

2. Eine kurze Einführung in Chiffriersysteme

2.1. Definition eines Chiffriersystems

Als Erstes sollte präzisiert werden, was genau unter einer Folge von Elementen einer Menge M zu verstehen ist, denn solche Folgen sind Basisobjekte von Chiffriersystemen.

Eine Folge von Elementen einer beliebigen Menge M ist eine Abbildung $f: N \rightarrow M$, wobei $N \subset \mathbb{N}$. Statt $f(n)$ für $n \in N$ wird gewöhnlich f_n geschrieben, oder die Folge wird direkt mit ihren Folgengliedern als $(f_n)_{n \in N}$ bezeichnet.

Die Menge aller Folgen von Elementen einer Menge M wird mit $\mathcal{F}_M$ bezeichnet, d.h.

$$\mathcal{F}_M = \{ (f_n)_{n \in N} \mid N \subset \mathbb{N} \wedge \bigwedge_{n \in N} f_n \in M \}$$

Die Menge N ist die *Indexmenge*, ihre Elemente sind die *Indizes* der Folge.

Bei einer endlichen Indexmenge $N = \{n_1, \dots, n_k\}$ kann eine Folge auch so bezeichnet werden, daß man die Folgenglieder nebeneinander in einer Reihe anschreibt, zur besseren Abgrenzung oft in Klammern gesetzt,

$$(f_{n_1}, \dots, f_{n_k})$$

speziell bei $N = \{a, b, c, d\}$ also als (f_a, f_b, f_c, f_d) . Spielt die Indexmenge keine Rolle, kommt es also nur auf die Reihenfolge der Folgenglieder an, können die Indizes auch unterdrückt werden, etwa in (o, p, q, r) . Das ist also so zu verstehen, daß es eine Indexmenge $N = \{n_1, n_2, n_3, n_4\} \subset \mathbb{N}$ und eine Abbildung $f: N \rightarrow \{o, p, q, r\}$ gibt mit $f_{n_1} = o$, $f_{n_2} = p$, $f_{n_3} = q$ und $f_{n_4} = r$. Man läßt oftmals sogar die Kommata weg, wie in $(o\ p\ q\ r)$.

In der Kryptographie betrachtet man gerne Folgen von Elementen von $\mathbb{Z}_{26}$, dabei werden die $n \in \mathbb{Z}_{26}$ mit den Großbuchstaben des Alphabetes bezeichnet, meistens A = 0, B = 1 und so fort bis Z = 25. Texte in Grobkuchstaben können dann als Folgen von Elementen aus $\mathbb{Z}_{26}$ aufgefasst werden. Der besseren Lesbarkeit wegen empfiehlt es sich jedoch, zu $\mathbb{Z}_{27}$ überzugehen und mit dem hinzugekommenen Element das Leerzeichen zu benennen, also $\sqcup = 26$. So wird beispielsweise der Text (die Zeichenfolge)

EIN $\sqcup$ KLARTEXT

zu der folgenden Folge von Elementen von $\mathbb{Z}_{27}$:

4 8 13 26 10 11 0 17 19 4 23 19

Diese Zahlenfolge repräsentiert eine unendliche Anzahl von endlichen Folgen der Länge 12 von Elementen von $\mathbb{Z}_{27}$. Eine dieser Folgen ist $f: \{2, 3, \dots, 12, 13\} \rightarrow \mathbb{Z}_{27}$ mit $f_2 = 4$, $f_3 = 8$ usw. bis $f_{13} = 19$.

Heutzutage werden allerdings weniger alberne Botschaften wie ANGRIFF $\sqcup$ IM $\sqcup$ MORGENGRAUEN verschlüsselt sondern unter Anderem auch ganze Computerdateien beliebigen Inhalts. Man hat es

2. Eine kurze Einführung in Chiffriersysteme

dann mit Folgen der kleinsten Einheit einer solchen Datei zu tun, also etwa mit Bytes oder 16-Bit-Worten, die als Elemente von $\mathbb{Z}_2^8$ bzw. $\mathbb{Z}_2^{16}$ interpretiert werden können.

Moderne Verschlüsselungsverfahren arbeiten nicht mehr mit Folgen von als Zahlen interpretierten Zeichen (Buchstaben, Ziffern, Satzzeichen), sondern mit mathematischen Objekten. Dazu zählt natürlich AES, das mit 4×4 -Matrizen mit Koeffizienten im endlichen Körper $\mathbb{K}_{2^8}$ arbeitet, andere Systeme arbeiten mit sehr großen natürlichen Zahlen. Diese systeminternen Klartexte und Geheimentexte werden zum praktischen Gebrauch in externe Texte verwandelt oder aus ihnen rückverwandelt.

Die Menge $\mathcal{F}_M$ aller Folgen mit Elementen in M ist für praktische Zwecke viel zu groß, man beschränkt sich deshalb auf die Menge aller *endlichen* Folgen:

$$\mathcal{E}_M = \{ \mathbf{f}: N \longrightarrow M \mid N \subset \mathbb{N} \wedge \#(N) < \#(\mathbb{N}) \}$$

Es ist natürlich $\mathcal{E}_M \subset \mathcal{F}_M$. Schließlich sei noch für beliebige Mengen A und B

$$\mathbf{I}\langle A, B \rangle$$

die Menge aller injektiven Abbildungen $\varphi: A \longrightarrow B$, also solcher Abbildungen mit der Eigenschaft, daß $a = \tilde{a}$ aus $\varphi(a) = \varphi(\tilde{a})$ folgt.

Mit diesen Bezeichnungen kann nun ein Chiffriersystem präzise beschrieben werden. Mit der nachfolgenden Definition werden praktisch alle existierenden Chiffriersysteme erfasst. Und zwar besteht ein Chiffriersystem aus

- (i) einem Klartextalphabet $\mathbf{T}$
- (ii) einem Klartextbereich $\mathcal{T} \subset \mathcal{E}_{\mathbf{T}}$
- (iii) einem Geheimentextalphabet $\mathbf{G}$
- (iv) einem Geheimentextbereich $\mathcal{G} \subset \mathcal{E}_{\mathbf{G}}$
- (v) einem Schlüsselbereich $\mathcal{K}$
- (vi) einer Abbildung $\Omega: \mathcal{K} \longrightarrow \mathbf{I}\langle \mathcal{T}, \mathcal{G} \rangle$

Jedem Schlüssel $\mathbf{k} \in \mathcal{K}$ ist also eine injektive Abbildung $\Omega(\mathbf{k})$ vom Klartextbereich $\mathcal{T}$ in den Geheimentextbereich $\mathcal{G}$ zugeordnet, nämlich die **Verschlüsselungsabbildung** oder **Chiffrierabbildung** des Schlüssels $\mathbf{k}$.

Die Umkehrfunktion von $\Omega(\mathbf{k})$, also die Abbildung $\Omega(\mathbf{k})^{-1}: \Omega(\mathbf{k})[\mathcal{T}] \longrightarrow \mathcal{T}$, ist die **Entschlüsselungsabbildung** oder **Dechiffrierungsabbildung**.

Diese auf den ersten Blick möglicherweise etwas furchteinflößende Definition ist tatsächlich leicht zu verstehen, wie das folgende einfache Beispiel zeigt.

- (i) $\mathbf{T} = \mathbb{Z}_{27}$
- (ii) $\mathcal{T} = \mathcal{E}_{\mathbb{Z}_{27}}$
- (iii) $\mathbf{G} = \mathbb{Z}_{27}$
- (iv) $\mathcal{G} = \mathcal{E}_{\mathbb{Z}_{27}}$
- (v) $\mathcal{K} = \mathbb{Z}_{27}$
- (vi) Die Abbildung $\Omega: \mathbb{Z}_{27} \longrightarrow \mathbf{I}\langle \mathcal{E}_{\mathbb{Z}_{27}}, \mathcal{E}_{\mathbb{Z}_{27}} \rangle$ ist gegeben als $\Omega(k)((t_\nu)_{\nu \in N}) = (t_\nu \oplus k)_{\nu \in N}$, wobei $N \subset \mathbb{N}$ endlich und $\mathbf{t}: N \longrightarrow \mathbb{Z}_{27}$ eine Folge von Elementen aus $\mathbb{Z}_{27}$ ist. Ferner ist $\oplus$ die Addition von $\mathbb{Z}_{27}$.

Das Verfahren ersetzt jedes Zeichen t_ν einer Folge $(t_\nu)_{\nu \in N}$ durch das Zeichen $t_\nu \oplus k$, oder um es mit einem Fremdwort auszudrücken, $t_\nu \oplus k$ *substituiert* t_ν . Dieses Chiffriersystem heißt deshalb

ein Chiffriersystem mit Substitutionsschlüssel.

