ist im folgenden stets anzugeben, wenn der Benutzer von diesem Kundensystem aus Nachrichten sendet. [...]

Da jedes Kreditinstitut die Kundensystem-ID unabhängig von anderen Kreditinstituten vergibt, muß das Kundenprodukt in der Lage sein, für jeden Kreditinstitutzugang eine eigene Kundensystem-ID zu verwalten.

HBCI 2.1 III.8 Synchronisierung

Eine Synchronisierung ist erforderlich, wenn für das vom Kunden verwendete Endgerät noch keine Kundensystem-ID vergeben wurde. [...] Im Rahmen der Dialoginitialisierungs-Antwortnachricht erhält das entsprechende Kundensystem eine neue Kundensystem-ID mitgeteilt.

HBCI 2.1 VI.4 Doppeleinreichungskontrolle (Verhinderung von Replay-Attacken)

Die Doppeleinreichungskontrolle wird mittels eines Zählers pro Signatur realisiert (Signatur-ID), dessen Inhalt jeweils in die Signatur(en) der Nachricht einfließt. Beim RDH-Verfahren wird zur Doppeleinreichungskontrolle z.Zt. zusätzlich zur Signatur-ID die Kundensystem-ID benötigt. [...]

Das heißt, die Kundensystem-ID ist ein eindeutiges Merkmal des Kundensystems und muß daher auf dem Kundensystem (Endgerät) gespeichert werden. Würde die Kundensystem-ID in der RDH Sicherheitsdatei gespeichert so hätte ein Benutzer immer dieselbe Kundensystem-ID unabhängig vom Kundensystem.

Die Signatur-ID wird vom Kundensystem erzeugt und bei jeder Signatur inkrementiert. Die Aufgabe der Signatur-ID ist es, die Eindeutigkeit einer Signatur zu gewährleisten. Damit das funktioniert sollte die Signatur-ID nicht durch den Endanwender manipuliert oder kopiert werden können. Das kann weitestgehend erreicht werden indem die Signatur-ID für den Anwender unsichtbar in einer versteckten Datei auf dem Kundensystem selbst abgelegt wird. Der Endanwender kann (und wird) eine Kopie der RDH Sicherheitsdatei anfertigen. Würde also die Signatur-ID in der RDH Sicherheitsdatei gespeichert so würde auch diese kopiert und somit nicht mehr eindeutig.

DataDesign HBCI Banking Application Components Format der RDH Sicherheitsdatei		Version: 0.94	Kapitel: I
Kapitel: Abschnitt:	Format der RDH Sicherheitsdatei Aufbau der RDH Sicherheitsdatei	Stand: 30. April 1999	Seite: 9

Nachdem die erste Signatur-ID vom Kundensystem gewählt wird (und ein Sicherheitsmedium auf verschiedenen Kundensystemen eingesetzt werden kann) besteht die Gefahr, daß verschiedene Kundensysteme dieselbe Signatur-ID wählen. Aus genau diesem Grund ist die Signatur-ID nur in Kombination mit der Kundensystem-ID (die ja das Kundensystem eindeutig identifiziert) wirklich eindeutig. Die Kundensystem-ID sollte also möglichst untrennbar zusammen (d.h. möglichst in nur einer Datei) mit der Signatur-ID auf dem Kundensystem gespeichert werden.

Wird ein Kundensystem erstmalig mit einer existierenden RDH Sicherheitsdatei konfrontiert so muß es für diesen Benutzer eine Kundensystem-ID durch eine Synchronisierung anfordern. Die vom HBCI System zugewiesene Kundensystem-ID muß dann zusammen mit einer neuen vom Kundensystem gewählten Signatur-ID auf dem Kundensystem gespeichert werden. Die gespeicherte Kundensystem-ID und Signatur-ID muß dem ursprünglichen Benutzer eindeutig zugeordnet werden können.

Online Systeme die keine Möglichkeit der lokalen Speicherung auf dem Kundensystem haben, z. B. JAVA Anwendung auf einem NC (Network Computer), müssen für jede Sitzung erneut eine Kundensystem-ID anfordern. Diese Kundensystem-ID steht dann bis zum Beenden der Anwendung zur Verfügung.

