erschien in: Datenschutz und Datensicherung DuD 15/2 (Februar 1991) 71-75

Effizientere fail-stop Schlüsselerzeugung für das DC-Netz

Jörg Lukat, Andreas Pfitzmann, Michael Waidner

Institut für Rechnerentwurf und Fehlertoleranz, Universität Karlsruhe
Postfach 6980, D-7500 Karlsruhe 1, Bundesrepublik Deutschland
Telefon: ++49-721-608-4024, Fax: ++49-721-608-4290
E-mail (CSnet): WAIDNER@IRA.UKA.DE

Kurzfassung: In [Journal of Cryptology 1/1 (1988) 65-75] beschreibt DAVID CHAUM ein Protokoll zum anonymen Senden und Empfangen, das **DC-Netz**. Die Unbeobachtbarkeit des Sendens ist unbedingt, d.h. sie hängt von keinerlei kryptographischen Annahmen ab. Die Unbeobachtbarkeit des Empfangens setzt jedoch implizit ein zuverlässiges Verteilnetz voraus. Die Unbeobachtbarkeit von Senden *und* Empfangen wird *unbedingt*, wenn man die für das DC-Netz benötigten kryptographischen Schlüssel auf geeignete Weise erzeugt, nämlich durch **fail-stop Schlüsselerzeugung**.

Die von Michael Waidner in [Eurocrypt '89, Houthalen 1989, Session 7, 4. Vortrag] vorgeschlagenen Verfahren verwenden zur fail-stop Schlüsselerzeugung einfache arithmetische Ausdrücke über endlichen Körpern. Der Aufwand zur Schlüsselerzeugung wird näherungsweise halbiert, indem innerhalb dieser arithmetischen Ausdrücke jeweils zwei Multiplikationen und eine Additionen durch eine Multiplikation und zwei Additionen ersetzt werden, d.h. $ax+by$ durch $(a+b)(x+y)$. Es wird gezeigt, daß die Sicherheit hiervon unberührt bleibt.

Schlüsselwörter: Datenschutz, Kommunikationsnetze, Kryptographie, Sicherheit, Unbeobachtbarkeit, zuverlässige Verteilung.

1 Einleitung

Aus den in [PfPW_88] dargelegten Gründen erfordert überprüfbarer Datenschutz in Kommunikationsnetzen nicht nur den Schutz der *Kommunikationsinhalte*, sondern auch den Schutz der *Identitäten* von *Sender* und *Empfänger* (*Unbeobachtbarkeit*).

Betrachtet man nur den Schutz des *Senders* einer Nachricht, so ist das im folgenden betrachtete DC-Netz[1] (vgl. §1.1) sicher das wirkungsvollste: Es wirkt selbst dann, wenn ein Angreifer alle Leitungen des Netzes abhört und beliebig viele der Teilnehmer kontrolliert. Seine Sicherheit kann bewiesen werden, ohne daß angenommen werden muß, ein Angreifer könne etwas prinzipiell Berechenbares aus Aufwandsgründen nicht berechnen.

Will man auch den Empfänger schützen, so sieht man leicht (vgl. §1.2), daß das DC-Netz auf einem „sicheren" Verteilnetz aufsetzen muß. „Sicher" heißt hierbei, daß stets alle Teilnehmer vom Netz dasselbe empfangen müssen (*konsistente Verteilung*). Gelingt es einem Angreifer, verschiedenen Teilnehmern verschiedene Nachrichten zuzustellen, so kann dieser Teilnehmer identifizieren. In §1.3 werden die prinzipiellen Möglichkeiten für konsistente Verteilung skizziert. §1.4 enthält einige Anmerkungen hinsichtlich eines praktischen Einsatzes der hier vorgestellten Verfahren.

---

1 *Dining Cryptographers Network*, nach einem Beispiel in [Chau_88].

In §2 werden die aus [Waid_89, WaPf_89] bekannten Grundverfahren nochmals kurz beschrieben und so modifiziert, daß ihr Berechnungsaufwand näherungsweise halbiert wird. Die Sicherheit der modifizierten Verfahren wird bewiesen.

1.1 Überlagerndes Senden

In [Cha3_85, Chau_88] beschreibt DAVID CHAUM ein Protokoll, das **DC-Netz**, das den Teilnehmern eines Kommunikationsnetzes erlauben soll, Nachrichten unbeobachtbar zu senden und zu empfangen.