Ganz konkret im Falle $k = 1$ wird also folgendermaßen substituiert: Jeder Buchstabe von A bis Y wird durch den rechten Nachbarbuchstaben im Alphabet ersetzt, d.h. A durch B, B durch C usw. bis schließlich Y durch Z. Weiter wird Z durch das Leerzeichen $\sqcup$ und endlich das Leerzeichen selbst durch den Buchstaben A ersetzt. Dieses Verschlüsselungssystem ist sehr alt, schon Julius Caesar soll es mit $k = 3$ eingesetzt haben.

Es ist allerdings noch nicht gezeigt worden, daß das beschriebene System tatsächlich ein Chiffriersystem ist. Dazu ist zu bestätigen, daß für jedes $k \in \mathbb{Z}_{27}$ die Abbildung $\Omega(k)$ injektiv ist. Es seien also $\mathbf{s} = (\mathbf{s}_\mu)_{\mu \in M}$ und $\mathbf{t} = (\mathbf{t}_\nu)_{\nu \in N}$ Folgen mit

$$(\mathbf{s}_\nu \oplus k)_{\nu \in M} = \Omega(k)((\mathbf{s}_\mu)_{\mu \in M}) = \Omega(k)((\mathbf{t}_\nu)_{\nu \in N}) = (\mathbf{t}_\nu \oplus k)_{\nu \in N}$$

Zwei Abbildungen, d.h. hier die beiden Folgen $\Omega(k)(\mathbf{s}): M \rightarrow \mathbb{Z}_{27}$ und $\Omega(k)(\mathbf{t}): N \rightarrow \mathbb{Z}_{27}$, können nur dann gleich sein, wenn sie denselben Definitionsbereich besitzen. Aus obiger Annahme folgt deshalb $M = N$. Weiter sind zwei Abbildungen (mit demselben Definitionsbereich) genau dann gleich, wenn ihre Bilder gleich sind. Aus der Annahme folgt also weiter $\mathbf{s}_\nu \oplus k = \mathbf{t}_\nu \oplus k$. Die Addition von $-k = 27 - k$ bringt hier natürlich das Gewünschte, nämlich $\mathbf{s}_\nu = \mathbf{t}_\nu$.

Es bleibt noch zu bestimmen, wie in diesem System entschlüsselt wird, d.h. es sind die Umkehrfunktionen der $\Omega(k)$ anzugeben. Aber die $\Omega(k)$ sind natürlich surjektiv, denn zu gegebener Folge $(\mathbf{t}_\nu)_{\nu \in N}$ ist

$$\Omega(k)((\mathbf{t}_\nu \ominus k)_{\nu \in N}) = (\mathbf{t}_\nu)_{\nu \in N}$$

womit auch die Umkehrabbildungen $\Omega(k)^{-1}$ gegeben sind, nämlich als

$$\Omega(k)^{-1}((\mathbf{t}_\nu)_{\nu \in N}) = (\mathbf{t}_\nu \ominus k)_{\nu \in N}$$

Dabei ist $u \ominus v$ eine Abkürzung von $u \oplus (-v)$.

Das soeben vorgestellte auf $\mathbb{Z}_{27}$ als Alphabet aufbauende Chiffriersystem mit Substitutionsschlüssel kann ganz allgemein formuliert werden. Es sei dazu für irgendeine Menge U

$$\mathbf{P} \langle U \rangle$$

die Menge aller bijektiven Abbildungen oder *Permutationen* $\xi: U \rightarrow U$ von U . Damit wird durch das folgende Schema

- (i) $\mathbf{T} = A$ mit einer endlichen Menge A
- (ii) $\mathcal{T} = \mathcal{E}_A$
- (iii) $\mathbf{G} = A$
- (iv) $\mathcal{G} = \mathcal{E}_A$
- (v) $\mathcal{K} = \mathbf{P} \langle A \rangle$
- (vi) Die Abbildung $\Omega: \mathbf{P} \langle A \rangle \rightarrow \mathbf{I} \langle \mathcal{E}_A, \mathcal{E}_A \rangle$ ist gegeben als

$$\Omega(\xi)((\mathbf{a}_\nu)_{\nu \in N}) = (\xi(\mathbf{a}_\nu))_{\nu \in N}$$

wobei $N \subset \mathbb{N}$ endlich und $\mathbf{a}: N \rightarrow A$ eine Folge von Elementen aus A ist.

ein Chiffriersystem mit Substitutionsschlüssel definiert. Hier hat man im Wesentlichen zu zeigen, daß $\mathbf{a}_\nu = \mathbf{b}_\nu$ aus $\xi(\mathbf{a}_\nu) = \xi(\mathbf{b}_\nu)$ folgt, aber das ist wegen der Bijektivität von ξ klar. Und es gilt natürlich die Gleichung $\Omega(\xi)^{-1} = \Omega(\xi^{-1})$.

2. Eine kurze Einführung in Chiffriersysteme

Offensichtlich wird durch $x \mapsto x \oplus k$ eine bijektive Abbildung $\mathbb{Z}_{27} \rightarrow \mathbb{Z}_{27}$ definiert, d.h. das Chiffriersystem mit Substitutionsschlüssel ist tatsächlich eine Spezialisierung des eben vorgestellten Systems mit Permutationen. Es gibt noch viele solcher Spezialisierungen, wie es auch noch viele andere mit Raffinesse ausgedachte Verschlüsselungssysteme gibt, die heute allerdings aus Gründen mangelnder Sicherheit höchstens noch historische Bedeutung haben. Manche setzen mechanische Hilfsmittel ein, deren Spektrum sich vom simplen Streifen Papier bis zu ausgeklügelten Zahnradmechaniken erstreckt. Ein diesbezüglich leicht lesbares Buch mit vielen Illustrationen ist [ASMW].

Die bisher betrachteten Verschlüsselungssysteme chiffrieren und dechiffrieren die Zeichen so, wie sie der Reihe nach anfallen (daher *stream cipher* genannt). Manche Systeme fassen jedoch eine bestimmte Anzahl von Zeichen in einem Block zusammen und chiffrieren und dechiffrieren diesen Block als eine Einheit (daher *block cipher* genannt). Zu diesem Typ zählen auch die im Buch ausführlich vorgestellten Systeme, also AES und das Rucksacksystem.

Um nun auch diese Verschlüsselungssysteme in das obige Schema zu bringen wird aus der Menge $\mathcal{F}_M$ aller Folgen mit Elementen in einer Menge M neben der Teilmenge aller endlichen Folgen eine weitere Teilmenge ausgesondert:

$$\mathcal{E}_M^{16} = \{ f: N \rightarrow M \mid N \subset \mathbb{N} \wedge \#(N) = 16 \}$$

Es ist also die Menge aller Folgen der Länge 16, und natürlich ist $\mathcal{E}_M^{16} \subset \mathcal{E}_M$. Weiterhin sei $\mathcal{M}_M$ die Menge der 4×4 -Matrizen mit Koeffizienten in der Menge M :

$$\mathcal{M}_M = \left\{ \begin{pmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \\ m_{30} & m_{31} & m_{32} & m_{33} \end{pmatrix} \mid m_{\nu\mu} \in M \right\}$$

Schließlich sei noch eine Abbildung $\mathbf{A}: \mathcal{E}_M^{16} \rightarrow \mathcal{M}_M$ definiert durch

$$\mathbf{A}((f_\nu)_{\nu \in N}) = \begin{pmatrix} f_{\nu_0} & f_{\nu_4} & f_{\nu_8} & f_{\nu_{12}} \\ f_{\nu_1} & f_{\nu_5} & f_{\nu_9} & f_{\nu_{13}} \\ f_{\nu_2} & f_{\nu_6} & f_{\nu_{10}} & f_{\nu_{14}} \\ f_{\nu_3} & f_{\nu_7} & f_{\nu_{11}} & f_{\nu_{15}} \end{pmatrix} \quad \text{mit } N = \{\nu_0, \nu_1, \dots, \nu_{15}\} \text{ und } \#(N) = 16$$

Diese Abbildung ist offensichtlich eine Bijektion, die eine Folge der Länge 16 von Elementen aus M zu einer 4×4 -Matrix zusammenfasst (siehe z.B. auch Abschnitt 4).