Kapitel: I	Version: 0.94	DataDesign HBCI Banking Application Components Format der RDH Sicherheitsdatei
Seite: 10	Stand: 30. April 1999	Kapitel: Format der RDH Sicherheitsdatei Abschnitt: Verschlüsselung mit Hilfe einer Paßphrase

I.4 Verschlüsselung mit Hilfe einer Paßphrase

I.4.1 Anforderungen an die Paßphrase

Eine Paßphrase ist eine Zeichenkette mit variabler Länge die vom Benutzer selbst gewählt werden kann. Paßphrasen sind geheim zu halten und der Benutzer ist auf seine Sorgfaltspflicht bei der erstmaligen Eingabe der Paßphrase explizit hinzuweisen.

Eine akzeptable Paßphrase sollte aus mindestens acht Zeichen bestehen. Um eine möglichst sichere Paßphrase zu erhalten sollte das Kundenprodukt auf die Eingabe einer Paßphrase mit Sonderzeichen und/oder Ziffern bestehen. Groß- und Kleinschreibung werden natürlich berücksichtigt.

Die maximale Länge der Paßphrase ist auf 255 Zeichen beschränkt. Der Zeichensatz der Paßphrase ist auf das gemeinsame Subset der Zeichencodes X'20' bis X'7E' ohne X'60' und X'7C' des ISO 8859-1 Zeichensatzes beschränkt (vgl. HBCI 2.1 Kap VIII.4).

I.4.2 Verschlüsselung des privaten Exponenten

Für die Verschlüsselung des privaten Exponenten wird das 2-Key-Triple-DES (2K3DES) Verfahren im CBC Modus eingesetzt. Ein Padding gemäß ISO 10126, wie bei der Verschlüsselung von HBCI Nachrichten, findet *nicht* statt. Der private Exponent wird statt dessen immer mit führenden Nullbits bis auf volle 768 Bit (96 Byte) Länge aufgefüllt. Die resultierende Byte-Länge ist durch acht teilbar und kann deshalb mit 2K3DES ohne zusätzliches Padding verarbeitet werden.

Der für die Verschlüsselung erforderliche 2K3DES Schlüssel wird aus der Paßphrase durch Bildung eines Hash-Wertes gewonnen. Als Hash-Funktion wird RIPEMD-160 wie in HBCI 2.1 VI.2.1.1 Abschnitt 1 beschrieben eingesetzt.

Aus dem 20 Byte Hash-Wert mit den Bytes $T_1 \dots T_{20}$ werden die 2K3DES Schlüssel S_1 und S_2 wie folgt gewonnen:

$$S_1 = T_3 \mid \dots \mid T_{10}$$

$$S_2 = T_{12} \mid \dots \mid T_{19}$$

T_i bezeichnet das Byte mit dem Index i des Hash-Wertes.

Die Verschlüsselung des privaten Exponenten (ohne Padding) wird mittels 2K3DES im CBC Modus gemäß ISO 10116 wie in HBCI 2.1 Kap VI.2.2 Abb. 13 gezeigt durchgeführt.

Das korrekte dechiffrieren des Exponenten und somit die Korrektheit der eingegebenen Paßphrase kann nur durch Testen des Ergebnisses in einer RSA Operation festgestellt werden. Beispiel: 96 Byte Testdaten werden mit Hilfe des öffentlichen Modulus in einer RSA Operation chiffriert und anschließend mit Hilfe des privaten Exponenten dechiffriert. Das Ergebnis wird mit den ursprünglichen Testdaten verglichen, ist es identisch kann daraus gefolgert werden, daß die Paßphrase korrekt eingegeben wurde.

DataDesign HBCI Banking Application Components Format der RDH Sicherheitsdatei	Version: 0.94	Kapitel: I
Kapitel: Format der RDH Sicherheitsdatei Abschnitt: Abläufe	Stand: 30. April 1999	Seite: 11

I.5 Abläufe

I.5.1 Anlegen einer neuen Sicherheitsdatei

Wird eine RDH Sicherheitsdatei neu angelegt, müssen alle Bytes nach dem Dateikopf mit binär null beschrieben werden. Dies wird vom Kundensystem als Leerdatei interpretiert. Dadurch wird implizit auch der Status der Bankverbindung als '00' beschrieben.

Im nächsten Schritt werden vom Kundensystem alle erforderlichen Bankverbindungsdaten und Schlüssel erfaßt und in den entsprechenden Feldern gespeichert.