Die Grundlage des DC-Netzes ist **überlagerndes Senden**, das einen unbeobachtbaren additiven Kanal mit Mehrfachzugriff realisiert: Das Protokoll wird in **Runden** ausgeführt. Für jede Runde, etwa die r-te, wählt jeder Teilnehmer T_i, $i = 1, \ldots, n$, ein **Nachrichtenzeichen** M_i^r aus einem endlichen Körper[2] F [Lips_81]. Außerhalb des betrachteten Kommunikationsnetzes hat jeder Teilnehmer T_i mit jedem Teilnehmer T_j für jede Runde einen **Schlüssel** $K_{ij}^r \in F$ ($K_{ji}^r = K_{ij}^r$) vereinbart, der gleichverteilt aus F gewählt wurde.[3]

Aus M_i^r und allen Schlüsseln K_{ij}^r bestimmt T_i seine **lokale Ausgabe**

$$O_i^r := M_i^r + \sum_{j=1}^{i-1} K_{ij}^r - \sum_{j=i+1}^{n} K_{ij}^r$$

und verteilt diese an alle Teilnehmer. Jeder T_j berechnet hieraus die **globale Summe** $I_j^r := O_1^r + \ldots + O_n^r$. Erfolgte die Verteilung der O_i^r *zuverlässig*, so gilt natürlich $I_i^r = I_j^r$ für alle $i, j = 1, \ldots, n$.

Da in der globalen Summe jeder Schlüssel genau einmal addiert und einmal subtrahiert wird, ist diese gerade die Summe aller Nachrichtenzeichen M_i^r. Hat insbesondere genau ein Teilnehmer $M_i^r \neq 0$ gewählt, so ist $I_j^r = M_i^r$, d.h. dieses Nachrichtenzeichen wurde *kollisionsfrei* übertragen. Zur Vermeidung oder Auflösung von *Kollisionen* sind zahlreiche Zugriffsprotokolle bekannt [Cha3_85, Pfi1_85, Chau_88, BoBo_89, Pfit_90]. Für die hier betrachteten Verfahren ist die Art des Mehrfachzugriffs unerheblich, da sie alle auf „unterster" Ebene — der Berechnung der lokalen Ausgaben — aufsetzen.

Die Unbeobachtbarkeit des Sendens ist dadurch gegeben, daß ein Angreifer, der *beliebig viele* Teilnehmer kontrollieren und *alle* Leitungen abhören kann, die zwischen zwei *gutartigen* Teilnehmern ausgetauschten Schlüssel jedoch *nicht* kennt, aus den lokalen Ausgaben über die Nachrichtenzeichen der gutartigen Teilnehmer nichts erfährt außer ihrer Summe [Cha3_85, Chau_88, Pfit_90][4].

Werden, wie hier, über den Angreifer keine Annahmen gemacht (außer der, daß er nur Beobachtungen innerhalb des Kommunikationsnetzes machen kann), so nennt man die erreichte Sicherheit **unbedingt**.

Werden — wie in [Cha3_85, Chau_88] angenommen — die lokalen Ausgaben über ein *zuverlässiges Verteilnetz* verteilt, so ist auch die Unbeobachtbarkeit des Empfangens gesichert.

1.2 Aktive Angriffe auf die Anonymität

Verzichtet man auf diese Annahme eines zuverlässigen Verteilnetzes, so kann ein hinreichend mächtiger Angreifer durch Manipulation der Verteilung auch sendende Teilnehmer über ihre Reaktion auf von ihnen Empfangenes identifizieren [Waid_89, WaPf_89]:

---

2 Statt eines Körpers kann man auch eine ABELsche Gruppe verwenden [Pfi1_85]. Die folgenden Verfahren zur Schlüsselerzeugung gehen jedoch alle von einem endlichen Körper aus. Für den praktischen Einsatz ist dies jedoch kein Nachteil [Waid_89].

3 Ist Z eine Zufallsvariable, die unabhängig von „allen" anderen Zufallsvariablen in F gleichverteilt ist, so wird dies hier mit $Z \in_R F$ abgekürzt. Was mit „allen anderen Zufallsvariablen" gemeint ist, wird stets dem Kontext zu entnehmen sein.

4 Um diese Aussage zu erfüllen, müssen meist nicht alle Paare von Teilnehmern Schlüssel ausgetauscht haben [Chau_88]. Für das folgende ist dies aber bedeutungslos.