Bei den bisher vorgestellten Verschlüsselungssystemen ist der Schlüssel sowohl beim Sender als auch beim Empfänger eines Textes streng geheim zu halten. Vor dem Empfang eines Textes muß der Schlüssel also auf einem sicheren Wege zum Empfänger gelangen. Denn bei diesen Systemen ist bei gegebenem Schlüssel $\mathbf{k} \in \mathcal{K}$ sowohl die Verschlüsselungsabbildung $\Omega(\mathbf{k}): \mathcal{T} \rightarrow \mathcal{G}$ als auch deren Umkehrung, die Entschlüsselungsabbildung $\Omega(\mathbf{k})^{-1}: \mathcal{G} \rightarrow \mathcal{T}$ bekannt (hier wird die Bijektivität von $\Omega(\mathbf{k})$ angenommen). Ein verschlüsselter Text $\mathbf{g} = \Omega(\mathbf{k})(\mathbf{t})$ kann also von jedem, der den Schlüssel $\mathbf{k}$ kennt, sofort als $\mathbf{t} = \Omega(\mathbf{k})^{-1}(\mathbf{g})$ entschlüsselt werden.

Betrachtet man jedoch die Verschlüsselungssysteme genauer, dann fällt auf, daß der Zusammenhang zwischen $\Omega(\mathbf{k})$ und $\Omega(\mathbf{k})^{-1}$ wie folgt beschrieben werden kann: Es gibt eine Abbildung $\zeta: \mathcal{K} \rightarrow \mathcal{K}$ so, daß

$$\Omega(\mathbf{k})^{-1} = \Omega(\zeta(\mathbf{k}))$$

gilt. Im Chiffriersystem mit $\mathcal{K} = \mathbb{Z}_{27}$ ist beispielsweise $\zeta(k) = -k$, im System mit $\mathcal{K} = \mathbf{P}\langle A \rangle$ ist $\zeta(\xi) = \xi^{-1}$, und (etwas vorgegriffen) im HILLSchen System ist $\zeta(\mathbf{K}) = \mathbf{K}^{-1}$. In diesen Beispielen ist $\zeta(\mathbf{k})$ aus $\mathbf{k}$ leicht zu berechnen.

Aber einmal angenommen, es wäre unmöglich oder zumindest praktisch unmöglich, $\tilde{\mathbf{k}} = \zeta(\mathbf{k})$ aus $\mathbf{k}$ zu bestimmen, ohne die Abbildung ζ zu kennen. Dann könnte doch die Chiffrierabbildung $\Omega(\mathbf{k})$ öffentlich so zugänglich gemacht werden, etwa in einer Datenbank, daß **jeder** damit einen Text verschlüsseln könnte. Entschlüsseln könnte diesen Text nur derjenige, der die Abbildung ζ und damit die Entschlüsselungsabbildung $\Omega(\mathbf{k})^{-1} = \Omega(\zeta(\mathbf{k}))$ kennt. Bei Einsatz einer solchen Abbildung ζ ist kein gesicherter Schlüsseltransfer vom Empfänger an den Sender notwendig.

Nun hält die Mathematik eine Reihe von Problemen bereit, die bei geeigneter Parameterwahl praktisch unlösbar sind. Das bekannteste Problem ist wohl die Zerlegung einer natürlichen Zahl in ihre Primzahlkomponenten, die bei derzeitiger Rechentechnik bei einer geeignet gewählten Zahl unmöglich zu berechnen ist.

Problematisch bei vielen dieser Probleme ist allerdings, daß nur *vermutet* wird, daß sie praktisch unlösbar sind. Ein Verschlüsselungssystem, das auf der angenommenen (praktischen) Unmöglichkeit der Lösung eines Problems beruht, wird über Nacht wertlos, wenn doch eine Lösungsmöglichkeit gefunden wird. Ein Problem jedoch, das schon viele Jahrhunderte über traktiert wurde, dürfte für die nähere Zukunft als sicher gelten.

Eine bijektive Abbildung $f: A \rightarrow B$, mit der $b = f(a)$ einfach zu berechnen ist, mit der es jedoch unmöglich oder doch sehr schwer ist, $a = f^{-1}(b)$ zu berechnen, wird oft aus offensichtlichen Gründen mit dem Namen **Falltürfunktion** bezeichnet. Hier wäre also $\Omega(\mathbf{k})$ die Falltürfunktion, deren Inverse $\Omega(\zeta(\mathbf{k}))$ ohne die Kenntnis der Abbildung ζ unmöglich zu bestimmen ist.

Ein bekanntes Beispiel ist die diskrete Exponentialfunktion. Nach Kapitel 10 gibt es zu jeder Primzahlpotenz $q = p^k$ ein $a \in \{1, \dots, q-1\}$ so, daß es zu jedem $u \in \{1, \dots, q-1\}$ **genau ein** $\nu \in \{0, \dots, q-2\}$ gibt mit $u = a^\nu$. Durch

$$\exp_a(\nu) = a^\nu$$

wird daher eine **bijektive** Abbildung $\exp_a: \{0, \dots, q-2\} \rightarrow \{1, \dots, q-1\}$ definiert, die diskrete Exponentialfunktion (zum primitiven Element a).

Die Bezeichnung *Exponentialfunktion* darf allerdings nicht darüber hinwegtäuschen, daß $\exp_a$ nur eine Approximation der reellen Exponentialfunktion darstellt. Man kann nicht erwarten, daß die diskrete Exponentialfunktion alle Eigenschaften der reellen Funktion getreulich nachahmt. Aus der Eigenschaft $\exp_a(x) \exp_a(y) = \exp_a(x+y)$ der reellen Exponentialfunktion wird nämlich

$$\exp_a(\nu) \exp_a(\mu) = \exp_a(\varrho_{q-1}(\nu + \mu))$$

Natürlich gilt für das Potenzenprodukt auch $a^\nu a^\mu = a^{\nu+\mu}$, für die Exponentialfunktion ist jedoch $\nu \in \{0, \dots, q-2\}$ gefordert, um $\exp_a(\nu)$ bilden zu können, d.h. der Exponent ist modulo $q-1$ zu reduzieren. Jedenfalls ist $\exp_a$ eine bijektive Abbildung, die eine Umkehrabbildung

$$\log_a: \{1, \dots, q-1\} \rightarrow \{0, \dots, q-2\}$$

besitzt, die entsprechend diskreter Logarithmus genannt wird. Auch die bekannte Eigenschaft $\log_a(xy) = \log_a(x) + \log_a(y)$ der reellen Logarithmusfunktion kann nur in approximativer Gestalt erwartet werden:

$$\log_a(\nu\mu) = \varrho_{q-1}(\log_a(\nu) + \log_a(\mu))$$

2. Eine kurze Einführung in Chiffriersysteme

In diesem Zusammenhang ist es allerdings bedeutender, daß die diskrete Exponentialfunktion eine Falltürfunktion ist. Die Berechnung von $\exp_a(\nu)$ ist sehr einfach auszuführen für kleine $\nu \in \{0, \dots, q-2\}$ und es gibt schnelle und sehr schnelle Algorithmen für große ν . Niemand kennt jedoch einen schnellen Algorithmus zur Berechnung von $\log_a(\mu)$ für große $\mu \in \{1, \dots, q-1\}$. In der Praxis ist q eine Primzahl mit mindestens 100 Dezimalziffern, es ist also unmöglich, alle $\nu \in \{0, \dots, q-2\}$ zu durchlaufen und zu prüfen, ob $\exp_a(\nu) = \mu$ gilt.

Diese Eigenschaft der diskreten Logarithmusfunktion, sehr schwierig berechenbar zu sein, wird auch tatsächlich in einem Chiffriersystem genutzt, und zwar in dem System von ELGAMAL.

Es folgen noch zwei Beispiele von Verschlüsselungssystemen. Das erste, das System von HILL, ist vom Anwender her gesehen dem System AES ein wenig ähnlich, ist aber sehr viel einfacher strukturiert und kann so als eine leicht verdauliche Einstimmung auf AES dienen. Das zweite System ist RSA, das bekannteste aller Falltürsysteme. Es ist vom Konzept her nicht sonderlich schwierig und als ein Beispiel für ein Falltürsystem gut geeignet.

2.2. Das Chiffriersystem von HILL

In diesem Abschnitt wird eine Abart des Chiffriersystems von HILL vorgestellt, die möglichst dem System AES angenähert ist, jedenfalls was das Äußere, also die Schnittstelle zum Anwender, betrifft. Es ist zwar schon etwas in die Jahre gekommen — es wurde am Anfang des vorigen Jahrhunderts entwickelt — kann aber mit etwas Vorsicht durchaus noch heute eingesetzt werden, wenn es nicht auf die allerhöchste Geheimhaltungsstufe ankommt.