Die öffentliche Schlüssel des Kreditinstituts können dabei entweder von einer ".pkd" Datei importiert oder online beim HBCI System angefordert werden. Bei einer online Anforderung werden die Schlüssel erst mit Status '07' auf die Karte geschrieben und der Kunde muß die Authentizität der Schlüssel manuell überprüfen. Erst nach Bestätigung durch den Kunden darf der Status auf '10' gesetzt werden.

I.5.2 Ersteinreichung

Vor einer Ersteinreichung sollte der Status der Bankverbindung auf '00' gesetzt sein.

Für die Ersteinreichung müssen zwei RSA Schlüsselpaare für den Kunden generiert und mit Status '07' auf der RDH Sicherheitsdatei abgelegt werden. Als Schlüsselnummer und Schlüsselversion soll "1" gewählt werden.

Die neuen Schlüssel werden in einem speziellen HBCI Dialog an das HBCI System übermittelt. Bevor dieser Dialog begonnen wird, muß der Status der Bankverbindung auf '01' gesetzt werden. Wird die Ersteinreichung erfolgreich abgeschlossen werden die Schlüsselstati auf '10' und der Status der Bankverbindung auf '02' gesetzt. Meldet das HBCI System einen Fehler kann die Ersteinreichung wiederholt werden.

Wird der HBCI Dialog abgebrochen, kann das Kundensystem nicht wissen ob die Ersteinreichung vom HBCI System bereits akzeptiert wurde oder nicht. Das Kundensystem erkennt diesen Zustand anhand des Status '01'. Um die Situation zu bereinigen versucht das Kundensystem zuerst einen signierten Synchronisierungsdialog. Ist dieser erfolgreich kann der Schlüsselstatus auf '10' gesetzt werden. Meldet das HBCI System den Code 9310 wurde die Ersteinreichung noch nicht akzeptiert und die Ersteinreichung kann wiederholt werden. Meldet das HBCI System den Code 9320, dann wurde die Ersteinreichung bereits akzeptiert und der Schlüsselstatus kann auf '10' gesetzt werden. Bei allen anderen Fehlercodes ist der Zustand weiterhin unklar und es muß angenommen werden daß die Ersteinreichung selbst fehlerhaft war.

I.5.3 Schlüsseländerung

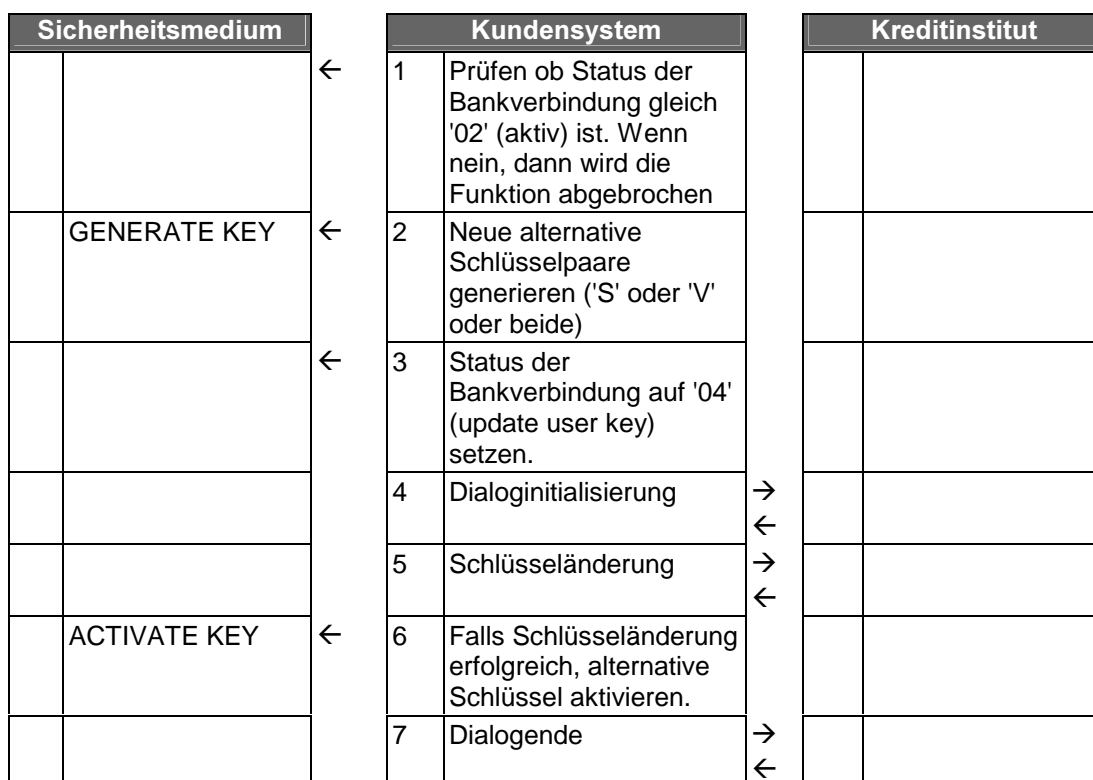
Werden die RSA Schlüsselpaare auf Diskette gespeichert, so empfiehlt sich eine Routinemäßige Schlüsseländerung durch den Kunden. Dabei erlaubt HBCI eine unabhängige Änderung des Signier- bzw. Chiffrierschlüsselpaares, d.h. der Kunde kann entweder ein Schlüsselpaar individuell oder beide Schlüsselpaare gleichzeitig ändern. Es wird empfohlen immer beide Schlüsselpaare gleichzeitig zu ändern.