Angenommen, der Angreifer ist in der Lage, bei jedem Teilnehmer T_i anstelle von I_i^r ein beliebiges anderes $I^*_i{}^r$ zu erzeugen (z.B. in einem Sternnetz zur Verteilung, dessen Zentrale der Angreifer kontrolliert). Sei weiterhin angenommen, ein angreifender Teilnehmer T_a kommuniziere mit Teilnehmer T_x, welcher dem Angreifer unbekannt ist, nach den Regeln eines längeren Protokolls, etwa eines digitalen Zahlungssystems. T_a weiß, daß T_x auf eine Nachricht M innerhalb einer bestimmten Zeit antwortet, indem er eine Nachricht M' sendet. Der Angreifer sendet nun M ausschließlich an die Teilnehmer T_i mit $i < n/2$, die anderen Teilnehmer erhalten eine bedeutungslose Nachricht. Empfängt er die erwartete Antwort, so gilt $x < n/2$, ansonsten $x \geq n/2$. Sukzessive kann der Angreifer damit in $\log(n)$ Schritten den Empfänger T_x identifizieren.

1.3 Mögliche Abhilfen

Ist zuverlässige Verteilung nicht physisch gesichert, so wird sie üblicherweise durch Protokolle zur **Byzantinischen Übereinstimmung** (BÜ) hergestellt [PeSL_80]. Die bekannten BÜ-Protokolle setzen jedoch einen zahlenmäßig oder in seinen Berechnungsfähigkeiten beschränkten Angreifer voraus [Fisc_83, Reis_87, ChDw_89, BaPW_90]. Werden sie für das DC-Netz verwendet, so schützt es also nur noch gegen schwächere Angreifer, was nicht wünschenswert ist.

Ist man bereit, bei aktiven Angriffen vorübergehend auf die Diensterbringung zu verzichten (vgl. hierzu [WaPf_89, WaPf1_89, PfWa1_91]), so ist leicht einzusehen, daß die Verteilung eigentlich gar nicht zuverlässig erfolgen muß. Es genügt, die Nachrichtenübertragung zu stoppen, sobald zwei gutartige Teilnehmer in einer Runde r unterschiedliche globale Summen erhalten (**fail-stop Verteilung**, [Waid_89, WaPf_89]).

Durch Austauschen der globalen Summen nach jeder Runde können solche Inkonsistenzen leicht festgestellt — jedoch nicht toleriert! — werden [Waid_89, WaPf_89]. Entdeckt ein gutartiger Teilnehmer T_i eine Inkonsistenz oder einen gestörten Austausch [GiMS_74], so ist auch das Stoppen der Nachrichtenübertragung leicht: T_i stört einfach das überlagernde Senden in den nächsten Runden $r' > r$, indem er $M_i^{r'}$ zufällig aus F wählt. Damit ist die globale Summe der Runden r' unabhängig von den Nachrichtenzeichen.

Eine effizientere Realisierung von fail-stop Verteilung erhält man durch Kombination der beiden Aufgaben, Inkonsistenzen zu entdecken und überlagerndes Senden zu stoppen: Hängen die Schlüssel $K_{ij}^{r'}$ (bzw. $K_{ji}^{r'}$) so von den früher empfangenen globalen Summen I_i^r (bzw. I_j^r) ab, daß nach einer Runde r mit $I_i^r \neq I_j^r$ stets die folgenden Schlüssel $K_{ij}^{r'}$ und $K_{ji}^{r'}$ stochastisch unabhängig voneinander sind, so stoppt eine solche Inkonsistenz automatisch das überlagernde Senden (**fail-stop Schlüsselerzeugung**).

1.4 Anmerkungen zur Praxis

Auf die *Realisierbarkeit* eines DC-Netzes mit fail-stop Schlüsselerzeugung wird im folgenden nicht weiter eingegangen. Da die Schlüsselerzeugung keine zusätzliche Bandbreite benötigt, gilt hier aber dasselbe wie für ein reines DC-Netz [PfPW_88, Pfit_90]: Bei Verwendung von Glasfasern auch im Teilnehmeranschlußbereich könnte das DC-Netz ohne weiteres die Grundlage eines öffentlichen Fernmeldenetzes darstellen. Für schmalbandige Dienste (z.B. Fernsprechen) könnten jeweils alle Teilnehmer einer Ortsvermittlungsstelle ein DC-Netz bilden. Für breitbandige Dienste könnte man wohl nur einige hundert Teilnehmer in einem DC-Netz zusammenfassen.

Hinsichtlich der *Praktikabilität* sei darauf hingewiesen, daß die fail-stop Schlüsselerzeugung aktive Angriffe nur hinsichtlich ihrer Auswirkung auf die Unbeobachtbarkeit wirkungslos macht. Möchte man auch die Diensterbringung trotz aktiver Angriffe sicherstellen, so müssen aufwendigere Verfahren angewandt werden [WaPf_89, WaPf1_89, PfWa1_91].