Die Klartexte und Geheimtexte des Chiffriersystems sind Elemente $\mathbf{T}$ der Menge $\mathcal{M}$ der 4×4 -Matrizen mit Koeffizienten im Körper $\mathbb{K}_{2^8} = \mathbb{K}_{256}$:

$$\mathbf{T} = \begin{pmatrix} \mathbf{t}_{00} & \mathbf{t}_{01} & \mathbf{t}_{02} & \mathbf{t}_{03} \\ \mathbf{t}_{10} & \mathbf{t}_{11} & \mathbf{t}_{12} & \mathbf{t}_{13} \\ \mathbf{t}_{20} & \mathbf{t}_{21} & \mathbf{t}_{22} & \mathbf{t}_{23} \\ \mathbf{t}_{30} & \mathbf{t}_{31} & \mathbf{t}_{32} & \mathbf{t}_{33} \end{pmatrix}$$

Die Schlüssel $\mathbf{S}$ des Systems sind ebenfalls Matrizen aus $\mathcal{M}$, sie müssen jedoch aus der Teilmenge $\mathcal{S} \subset \mathcal{M}$ der **regulären** Matrizen gewählt werden. Das bedeutet also, daß es zu jedem Schlüssel $\mathbf{S}$ einen Schlüssel $\mathbf{S}^{-1}$ gibt mit

$$\mathbf{S}\mathbf{S}^{-1} = \mathbf{S}^{-1}\mathbf{S} = \mathbf{I}$$

dabei ist $\mathbf{I}$ die 4×4 -Einheitsmatrix mit Einsen (also 01_{16}) in der Hauptdiagonalen und überall sonst Nullen (also 00_{16}). Die Multiplikation der Matrizen $\mathbf{S}$ und $\mathbf{S}^{-1}$ ist natürlich mit der Addition und Multiplikation des Körpers $\mathbb{K}_{2^8}$ auszuführen (hier weicht das vorgestellte Verschlüsselungssystem vom ursprünglichen HILLschen System ab). Und schließlich sind die zu einem Schlüssel $\mathbf{S}$ gehörigen Chiffrierabbildungen

$$\Psi_{\mathbf{S}}: \mathcal{M} \longrightarrow \mathcal{M} \quad \Psi_{\mathbf{S}}^{-1}: \mathcal{M} \longrightarrow \mathcal{M}$$

definiert als die folgenden Matrizenprodukte:

$$\Psi_{\mathbf{S}}(\mathbf{T}) = \mathbf{S}\mathbf{T} \quad \Psi_{\mathbf{S}}^{-1}(\mathbf{T}) = \mathbf{S}^{-1}\mathbf{T}$$

Hier besteht also der einfache Zusammenhang $\Psi_{\mathbf{S}}^{-1} = \Psi_{\mathbf{S}^{-1}}$, und $\Psi_{\mathbf{S}}$ und $\Psi_{\mathbf{S}}^{-1}$ sind offensichtlich invers zueinander:

$$\Psi_{\mathbf{S}}^{-1}(\Psi_{\mathbf{S}}(\mathbf{T})) = \mathbf{S}^{-1}\mathbf{S}\mathbf{T} = \mathbf{I}\mathbf{T} = \mathbf{T}$$

Ausführlich geschrieben wird die Chiffrierfunktion (wie auch ihre Umkehrung) wie folgt gebildet:

$$\mathbf{S}\mathbf{T} = \begin{pmatrix} \mathbf{s}_{00} & \mathbf{s}_{01} & \mathbf{s}_{02} & \mathbf{s}_{03} \\ \mathbf{s}_{10} & \mathbf{s}_{11} & \mathbf{s}_{12} & \mathbf{s}_{13} \\ \mathbf{s}_{20} & \mathbf{s}_{21} & \mathbf{s}_{22} & \mathbf{s}_{23} \\ \mathbf{s}_{30} & \mathbf{s}_{31} & \mathbf{s}_{32} & \mathbf{s}_{33} \end{pmatrix} \begin{pmatrix} \mathbf{t}_{00} & \mathbf{t}_{01} & \mathbf{t}_{02} & \mathbf{t}_{03} \\ \mathbf{t}_{10} & \mathbf{t}_{11} & \mathbf{t}_{12} & \mathbf{t}_{13} \\ \mathbf{t}_{20} & \mathbf{t}_{21} & \mathbf{t}_{22} & \mathbf{t}_{23} \\ \mathbf{t}_{30} & \mathbf{t}_{31} & \mathbf{t}_{32} & \mathbf{t}_{33} \end{pmatrix} = \begin{pmatrix} \mathbf{g}_{00} & \mathbf{g}_{01} & \mathbf{g}_{02} & \mathbf{g}_{03} \\ \mathbf{g}_{10} & \mathbf{g}_{11} & \mathbf{g}_{12} & \mathbf{g}_{13} \\ \mathbf{g}_{20} & \mathbf{g}_{21} & \mathbf{g}_{22} & \mathbf{g}_{23} \\ \mathbf{g}_{30} & \mathbf{g}_{31} & \mathbf{g}_{32} & \mathbf{g}_{33} \end{pmatrix} = \mathbf{G}$$

Die Koeffizienten $\mathbf{g}_{\mu\nu}$ der Matrix $\mathbf{G}$ sind gegeben als

$$\mathbf{g}_{\mu\nu} = \mathbf{s}_{\mu 0}\mathbf{t}_{0\nu} + \mathbf{s}_{\mu 1}\mathbf{t}_{1\nu} + \mathbf{s}_{\mu 2}\mathbf{t}_{2\nu} + \mathbf{s}_{\mu 3}\mathbf{t}_{3\nu} \quad \mu, \nu \in \{0, 1, 2, 3\}$$

Die Additionen und Multiplikationen sind dabei die des Körpers $\mathbb{K}_{2^8}$. Eine solche Multiplikation wird mit Hilfe von Polynommultiplikationen und Polynomdivisionen ausgeführt, die beide hoch

2. Eine kurze Einführung in Chiffriersysteme

nichtlinear sind. Die Chiffrierabbildung besitzt deshalb einen gut ausgebildeten Verschleierungseffekt, womit auf die von SHANNON geforderte Diffusion angespielt ist.

Nun sind Matrizen mit Koeffizienten in $\mathbb{K}_{2^8}$ für Anwender des Chiffrierverfahrens sicherlich ungewohnte wenn nicht sogar befremdliche Klartexte und Geheimtexte. Um eine angenehmere Anwendungsschnittstelle zu bekommen, wird wie folgt vorgegangen:

- Es wird von der speziellen Struktur der Matrixkoeffizienten abgelassen, sie werden nur noch als eben strukturelose Bytes oder Bitoktets $t_{\mu\nu}$ angesehen.
- Die Matrixspalten werden zu einem Vektor $\mathbf{t}$ von 16 Bytes zusammengesetzt.

Es ist eigentlich gleichgültig, auf welche Weise der Vektor $\mathbf{t}$ aus den Spalten der Matrix gebildet wird. Hier geschieht das folgendermaßen:

$$\mathbf{t} = (t_0 \ t_1 \ t_2 \ t_3 \ t_4 \ \cdots \ t_{15}) = (\mathbf{t}_{00} \ \mathbf{t}_{10} \ \mathbf{t}_{20} \ \mathbf{t}_{30} \ \mathbf{t}_{01} \ \mathbf{t}_{11} \ \mathbf{t}_{21} \ \cdots \ \mathbf{t}_{23} \ \mathbf{t}_{33})$$

Als ein Beispiel soll die bekannte Textzeile `Habe_nun,_ach!_` verschlüsselt werden. Verwendet man für die Zeichen des Textes den ASCII-Code, so erhält man den Anwenderklartext

$$\mathbf{t} = (48_{16} \ 61_{16} \ 62_{16} \ 65_{16} \ 20_{16} \ 6E_{16} \ 75_{16} \ 6E_{16} \ 2C_{16} \ 20_{16} \ 61_{16} \ 63_{16} \ 68_{16} \ 21_{16} \ 20_{16} \ 20_{16})$$

und daraus den Klartext des Chiffriersystems als

$$\mathbf{T} = \begin{pmatrix} 48_{16} & 20_{16} & 2C_{16} & 68_{16} \\ 61_{16} & 6E_{16} & 20_{16} & 21_{16} \\ 62_{16} & 75_{16} & 61_{16} & 20_{16} \\ 65_{16} & 6E_{16} & 63_{16} & 20_{16} \end{pmatrix}$$