Kapitel: I	Version: 0.94	DataDesign HBCI Banking Application Components Format der RDH Sicherheitsdatei
Seite: 12	Stand: 30. April 1999	Kapitel: Format der RDH Sicherheitsdatei Abschnitt: Abläufe

Voraussetzung für eine Schlüsseländerung ist, daß der Status der Bankverbindung den Wert '02' hat.

Für eine Schlüsseländerung müssen zuerst die gewünschten neuen Schlüsselpaare generiert werden. Dabei muß die Versionsnummer des neuen Schlüsselpaares höher als die Versionsnummer des aktiven Schlüsselpaares sein.

Das Kundensystem setzt den Status der Bankverbindung auf '04' und führt die online Schlüsseländerung durch. Ist diese erfolgreich werden die alternativen Schlüssel aktiviert, d.h. sie werden über die aktiven Benutzerschlüssel kopiert und ihr ursprünglicher Platz wird mit Nullbytes überschrieben. Der Status des Bankzugangs wird wieder auf '02' gesetzt.



Wird die Schlüsseländerung abgebrochen, so kann dies durch den Status der Bankverbindung '04' erkannt werden. Das Kundensystem kann darauf herausfinden welches Schlüsselpaar aktiv ist und die Schlüsseländerung gegebenenfalls wiederholen.

I.5.4 Schlüsselsperrung

Bei Verlust des Sicherheitsmediums oder wenn eine Kompromittierung der eigenen Schlüssel befürchtet wird kann ein Kunde seine Schlüssel und somit seinen Homebankingzugang selbst sperren. Die Schlüsselsperrung wird durch einen speziellen HBCI Dialog realisiert.

Bei Verlust des Sicherheitsmediums kann eine Sperrung nur anonym erfolgen. Dieser Fall wird hier nicht betrachtet.

Bei Verdacht auf Kompromittierung erfolgt die Schlüsselsperrung durch einen signierten HBCI Dialog. Im Verlauf dieses Dialoges werden die Schlüssel des

Kunden auf dem Sicherheitsmedium physikalisch gelöscht und der Status der Bankverbindung auf '00' zurückgesetzt.

Der Ablauf ein Schlüsselsperrung stellt sich wie folgt dar:

Sicherheitsmedium		Kundensystem		Kreditinstitut
Status der Bankverbindung	→	1 Prüfen ob Status der Bankverbindung gleich '02' ist. Wenn nein, dann wird die Funktion abgebrochen		
Status des Schlüssels	→	2 Prüfen ob beide Schlüsselpaare des Kunden aktiv sind.		
Status der Bankverbindung = '03'	←	3 Status der Bankverbindung auf '03' setzen.		
		4 Dialoginitialisierung	→ ←	
		5 Schlüsselsperrung	→ ←	
a) Schlüsselpaare werden gelöscht b) Status der Bankverbindung = '00'	←	6 Falls Schlüsselsperrung erfolgreich (Bestätigung durch HISSP Segment), dann werden beide Schlüsselpaare des Kunden gelöscht. Der Status der Bankverbindung wird auf '00' gesetzt.		
		7 Dialogende	→ ←	

Nach erfolgter Sperrung kann der Homebankingzugang nur durch eine erneute Erstinitialisierung mit Ini-Brief wieder freigeschalten werden. Dazu müssen auf dem Sicherheitsmedium zwei neue Schlüsselpaare generiert werden.