2 Effizientere Schemata zur fail-stop Schlüsselerzeugung

Ein Verfahren zur **fail-stop Schlüsselerzeugung** des gemeinsamen Schlüssels von T_i und T_j muß die beiden folgenden Bedingungen erfüllen (vgl. §1.3):

ÜS *Überlagerndes Senden*: Wenn für alle Runden $r = 1, \ldots, r' - 1$ die Gleichung $I_i^r = I_j^r$ gilt, sind die Schlüssel $K_{ij}^{r'}$ und $K_{ji}^{r'}$ für Runde r' *gleich* und zufällig in F gewählt.

FS *Fail-stop*: Ist $I_i^r \neq I_j^r$ für eine Runde r, so sind für alle folgenden Runden r' die Schlüssel $K_{ij}^{r'}$ und $K_{ji}^{r'}$ *unabhängig* und zufällig in F gewählt.

Solange die Verteilung zuverlässig erfolgt, haben die globalen Summen die übliche Bedeutung, nämlich $I_i^{r'} = M_1^{r'} + \ldots + M_n^{r'}$. Nach der ersten Inkonsistenz gilt aber $K_{ij}^{r'} - K_{ji}^{r'} \in_R F$, womit auch die globale Summe zufällig in F verteilt ist und keine Information mehr über $M_1^{r'} + \ldots + M_n^{r'}$ enthält.

Im folgenden werden zwei Verfahren zur fail-stop Schlüsselerzeugung beschrieben, ein sehr effizientes, das die obige Definition aber nur im probabilistischen Sinne erfüllt (2.1), und ein weniger effizientes, das die Definition dafür aber deterministisch erfüllt (2.2).

Beide Schemata unterscheiden sich von den ursprünglichen aus [Waid_89, WaPf_89] durch die Anzahl der benötigten Körperadditionen und -multiplikationen. Erstere ist etwas erhöht, während letztere in etwa halbiert wird. Da in größeren Körpern der Aufwand der Multiplikation den der Addition um ein Vielfaches übersteigt [Lips_81], wird der Gesamtaufwand damit in etwa halbiert. Die Schemata werden ausführlicher in [Luka_91] beschrieben.

Zur Vereinfachung der Schreibweise sei für alle $i, j = 1, \ldots, n$, $r \in \mathbb{N}$

$$\delta_{ij}^r := K_{ij}^r - K_{ji}^r \text{ und } \varepsilon_{ij}^r := I_i^r - I_j^r.$$

Damit lauten die beiden obigen Bedingungen kurz:

$$\forall\, r' \in \mathbb{N}\, [\forall\, r \in \{1, \ldots, r'-1\}\, [\varepsilon_{ij}^r = 0] \Rightarrow [K_{ij}^{r'} \in_R F \wedge \delta_{ij}^{r'} = 0]] \qquad (\text{ÜS})$$

$$\forall\, r \in \mathbb{N}\, [\varepsilon_{ij}^r \neq 0 \Rightarrow\ \forall\, r' > r\, [K_{ij}^{r'} \in_R F \wedge \delta_{ij}^{r'} \in_R F]] \qquad (\text{FS})$$

Im Rest dieses Kapitels wird ein beliebiges, aber festes Schlüsselpaar (K_{ij}, K_{ji}) mit gutartigen T_i und T_j betrachtet. Es wird der mächtigste Angreifer unterstellt: Er kann die Werte von K_{ij}^r und K_{ji}^r für jede Runde r direkt beobachten (z.B. weil T_i und T_j meistens $M_i = 0 = M_j$ senden) und er kann T_i und T_j mit beliebigen globalen Summen I_i^r und I_j^r versorgen. Die Teilnehmer T_i und T_j werden als unsynchronisiert angenommen, so daß der Angreifer auf K_{ij}^{r+1} warten kann, bevor er T_j mit I_j^r versorgt (der in §1.2 skizzierte Angriff ist auch für ein perfekt synchronisiertes Netz erfolgreich).

2.1 Probabilistische fail-stop Schlüsselerzeugung

Das folgende Schema zur probabilistischen fail-stop Schlüsselerzeugung kommt im Vergleich mit dem in [Waid_89, WaPf_89] vorgestellten Schema, das zwei Additionen und zwei Multiplikationen benötigt, mit drei Additionen und nur einer Multiplikation aus.