Wie man sich Schlüssel verschaffen kann wird weiter unten ausführlich diskutiert. In diesem Beispiel wird der Schlüssel

$$\mathbf{S} = \begin{pmatrix} 01_{16} & 23_{16} & 71_{16} & 15_{16} \\ 01_{16} & 7F_{16} & F1_{16} & 40_{16} \\ 01_{16} & B1_{16} & 7B_{16} & 08_{16} \\ 01_{16} & EA_{16} & EE_{16} & 61_{16} \end{pmatrix} \quad \mathbf{S}^{-1} = \begin{pmatrix} 54_{16} & 16_{16} & A0_{16} & E3_{16} \\ 91_{16} & 8B_{16} & 33_{16} & 29_{16} \\ F6_{16} & BF_{16} & 83_{16} & CA_{16} \\ E0_{16} & 6B_{16} & 39_{16} & B2_{16} \end{pmatrix}$$

eingesetzt. Zur Not kann die Chiffrierung auch mit den Tabellen in Kapitel 10 durchgeführt werden. Wie es auch gemacht wird, der Geheimtext ist jedenfalls

$$\mathbf{ST} = \begin{pmatrix} 01_{16} & 23_{16} & 71_{16} & 15_{16} \\ 01_{16} & 7F_{16} & F1_{16} & 40_{16} \\ 01_{16} & B1_{16} & 7B_{16} & 08_{16} \\ 01_{16} & EA_{16} & EE_{16} & 61_{16} \end{pmatrix} \begin{pmatrix} 48_{16} & 20_{16} & 2C_{16} & 68_{16} \\ 61_{16} & 6E_{16} & 20_{16} & 21_{16} \\ 62_{16} & 75_{16} & 61_{16} & 20_{16} \\ 65_{16} & 6E_{16} & 63_{16} & 20_{16} \end{pmatrix} = \begin{pmatrix} D0_{16} & CA_{16} & 52_{16} & 43_{16} \\ 80_{16} & EB_{16} & BE_{16} & EF_{16} \\ 88_{16} & F6_{16} & BB_{16} & BC_{16} \\ DF_{16} & 62_{16} & 3D_{16} & BE_{16} \end{pmatrix} = \mathbf{G}$$

was zu folgendem Anwendergeheimtext führt:

$$\mathbf{g} = (D0_{16} \ 80_{16} \ 88_{16} \ DF_{16} \ CA_{16} \ EB_{16} \ F6_{16} \ 62_{16} \ 52_{16} \ BE_{16} \ BB_{16} \ 3D_{16} \ 43_{16} \ EF_{16} \ BC_{16} \ BE_{16})$$

Wie ein Vergleich von $\mathbf{t}$ und $\mathbf{g}$ zeigt, werden mehrfach vorkommende Buchstaben (also die Buchstaben **a** und **n**) verschieden verschlüsselt. Mit statistischen Untersuchungen über die Häufigkeit gewisser Buchstaben in einem Text einer Sprache kann man dieses Chiffriersystem daher nicht überlisten. Das gelingt jedoch unter gewissen Voraussetzungen auf einfache algebraische Weise,

wie weiter unten gezeigt wird.

Zunächst aber ist noch zu klären, auf welchen Wegen man sich Schlüssel verschaffen kann: Gesucht sind reguläre (invertierbare) 4×4 -Matrizen mit Koeffizienten aus dem Körper $\mathbb{K}_{2^8}$.

Die sich hier sofort stellende Gretchenfrage ist natürlich: Wie kann man die Regularität (oder entgegengesetzt die Singularität) einer Matrix aus $\mathcal{M}$ erkennen? Es gibt den klassischen Weg über die Determinante, doch ist deren orthodoxe Berechnung über die Adjungierten der Matrix rechentechnisch ein Alptraum und sollte vermieden werden. Es gelingt auf viel einfacherem Wege, indem die Matrix in eine obere Dreiecksmatrix umgeformt wird: Ist die Hauptdiagonale der Dreiecksmatrix voll besetzt, dann ist die Matrix regulär.

Man kann so vorgehen, daß man Matrizen mit Zufallszahlen aufbaut, die dann auf Regularität getestet werden. Allerdings ist die Wahrscheinlichkeit, auf diese Weise eine reguläre Matrix zu finden, nicht sehr groß. Hier sind beispielsweise alle 2×2 -Matrizen mit Koeffizienten aus dem Körper $\mathbb{K}_2$:

$$\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

Wie unschwer zu erkennen ist (die Matrixspalten müssen linear unabhängig sein), sind sechs dieser 16 Matrizen regulär, nämlich

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

folglich ist die Wahrscheinlichkeit, daß solch eine mit Zufallszahlen erzeugte Matrix regulär ist, $\frac{3}{8}$ oder 37,5%. Es ist aber sehr einfach, eine große Zahl von regulären Matrizen direkt anzugeben. Sind nämlich $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d} \in \mathbb{K}_{2^8}$, dann ist die damit gebildete Matrix von VANDERMONDE

$$\mathbf{V}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) = \begin{pmatrix} 1 & \mathbf{a} & \mathbf{a}^2 & \mathbf{a}^3 \\ 1 & \mathbf{b} & \mathbf{b}^2 & \mathbf{b}^3 \\ 1 & \mathbf{c} & \mathbf{c}^2 & \mathbf{c}^3 \\ 1 & \mathbf{d} & \mathbf{d}^2 & \mathbf{d}^3 \end{pmatrix}$$

genau dann regulär, wenn die vier Parameter verschieden sind. Man erhält auf diese Weise daher $256 \cdot 255 \cdot 254 \cdot 253 = 4195023360$ reguläre Matrizen. Die obige Beispielsmatrix ist eine solche reguläre VANDERMONDESche Matrix:

$$\begin{pmatrix} 01_{16} & 23_{16} & 71_{16} & 15_{16} \\ 01_{16} & 7F_{16} & F1_{16} & 40_{16} \\ 01_{16} & B1_{16} & 7B_{16} & 08_{16} \\ 01_{16} & EA_{16} & EE_{16} & 61_{16} \end{pmatrix} = \mathbf{V}(23_{16}, 7F_{16}, B1_{16}, EA_{16})$$

Wenn keine Software zur Invertierung von Matrizen mit Koeffizienten in $\mathbb{K}_{2^8}$ zur Verfügung steht, kann die inverse Schlüsselmatrix auch mit einem AVR-Mikrocontroller berechnet werden, siehe dazu [HSMS] 5.2..

Die Implementierung dieses Chiffriersystems ist simpel genug, es gilt nur, das Produkt zwei-

2. Eine kurze Einführung in Chiffriersysteme

er 4×4 -Matrizen zu berechnen. Allerdings gehört die Berechnung solcher Produkte auch zum Chiffriersystem AES, weshalb sich hier eine Probeimplementierung für AVR empfiehlt.

Unterprogramm mck284x4Mul

Es wird das Produkt $\mathbf{G} = \mathbf{ST}$ zweier 4×4 -Matrizen $\mathbf{S}$ und $\mathbf{T}$ mit Koeffizienten im Körper $\mathbb{K}_{2^8}$ berechnet.

Input

- r29:r28** Die Adresse σ einer 4×4 -Matrix $\mathbf{S}$
- r31:r30** Die Adresse τ einer 4×4 -Matrix $\mathbf{T}$
- r27:r26** Die Adresse γ einer 4×4 -Matrix $\mathbf{G}$

Die Koeffizienten der Matrizen werden zeilenweise gespeichert.
Das Unterprogramm kann keine Fehler erzeugen.