Die Bedingung (FS) wird, wie angekündigt, von diesen Schlüsselerzeugungsschemata nur probabilistisch erfüllt. Sei hierzu für alle $d \in \mathbb{N}$:

$$\forall\, r \in \mathbb{N}\, \left[[\varepsilon_{ij}^r \neq 0 \wedge \forall\, s < r\, [\varepsilon_{ij}^s = 0]] \Rightarrow \forall\, r' \in \{r+1, \ldots, r+d\}\, [K_{ij}^{r'} \in_R F \wedge \delta_{ij}^{r'} \in_R F]\right] \qquad (\text{FS}_d)$$

Anschaulich heißt dies: Wenn in Runde r erstmals eine Inkonsistenz auftritt, so stören die beiden gutartigen Teilnehmer T_i und T_j für die nächsten d Runden das überlagernde Senden, indem sie ihre Schlüssel

unabhängig voneinander aus F wählen. Das maximale d, für welches (FS_d) erfüllt ist, ist eine Zufallsvariable mit Wahrscheinlichkeitsverteilung P(d)[5].

Schema 1: Seien $(a^1,a^2,...), (b^3,b^4,...) \in_R F^{\mathbb{N}}$, $c \in_R F$, und seien b^1, b^2, K^0, $I^0 := 0$. Dann sei für alle $r \in \mathbb{N}$

$$K^r := a^r + (c + K^{r-1})\ (b^r + I^{r-1}).$$

Die Teilnehmer T_i und T_j müssen die Werte a^r, b^r, c außerhalb des betrachteten Kommunikationsnetzes vertraulich austauschen. T_i (bzw. T_j) setzt für K^{r-1} bzw. I^{r-1} jeweils die subjektiven Werte K_{ij}^{r-1} und I_i^{r-1} (bzw. K_{ji}^{r-1} und I_j^{r-1}) ein.

Möchten die Teilnehmer c nicht für alle Zeiten unverändert lassen, etwa weil sie sich der Geheimhaltung von c nicht mehr sicher sind, können sie für c jederzeit einen neuen Wert wählen.

Satz 1 Schema 1 erfüllt die Bedingung (ÜS). Die maximale Anzahl d, für welche (FS_d) erfüllt ist, ist eine geometrisch verteilte Zufallsvariable:

$$P(d) = \frac{1}{|F|}\left(1 - \frac{1}{|F|}\right)^{d-1}$$

Werden maximal r_{max} Runden überlagernden Sendens ausgeführt, so ist die Wahrscheinlichkeit eines erfolgreichen Angriffs

$$P_A \leq \frac{r_{max}}{|F|}.$$ ♦

Beweis von Satz 1: Aus $(a^1,a^2,...) \in_R F^{\mathbb{N}}$ folgt sofort $(K_{ij}^1,K_{ij}^2,...) \in_R F^{\mathbb{N}}$, denn allgemein gilt: ist $x \in F$ beliebig, $y \in_R F$, so ist auch $(x + y) \in_R F$ [Sha1_49]. Gilt für ein $r \in \mathbb{N}$ und alle $r' < r$ jeweils $\varepsilon_{ij}^{r'} = 0$, so gilt offensichtlich auch für alle $r' < r$ jeweils $K_{ij}^{r'} = K_{ji}^{r'}$ und damit auch $K_{ij}^r = K_{ji}^r$. Also ist (ÜS) erfüllt.

Die Verteilungen von $(K_{ij}^1,K_{ij}^2,...)$ und $(\delta^1, \delta^2, \ldots)$ sind unabhängig voneinander, da erstere ausschließlich durch $(a^1,a^2,...) \in_R F^{\mathbb{N}}$ bestimmt wurde und letztere von $(a^1,a^2,...)$ überhaupt nicht abhängt.

Sei s die erste Runde, in der gestört wird, d.h. es ist erstmals $\varepsilon_{ij}^s \neq 0$. Dann gilt für Runde s+1:

$$\delta_{ij}^{s+1} = c\ \varepsilon_{ij}^s + b^{s+1}\ \delta_{ij}^s + K_{ij}^s\ I_i^s - K_{ji}^s\ I_j^s = (c + K_{ij}^s)\ \varepsilon_{ij}^s$$

denn nach (ÜS) gilt ja $K_{ij}^s = K_{ji}^s$. Wegen $c \in_R F$ ist auch $(c + K_{ij}^s) \in_R F$, nach Voraussetzung ist $\varepsilon_{ij}^s \neq 0$. Es folgt $(c + K_{ij}^s)\ \varepsilon_{ij}^s \in_R F$, denn allgemein gilt: ist $x \in F\backslash\{0\}$ beliebig, $y \in_R F$, so ist auch $(x \bullet y) \in_R F$ [CaWe_79].

Für $u > 0$ beliebig betrachte man die Runde s+u+1. Gilt $\delta_{ij}^{s+u} \neq 0$, so ist

$$\delta_{ij}^{s+u+1} = b^{s+u+1}\ \delta_{ij}^{s+u} + (c\ \varepsilon_{ij}^{s+u} + K_{ij}^{s+u}\ I_i^{s+u} - K_{ji}^{s+u}\ I_j^{s+u})$$

Wegen $b^{s+u+1} \in_R F$ und $\delta_{ij}^{s+u} \neq 0$, ist $b^{s+u+1}\ \delta_{ij}^{s+u} \in_R F$. Der eingeklammerte Term ist unabhängig von b^{s+u+1}, so daß insgesamt folgt: $\delta_{ij}^{s+u+1} \in_R F$.

Ist $\delta_{ij}^{s+u} = 0$, so ist δ_{ij}^{s+u+1} sicher nicht mehr zufällig in F verteilt: gilt z.B. $\varepsilon^{s+u} = 0$, so folgt deterministisch $\delta_{ij}^{s+u+1} = 0$.

d ist damit gegeben durch den ersten Wert $d \geq 1$ für den gilt $\delta_{ij}^{s+d} = 0$. Aus dem bisher gezeigten folgt:

$$P(\delta_{ij}^{s+1} \neq 0) = 1 - \frac{1}{|F|} \text{ und } P(\delta_{ij}^{s+u+1} \neq 0 \mid \delta_{ij}^{s+u} \neq 0) = 1 - \frac{1}{|F|} \text{ für } u = 1, 2, \ldots.$$

---

[5] Die Verteilung von d ist im allgemeinen definiert über alle zufälligen Entscheidungen der gutartigen Teilnehmer — hier also die Gleichverteilung von Schlüsseln aus F — und das Verhalten des Angreifers. Letzteres hat für die folgenden Verfahren keine Bedeutung.

Daraus folgt

$$\mathrm{P}(d) = \frac{1}{|F|}\,(1-\frac{1}{|F|})^{d-1}.$$

Die Wahrscheinlichkeit P_A eines erfolgreichen Angriffs ist die Wahrscheinlichkeit für $s + d \le r_{max}$:

$$P_A = \mathrm{P}(d \le r_{max} - s) \le \mathrm{P}(d \le r_{max}) = 1 - \mathrm{P}(d > r_{max}) = 1 - (1 - \frac{1}{|F|})^{r_{max}}.$$

Durch die einfach zu verifizierende Abschätzung $(1-x)^k \ge 1 - k\,x$ erhält man $P_A \le \frac{r_{max}}{|F|}$. □

Vergleicht man Schema 1 mit dem entsprechenden aus [Waid_89, WaPf_89], so sieht man im Nachhinein recht schnell, warum die neue Lösung das Gleiche bezüglich (ÜS) und (FS_d) leistet:

Schema aus [Waid_89, WaPf_89]: $K^r := a^r + b^r\,K^{r-1} + c\,I^{r-1}$

neues Schema:
$$K^r := a^r + (c + K^{r-1})\,(b^r + I^{r-1})$$
$$= a^r + b^r\,K^{r-1} + c\,I^{r-1} + b^r\,c + K^{r-1}\,I^{r-1}$$

Das neue Schema gleicht dem alten, multipliziert man die Klammern aus, bis auf die beiden Terme $b^r\,c$ und $K^{r-1}\,I^{r-1}$. Diese haben auf die Sicherheit des Verfahrens (Bedingungen (ÜS) und (FS)) keinen Einfluß.

Im Vergleich zu einfachem überlagernden Senden (vgl. §1.1) benötigt Schema 1 fast doppelt soviel Schlüsselzeichen (für r Runden $2r-1$ statt r).