1	mck284x4Mul: push5	r16,r17,r18,r19,r20	s×2	
2		ldi	r20,1	1 $\mathbf{r}_{20} \leftarrow k = 1, \text{ d.h. } i = 0$
3	mck284x4Mul05:ld	r16,Y		2 $\mathbf{r}_{16} \leftarrow \mathbf{s}_{i0}$
4		ld	r17,Z	2 $\mathbf{r}_{17} \leftarrow \mathbf{t}_{00}$
5		rcall	mck28Mul	3+ $\mathbf{r}_{18} \leftarrow \mathbf{s}_{i0}\mathbf{t}_{00}$
6		mov	r19,r18	1 $\mathbf{r}_{19} \leftarrow \mathbf{h} = \mathbf{s}_{i0}\mathbf{t}_{00}$
7		ldd	r16,Y+1	2 $\mathbf{r}_{16} \leftarrow \mathbf{s}_{i1}$
8		ldd	r17,Z+4	2 $\mathbf{r}_{17} \leftarrow \mathbf{t}_{10}$
9		rcall	mck28Mul	3+ $\mathbf{r}_{18} \leftarrow \mathbf{s}_{i1}\mathbf{t}_{10}$
10		eor	r19,r18	1 $\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i1}\mathbf{t}_{10}$
11		ldd	r16,Y+2	2 $\mathbf{r}_{16} \leftarrow \mathbf{s}_{i2}$
12		ldd	r17,Z+8	2 $\mathbf{r}_{17} \leftarrow \mathbf{t}_{20}$
13		rcall	mck28Mul	2+ $\mathbf{r}_{18} \leftarrow \mathbf{s}_{i2}\mathbf{t}_{20}$
14		eor	r19,r18	1 $\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i2}\mathbf{t}_{20}$
15		ldd	r16,Y+3	2 $\mathbf{r}_{16} \leftarrow \mathbf{s}_{i3}$
16		ldd	r17,Z+12	2 $\mathbf{r}_{17} \leftarrow \mathbf{t}_{30}$
17		rcall	mck28Mul	3+ $\mathbf{r}_{18} \leftarrow \mathbf{s}_{i3}\mathbf{t}_{30}$
18		eor	r19,r18	1 $\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i3}\mathbf{t}_{30}$
19		st	X+,r19	2 $\mathbf{g}_{i0} \leftarrow \mathbf{h}$
20		ld	r16,Y	2 $\mathbf{r}_{16} \leftarrow \mathbf{s}_{i0}$
21		ldd	r17,Z+1	2 $\mathbf{r}_{17} \leftarrow \mathbf{t}_{01}$
22		rcall	mck28Mul	3+ $\mathbf{r}_{18} \leftarrow \mathbf{s}_{i0}\mathbf{t}_{01}$
23		mov	r19,r18	1 $\mathbf{r}_{19} \leftarrow \mathbf{h} = \mathbf{s}_{i0}\mathbf{t}_{01}$
24		ldd	r16,Y+1	2 $\mathbf{r}_{16} \leftarrow \mathbf{s}_{i1}$
25		ldd	r17,Z+5	2 $\mathbf{r}_{17} \leftarrow \mathbf{t}_{11}$
26		rcall	mck28Mul	3+ $\mathbf{r}_{18} \leftarrow \mathbf{s}_{i1}\mathbf{t}_{11}$
27		eor	r19,r18	1 $\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i1}\mathbf{t}_{11}$
28		ldd	r16,Y+2	2 $\mathbf{r}_{16} \leftarrow \mathbf{s}_{i2}$
29		ldd	r17,Z+9	2 $\mathbf{r}_{17} \leftarrow \mathbf{t}_{21}$
30		rcall	mck28Mul	3+ $\mathbf{r}_{18} \leftarrow \mathbf{s}_{i2}\mathbf{t}_{21}$
31		eor	r19,r18	1 $\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i2}\mathbf{t}_{21}$
32		ldd	r16,Y+3	2 $\mathbf{r}_{16} \leftarrow \mathbf{s}_{i3}$
33		ldd	r17,Z+13	2 $\mathbf{r}_{17} \leftarrow \mathbf{t}_{31}$
34		rcall	mck28Mul	3+ $\mathbf{r}_{18} \leftarrow \mathbf{s}_{i3}\mathbf{t}_{31}$
35		eor	r19,r18	1 $\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i3}\mathbf{t}_{31}$

36	st	X+,r19	2	$\mathbf{g}_{i1} \leftarrow \mathbf{h}$
37	ld	r16,Y	2	$\mathbf{r}_{16} \leftarrow \mathbf{s}_{i0}$
38	ldd	r17,Z+2	2	$\mathbf{r}_{17} \leftarrow \mathbf{t}_{02}$
39	rcall	mcK28Mul	3+	$\mathbf{r}_{18} \leftarrow \mathbf{s}_{i0}\mathbf{t}_{02}$
40	mov	r19,r18	1	$\mathbf{r}_{19} \leftarrow \mathbf{h} = \mathbf{s}_{i0}\mathbf{t}_{02}$
41	ldd	r16,Y+1	2	$\mathbf{r}_{16} \leftarrow \mathbf{s}_{i1}$
42	ldd	r17,Z+6	2	$\mathbf{r}_{17} \leftarrow \mathbf{t}_{12}$
43	rcall	mcK28Mul	3+	$\mathbf{r}_{18} \leftarrow \mathbf{s}_{i1}\mathbf{t}_{12}$
44	eor	r19,r18	1	$\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i1}\mathbf{t}_{12}$
45	ldd	r16,Y+2	2	$\mathbf{r}_{16} \leftarrow \mathbf{s}_{i2}$
46	ldd	r17,Z+10	2	$\mathbf{r}_{17} \leftarrow \mathbf{t}_{22}$
47	rcall	mcK28Mul	3+	$\mathbf{r}_{18} \leftarrow \mathbf{s}_{i2}\mathbf{t}_{22}$
48	eor	r19,r18	1	$\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i2}\mathbf{t}_{22}$
49	ldd	r16,Y+3	2	$\mathbf{r}_{16} \leftarrow \mathbf{s}_{i3}$
50	ldd	r17,Z+14	2	$\mathbf{r}_{17} \leftarrow \mathbf{t}_{32}$
51	rcall	mcK28Mul	3+	$\mathbf{r}_{18} \leftarrow \mathbf{s}_{i3}\mathbf{t}_{32}$
52	eor	r19,r18	1	$\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i3}\mathbf{t}_{32}$
53	st	X+,r19	2	$\mathbf{g}_{i2} \leftarrow \mathbf{h}$
54	ld	r16,Y+	2	$\mathbf{r}_{16} \leftarrow \mathbf{s}_{i0}, \mathbf{r}_{29:28} \leftarrow \mathcal{A}(\mathbf{s}_{i1})$
55	ldd	r17,Z+3	2	$\mathbf{r}_{17} \leftarrow \mathbf{t}_{03}$
56	rcall	mcK28Mul	3+	$\mathbf{r}_{18} \leftarrow \mathbf{s}_{i0}\mathbf{t}_{03}$
57	mov	r19,r18	1	$\mathbf{r}_{19} \leftarrow \mathbf{h} = \mathbf{s}_{i0}\mathbf{t}_{03}$
58	ld	r16,Y+	2	$\mathbf{r}_{16} \leftarrow \mathbf{s}_{i1}, \mathbf{r}_{29:28} \leftarrow \mathcal{A}(\mathbf{s}_{i2})$
59	ldd	r17,Z+7	2	$\mathbf{r}_{17} \leftarrow \mathbf{t}_{13}$
60	rcall	mcK28Mul	3+	$\mathbf{r}_{18} \leftarrow \mathbf{s}_{i1}\mathbf{t}_{13}$
61	eor	r19,r18	1	$\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i1}\mathbf{t}_{13}$
62	ld	r16,Y+	2	$\mathbf{r}_{16} \leftarrow \mathbf{s}_{i2}, \mathbf{r}_{29:28} \leftarrow \mathcal{A}(\mathbf{s}_{i3})$
63	ldd	r17,Z+11	2	$\mathbf{r}_{17} \leftarrow \mathbf{t}_{23}$
64	rcall	mcK28Mul	3+	$\mathbf{r}_{18} \leftarrow \mathbf{s}_{i2}\mathbf{t}_{23}$
65	eor	r19,r18	1	$\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i2}\mathbf{t}_{23}$
66	ld	r16,Y+	2	$\mathbf{r}_{16} \leftarrow \mathbf{s}_{i3}, \mathbf{r}_{29:28} \leftarrow \mathcal{A}(\mathbf{s}_{i+1,3})$
67	ldd	r17,Z+15	2	$\mathbf{r}_{17} \leftarrow \mathbf{t}_{33}$
68	rcall	mcK28Mul	3+	$\mathbf{r}_{18} \leftarrow \mathbf{s}_{i3}\mathbf{t}_{33}$
69	eor	r19,r18	1	$\mathbf{r}_{19} \leftarrow \mathbf{h} + \mathbf{s}_{i3}\mathbf{t}_{33}$
70	st	X+,r19	2	$\mathbf{g}_{i3} \leftarrow \mathbf{h}$
71	lsl	r20	1	$k \leftarrow 2k$
72	sbrs	r20,4	1/2	Falls $k < 8$:
73	rjmp	mcK284x4Mul05	2	Zur $i + 1$ -ten Zeile von $\mathbf{S}$
74	sbiw	r27:r26,16	2	$\mathbf{r}_{27:r26}$ restaurieren
75	sbiw	r29:r28,16	2	$\mathbf{r}_{29:r28}$ restaurieren
76	pop5	r20,r19,r18,r17,r16	5×2	
77	ret		4	

Die Koeffizienten der Matrix $\mathbf{G}$ werden zeilenweise berechnet. Daraus folgt unmittelbar, in welcher Reihenfolge die Zeilen von $\mathbf{S}$ mit den Spalten von $\mathbf{T}$ kombiniert werden müssen. Denn aus

$$\mathbf{g}_{\mu\nu} = \mathbf{s}_{\mu 0}\mathbf{t}_{0\nu} + \mathbf{s}_{\mu 1}\mathbf{t}_{1\nu} + \mathbf{s}_{\mu 2}\mathbf{t}_{2\nu} + \mathbf{s}_{\mu 3}\mathbf{t}_{3\nu} \quad \mu, \nu \in \{0, 1, 2, 3\}$$

2. Eine kurze Einführung in Chiffriersysteme

folgt, daß zuerst die erste Zeile von **S** der Reihe nach mit den vier Spalten von **T** zu kombinieren ist, dann die zweite Zeile von **S** der Reihe nach mit den vier Spalten von **T**, und so fort. Die vier Zeilen von **S** werden in einer Schleife durchlaufen, die Kombination mit den vier Spalten von **T** gemäß obiger Formel wird ohne Schleife direkt ausgeführt.