2.2 Deterministische fail-stop Schlüsselerzeugung

Die in 2.1 gefundene Möglichkeit, die Summe zweier Produkte ($b^t\,K^{t-1} + c\,I^{t-1}$) durch das Produkt zweier Summen ($(c + K^{t-1})\,(b^t + I^{t-1})$) zu ersetzten, läßt sich auch auf die deterministische fail-stop Schlüsselerzeugung aus [Waid_89, WaPf_89] übertragen: Dort war

$$K^r := a^r + \sum_{k=1}^{r-1} b^{r-k}\,I^k$$

für $(a^1,a^2,...), (b^1,b^2,...) \in_R F^{\mathbb{N}}$. Hieraus gewinnt man:

Schema 2: Seien $(a^1,a^2,...), (b^1,b^2,...)$ zufällige Folgen aus F. Dann sei für alle $r \in \mathbb{N}$

für r ungerade: $$K^r := a^r + \sum_{k=1}^{(r-1)/2} (b^{r-(2k-1)}+I^{2k})\,(I^{2k-1}+b^{r-2k}),$$

für r gerade: $$K^r := a^r + \sum_{k=1}^{(r-2)/2} (b^{r-(2k-1)}+I^{2k})\,(I^{2k-1}+b^{r-2k}) + b^1\,I^{r-1}.$$

Satz 2 Schema 2 erfüllt die Bedingungen (ÜS) und (FS). ♦

Beweis von Satz 2: Wie in Satz 1 folgt aus $(a^1,a^2,...) \in_R F^{\mathbb{N}}$ sofort $(K_{ij}^1,K_{ij}^2,...) \in_R F^{\mathbb{N}}$ und damit die Unabhängigkeit von $(K_{ij}^1,K_{ij}^2,...)$ und $(\delta^1, \delta^2, \ldots)$.

Ist $\varepsilon_{ij}^r = 0$ für alle $r < r'$, so ist offensichtlich $\delta_{ij}^{r'} = 0$ und (ÜS) erfüllt.

Sei s die erste Runde mit $\varepsilon_{ij}^s \ne 0$ und sei $r > s$ beliebig. Der Einfachheit halber sei $\varepsilon^u := \varepsilon_{ij}^u$ und $\delta^u := \delta_{ij}^u$ für $u = s+1, \ldots, r$. Die Differenzen δ^u werden nach dem folgenden System linearer Gleichungen gebildet:

$$\delta^u = 0 \text{ für } u = 1, \ldots, s$$

$$\begin{pmatrix} \delta^{s+1} \\ \delta^{s+2} \\ \ldots \\ \delta^{r-1} \\ \delta^{r} \end{pmatrix} = \begin{pmatrix} \varepsilon^s & 0 & \ldots & 0 & 0 \\ \varepsilon^{s+1} & \varepsilon^s & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \varepsilon^{r-2} & \varepsilon^{r-3} & \ldots & \varepsilon^s & 0 \\ \varepsilon^{r-1} & \varepsilon^{r-2} & \ldots & \varepsilon^{s+1} & \varepsilon^s \end{pmatrix} \bullet \begin{pmatrix} b^1 \\ b^2 \\ \ldots \\ b^{r-s-1} \\ b^{r-s} \end{pmatrix} + \begin{pmatrix} \Delta^{s+1} \\ \Delta^{s+2} \\ \ldots \\ \Delta^{r-1} \\ \Delta^{r} \end{pmatrix}$$

wobei für $u = s+1, \ldots, r$ gilt

für ungerade u: $$\Delta^u = \sum_{k=s+1}^{(u-1)/2} I_{ij}{}^{2k}\, I_{ij}{}^{2k-1} - I_{ji}{}^{2k}\, I_{ji}{}^{2k-1},$$

für gerade u: $$\Delta^u = \sum_{k=s+1}^{(u-2)/2} I_{ij}{}^{2k}\, I_{ij}{}^{2k-1} - I_{ji}{}^{2k}\, I_{ji}{}^{2k-1}.$$

Wegen $\varepsilon^s \neq 0$ ist die Matrix regulär und definiert eine bijektive Abbildung. Da $(b^1,..., b^{r-s}) \in_R F^{r-s}$ und unabhängig von $(\Delta^{s+1},..., \Delta^r)$ ist, folgt $(\delta^{s+1},...,\delta^r) \in_R F^{r-s}$. □

Schema 2 benötigt ebenso viele Schlüsselzeichen wie Schema 1 und zusätzlich genügend Speicherplatz, um alle früher empfangenen globalen Summen speichern zu können. Im Unterschied zum entsprechenden Schema aus [Waid_89, WaPf_89] benötigt Schema 2 zur Berechnung von K^r jedoch statt $r-1$ Multiplikationen und Additionen nur $\lfloor r / 2 \rfloor$ Multiplikationen und im schlechteren Fall $3\lfloor r / 2 \rfloor$ Additionen.

Dank: Wir danken Prof. Dr. *Joachim Biskup* und *Dirk Fox* für hilfreiche Anregungen, *Manfred Böttger* und *Birgit Pfitzmann* für ihre Unterstützung bei der Implementierung von Schema 1 und der Einbindung in eine Versuchsumgebung für kryptographische Protokolle, und der *DFG* für ihre finanzielle Unterstützung.

Literatur

BaPW_90 Birgit Baum-Waidner, Birgit Pfitzmann, Michael Waidner: Unconditional Byzantine Agreement with Good Majority; Symposium on Theoretical Aspects of Computer Science (STACS), Hamburg, Feb. 1991; will be published in LNCS, Springer-Verlag, Berlin 1991.

BoBo_89 Jurjen Bos, Bert den Boer: Detection of Disrupters in the DC Protocol; Eurocrypt '89, LNCS 434, Springer-Verlag, Berlin 1990, 320-327.

CaWe_79 J. Lawrence Carter, Mark N. Wegman: Universal Classes of Hash Functions; Journal of Computer and System Sciences 18 (1979) 143-154.

Cha3_85 David Chaum: The Dining Cryptographers Problem. Unconditional Sender Anonymity; Amsterdam 1985.

Chau_88 David Chaum: The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability; Journal of Cryptology 1/1 (1988) 65-75.

ChDw_89 Benny Chor, Cynthia Dwork: Randomization in Byzantine Agreement; JAI Press, Advances in Computing Research Vol. 5, Greenwich (Connecticut) 1989, 443-497.

Fisc_83 Michael J. Fischer: The Consensus Problem in Unreliable Distributed Systems (A Brief Survey); 4th Conference on Foundations of Computation Theory, 1983, 127–140.

GiMS_74 E. N. Gilbert, F. J. Mac Williams, N. J. A. Sloane: Codes which detect deception; Bell System Technical Journal 53/3 (1974) 405-424.

Lips_81 John D. Lipson: Elements of algebra and algebraic computing; Addison-Wesley; Reading 1981.

Luka_91 Jörg Lukat: Analyse, Verbesserung und Implementierung einiger Verfahren zur Tolerierung aktiver Angriffe im DC-Netz; Diplomarbeit am Institut für Rechnerentwurf und Fehlertoleranz der Universität Karlsruhe, Januar 1991.

PeSL_80 Marshall Pease, Robert Shostak, Leslie Lamport: Reaching Agreement in the Presence of Faults; Journal of the ACM 27/2 (1980) 228-234.

Pfi1_85 Andreas Pfitzmann: How to implement ISDNs without user observability - Some remarks; Fakultät für Informatik, Universität Karlsruhe, Interner Bericht 14/85.

Pfit_90 Andreas Pfitzmann: Diensteintegrierende Kommunikationsnetze mit teilnehmerüberprüfbarem Datenschutz; IFB 234, Springer-Verlag, Heidelberg 1990.

PfPW_88 Andreas Pfitzmann, Birgit Pfitzmann, Michael Waidner: Datenschutz garantierende offene Kommunikationsnetze; Informatik-Spektrum 11/3 (1988) 118-142.

PfWa1_91 Birgit Pfitzmann, Michael Waidner: Unbedingte Unbeobachtbarkeit mit kryptographischer Robustheit; erscheint in: Proc. Verläßliche Informationssysteme (VIS'91), März 1991, Darmstadt, Informatik-Fachberichte, Springer-Verlag, Heidelberg 1991.

Reis_87 Rüdiger Reischuk: Konsistenz und Fehlertoleranz in Verteilten Systemen - Das Problem der Byzantinischen Generäle; 17. GI Jahrestagung, IFB 156, Springer-Verlag, Berlin 1987, 65-81.

Sha1_49 C. E. Shannon: Communication Theory of Secrecy Systems; Bell System Technical Journal 28/4 (1949) 656-715.

Waid_89 Michael Waidner: Unconditional Sender and Recipient Untraceability in spite of Active Attacks; Eurocrypt '89, LNCS 434, Springer-Verlag, Berlin 1990, 302-319.

WaPf_89 Michael Waidner, Birgit Pfitzmann: Unconditional Sender and Recipient Untraceability in spite of Active Attacks – Some Remarks; Fakultät für Informatik, Universität Karlsruhe, Interner Bericht 5/89, März 1989.

WaPf1_89 Michael Waidner, Birgit Pfitzmann: The Dining Cryptographers in the Disco: Unconditional Sender and Recipient Untraceability with Computationally Secure Serviceability; Universität Karlsruhe 1989, präsentiert auf der Rump-Session der Eurocrypt'89.