Es bleibt nur noch zu klären, wie die Adressenregister genutzt werden. Die nachfolgenden Skizzen erläutern die Vorgehensweise. Die erste Reihe zeigt die Berechnung von $\mathbf{g}_{00}$, die zweite Reihe die Berechnung von $\mathbf{g}_{02}$ und die dritte Reihe die Berechnung von $\mathbf{g}_{23}$.

S00	S01	S02	S03

$\mathbf{t}_{00}$			
$\mathbf{t}_{10}$			
$\mathbf{t}_{20}$			
$\mathbf{t}_{30}$			

$\mathbf{g}_{00}$			

s_{00}	s_{01}	s_{02}	s_{03}

•		t_{02}	
		t_{12}	
		t_{22}	
		t_{32}	

		$\bullet$ g_{02}	

$\bullet$	s_{20}	s_{21}	s_{22}

$\bullet$			$\mathbf{t}_{03}$
			$\mathbf{t}_{13}$
			$\mathbf{t}_{23}$
			$\mathbf{t}_{33}$

			$\bullet$ g_{23}

Die Punkte zeigen an, welche Adresse das Adressregister bei der Berechnung von $\mathbf{g}_{\mu\nu}$ enthält. Es wird wie folgt verfahren:

- Register **X** enthält die Adresse des $\mathbf{g}_{\mu\nu}$, das gerade berechnet wird.
- Register **Y** enthält stets die Adresse von $\mathbf{t}_{00}$.
- Bei der Kombination der μ -ten Zeile von **S** mit den vier Spalten von **T** enthält Register **Z** die Adresse von $\mathbf{g}_{\mu 0}$.

Damit liegen auch die relativen Adressen (*offsets*) relativ zu den in den Adressregistern enthaltenen Adressen fest, mit welchen auf die Matrixkoeffizienten zugegriffen wird. Beispielsweise hat $\mathbf{s}_{22}$ bezüglich $\mathbf{Y}$ die relative Adresse 2 und $\mathbf{t}_{23}$ hat bezüglich Register $\mathbf{Z}$ die relative Adresse 11.

Kennt man den Schlüssel **S** nicht, aber auch **keinen** Klartext **T**, kann man den verwendeten Schlüssel nur durch Ausprobieren aller Schlüssel und Klartexte herausbekommen. Das ist bei der großen Zahl der möglichen Matrizen allerdings ein aufwendiges Unterfangen. Hat man jedoch Zugang zu einem Klartext **T** und einem Geheimtext **G**, dann hat man in

$$\mathbf{S}^T = \mathbf{G}$$

ein System linearer Gleichungen für die Koeffizienten $\mathbf{s}_{\mu\nu}$ von $\mathbf{S}$. Ist man vom Glück begünstigt und ist der Klartext $\mathbf{T}$ sogar regulär, dann ergibt sich der Schlüssel direkt als

$$\mathbf{S} = \mathbf{G}\mathbf{T}^{-1}$$

Ist das nicht der Fall, dann lassen sich aus den Gleichungen einige der $\mathbf{s}_{\mu\nu}$ berechnen, für die Übrigen erhält man immerhin einige lineare Gleichungen. Wenn nur wenige Koeffizienten unbestimmt bleiben, kann man hier mit Durchprobieren zum Ziel kommen.

Ist zwar $\mathbf{S}$ unbekannt, hat man jedoch Zugang zum Verschlüsselungsprozess, etwa durch einen Mikroprozessor, der mit dem Verfahren programmiert ist, kann man also den Klartext selbst bestimmen, dann wählt man als Klartext einfach die identische Matrix $\mathbf{I}$ und erhält damit

$$\mathbf{S} = \mathbf{S}\mathbf{I} = \mathbf{G}$$

Das bedeutet also, daß bei diesem Verfahren sowohl Schlüssel als auch alle Klartexte im Verborgenen bleiben müssen.

Zum Schluss wird noch das Chiffrierverfahren von HILL in das oben gegebene Chiffrierschema eingeordnet. Man kann wie folgt vorgehen.

- (i) $\mathbf{T} = \mathbb{K}_{2^8}^{16}$
- (ii) $\mathcal{T} = \mathcal{E}_{\mathbb{K}_{2^8}}^{16}$
- (iii) $\mathbf{G} = \mathbb{K}_{2^8}^{16}$
- (iv) $\mathcal{G} = \mathcal{E}_{\mathbb{K}_{2^8}}^{16}$
- (v) $\mathcal{K} = \{ \mathbf{K} \in \mathcal{M}_{\mathbb{K}_{2^8}} \mid \text{Rang}(\mathbf{K}) = 4 \}$, d.h. die regulären Matrizen aus $\mathcal{M}_{\mathbb{K}_{2^8}}$
- (vi) Die Abbildung $\Omega: \mathcal{K} \longrightarrow \mathbf{P}(\mathcal{E}_{\mathbb{K}_{2^8}}^{16})$ ist gegeben als

$$\Omega(\mathbf{K}) = \Lambda^{-1} \circ \Psi_{\mathbf{K}} \circ \Lambda \quad \text{d.h.} \quad \Omega(\mathbf{K})((f_{\nu})_{\nu \in N}) = \Lambda^{-1}(\Lambda((f_{\nu})_{\nu \in N})\mathbf{K})$$

mit der Chiffrierabbildung $\Psi_{\mathbf{K}}$ des HILLSchen Systems.

Der Schlüsselbereich besteht aus allen regulären, d.h. invertierbaren, 4×4 -Matrizen mit Koeffizienten aus dem Körper $\mathbb{K}_{2^8}$, die Chiffrierabbildungen $\Psi_{\mathbf{K}}$ sind daher bijektiv und damit schließlich auch die Chiffrierabbildungen $\Omega(\mathbf{K})$.

In der Praxis besteht das Alphabet natürlich nicht aus $\mathbb{K}_{2^8}$, sondern aus $\mathbb{K}_2^8$. Also nicht aus Elementen des endlichen Körpers $\mathbb{K}_{2^8}$, der wohl nur sehr wenigen Anwendern des Verfahrens geläufig ist, sondern aus schlichten Bytes oder Bitoktets. Es ist lediglich eine Angelegenheit der Interpretation, die selbstverständlich auch formalisiert werden könnte. Ein klein wenig problematisch ist die Angelegenheit nur deshalb, weil die Polynome, aus welchen die Elemente des Körpers $\mathbb{K}_{2^8}$ hier nach Konstruktion bestehen (siehe Abschnitt A.5), abkürzend (und von ihrer wahren Natur dabei ablenkend) mit Bitoktets oder noch kompakter mit zwei Hexadezimalziffern bezeichnet werden. Würde tatsächlich Polynomnotation verwendet, dann wären noch Übergangsabbildungen von $\mathbb{K}_{2^8}$ nach $\mathbb{K}_2^8$ und zurück einzuführen.

2.3. RSA

Im Jahre 1978 erlebte die Kryptographie eine Revolution, als *Ronald Rivest*, *Adi Shamir* und *Leonard Adleman* das erste Verschlüsselungssystem mit öffentlich bekanntem Schlüssel vorstellten. Man übertreibt sicher nicht, wenn man das System für einen der wichtigsten Beiträge zur Kryptographie überhaupt ansieht.

Das RSA-Chiffriersystem ist leicht zu beschreiben, es ist wie folgt aufgebaut. Es seien p und q zwei (große) Primzahlen, und es sei

$$n = pq \quad m = (p-1)(q-1)$$

Die internen Klartexte t und Geheimtexte g des Verfahrens sind die Elemente des Ringes $\mathbb{Z}_n$, d.h. es gilt $0 \leq t < n$ und $0 \leq g < n$. Weiterhin sei $e \in \mathbb{Z}_m^* = \mathbb{Z}_m \setminus \{0\}$ so gewählt, daß

$$\text{ggT}(e, m) = 1$$

gilt. D.h. e ist ein **invertierbares Element** des Ringes $\mathbb{Z}_m$, das mit m keine echten Teiler gemeinsam hat. Die Zahl e ist der Schlüssel des Verfahrens. Die Verschlüsselungsfunktion $\Psi_e : \mathbb{Z}_n \rightarrow \mathbb{Z}_n$ ist gegeben als

$$\Psi_e(u) = u^e$$

dabei ist die Potenzierung natürlich im Ring $\mathbb{Z}_n$ auszuführen. Wenn also die Multiplikation von $\mathbb{Z}_n$ mit $\otimes$ bezeichnet wird, dann ist

$$\Psi_e(u) = \underbrace{u \otimes u \otimes \cdots \otimes u}_{e\text{-mal}} = \varrho_n(u^e)$$

Die Entschlüsselungsfunktion $\Psi_e^{-1} : \mathbb{Z}_n \rightarrow \mathbb{Z}_n$ schließlich ist gegeben durch

$$\Psi_e^{-1}(u) = u^{\zeta_{p,q}(e)} = \Psi_{\zeta_{p,q}(e)}(u)$$

wobei auch diese Potenzierung natürlich mit der Arithmetik des Ringes $\mathbb{Z}_n$ durchzuführen ist:

$$\Psi_e^{-1}(u) = \underbrace{u \otimes u \otimes \cdots \otimes u}_{\zeta_{p,q}(e)\text{-mal}} = \varrho_n(u^{\zeta_{p,q}(e)})$$

Darin ist die Abbildung $\zeta_{p,q} : \mathbb{Z}_m^* \rightarrow \mathbb{Z}_m^*$ gegeben als

$$\zeta_{p,q}(e) = e^{-1}$$

wobei e^{-1} das zu e inverse Element **im Ring** $\mathbb{Z}_m$ ist, d.h. wenn die Multiplikation von $\mathbb{Z}_m$ mit $\odot$ bezeichnet wird, dann gilt

$$e \odot e^{-1} = 1$$

Ohne die Kenntnis von p und q und damit auch von m ist der Ring $\mathbb{Z}_m$ nicht bekannt, in dem das zu e inverse Element e^{-1} zu berechnen ist. Um aber p und q zu bestimmen ist die Zahl n in ihre beiden Primfaktoren p und q zu zerlegen, und in der Praxis werden die beiden Primzahlen so gewählt, daß diese Zerlegung unmöglich ist, jedenfalls mit den heutigen rechnerischen Mitteln. Beispielsweise sollen p und q Zahlen mit mindestens 100 Dezimalziffern sein, doch ist die Größe

der Primzahlen nicht das einzige zu beachtende Kriterium. Das Teufelchen steckt hier wie so oft in den Details. Jedenfalls liegt in Ψ_e eine echte Falltürfunktion vor.

Es ist aber noch zu zeigen, daß $\Psi_{\zeta_{p,q}(e)}$ tatsächlich die Dechiffrierfunktion ist, d.h. daß wirklich $\Psi_{\zeta_{p,q}(e)} = \Psi_e^{-1}$ gilt. Dieser Beweis ist allerdings nicht ganz einfach.

Dazu ist zunächst zu bemerken, daß nach Konstruktion $e \odot e^{-1} = 1$ im Ring $\mathbb{Z}_m$ gilt. Dessen Multiplikation $\odot$ ist nun für $u, v \in \mathbb{Z}_m$ definiert als

$$u \odot v = \varrho_m(u \cdot v)$$

wobei der Punkt in $u \cdot v$ die Multiplikation in $\mathbb{N}$ bedeutet! Also ist insbesondere

$$e \odot e^{-1} = \varrho_m(e \cdot e^{-1}) = 1$$

Aber $\varrho_m(e \cdot e^{-1})$ ist der Teilerrest, den man erhält, wenn $e \cdot e^{-1}$ durch m dividiert wird, es gibt daher ein $a \in \mathbb{N}$ mit

$$e \cdot e^{-1} = am + \varrho_m(e \cdot e^{-1}) = am + 1 = a(p-1)(q-1) + 1$$

Nun ist offensichtlich $\Psi_{\zeta_{p,q}(e)}(\Psi_e(0)) = 0$ und $\Psi_{\zeta_{p,q}(e)}(\Psi_e(1)) = 1$. Es bleibt also für $u \in \mathbb{Z}_n \setminus \{0, 1\}$ zu zeigen daß $\Psi_{\zeta_{p,q}}(\Psi_e(u)) = u$ gilt. Zunächst hat man mit A.2e

$$\Psi_{\zeta_{p,q}}(\Psi_e(u)) = \Psi_{\zeta_{p,q}}(\varrho_n(u^e)) = \varrho_n(\varrho_n(u^e)^{e^{-1}}) = \varrho_n(u^{e \cdot e^{-1}}) = \varrho_n(u^{a(p-1)(q-1)+1})$$

Angenommen, es ist $\text{ggT}(u, n) = 1$. Nach dem Satz von EULER gibt es ein $b \in \mathbb{N}$ mit

$$u^{(p-1)(q-1)} = u^{\phi(n)} = bn + 1$$

Durch Einsetzen erhält man damit

$$\begin{aligned} \varrho_n(u^{a(p-1)(q-1)+1}) &= \varrho_n((u^{\phi(n)})^a u) = \varrho_n((bn + 1)^a u) = \\ &= \varrho_n(\varrho_n((bn + 1)^a) \varrho_n(u)) = \varrho_n(\varrho_n((bn + 1)^a) u) = \varrho_n(cu) \end{aligned}$$

Die weitere Entwicklung ist mit Einsatz der Eigenschaften von ϱ_n wie folgt:

$$\begin{aligned} c &= \varrho_n((bn + 1)^a) = \varrho_n(\varrho_n(bn + 1)^a) = \varrho_n\left(\left(\varrho_n(\varrho_n(bn) + \varrho_n(1))\right)^a\right) = \\ &= \varrho_n\left(\left(\varrho_n(0 + 1)\right)^a\right) = \varrho_n(1^a) = \varrho_n(1) = 1 \end{aligned}$$

Damit ist wegen $u < n$, d.h. $\varrho_n(u) = u$, die Behauptung für den Fall $\text{ggT}(u, n) = 1$ bewiesen.

Es darf aber nicht übersehen werden, daß $\phi(pq) = (p-1)(q-1)$ nur bei $p \neq q$ gilt, denn bei $p = q$ ist $\phi(p^2) = (p-1)p$. Der vorangehende Beweis, der $\phi(pq) = (p-1)(q-1)$ verwendet, ist bei $p = q$ daher ungültig, und daraus folgt:

Die beiden Primzahlen p und q müssen verschieden gewählt werden

Es sei jetzt $\text{ggT}(u, n) \neq 1$. Aber die einzigen Teiler, die u und $n = pq$ gemeinsam haben können sind p oder q , d.h. es ist $u = p$ oder $u = q$, denn oben wurde $u > 1$ vorausgesetzt. Aus

2. Eine kurze Einführung in Chiffriersysteme

Symmetriegründen genügt es, den Beweis für eine der beiden Primzahlen zu führen, etwa für den Fall $u = p$.

Nun ist $p \neq q$, d.h. es gilt $\text{ggT}(p, q) = 1$. Nach dem Satz von EULER (siehe Abschnitt A.2) gibt es daher ein $b \in \mathbb{N}$ mit $p^{q-1} = p^{\phi(q)} = bq + 1$. Damit erhält man

$$\begin{aligned} p^{a(p-1)(q-1)+1} &= p(p^{(q-1)})^{a(p-1)} = p(bq + 1)^{a(p-1)} = p(bq + 1)^k = \\ &= p \sum_{\kappa=0}^k \binom{k}{\kappa} b^\kappa q^\kappa = p(1 + \sum_{\kappa=1}^k \binom{k}{\kappa} b^\kappa q^\kappa) = p + pq \sum_{\kappa=1}^k \binom{k}{\kappa} b^\kappa q^{\kappa-1} = p + pqc = p + nc \end{aligned}$$

Wegen $p < n$ bedeutet diese Gleichung natürlich $\varrho_n(p^{a(p-1)(q-1)+1}) = p$, was zu zeigen war. Daß auch $\Psi_e(\Psi_{\zeta_{p,q}}(u)) = u$ gilt folgt daraus, daß die Rollen von e und e^{-1} offensichtlich vertauscht werden können.

Es bleibt nun noch zu klären, wie e^{-1} bestimmt werden kann. Das gelingt über die bisher noch gar nicht benutzte Forderung, daß $\text{ggT}(e, m) = 1$ gelten soll. Denn daraus folgt, daß es $a, b \in \mathbb{Z}$ gibt mit $ae + bm = 1$. Die beiden Zahlen a und b können mit dem erweiterten Algorithmus von EUKLID berechnet werden.

Offensichtlich können a und b nicht beide positiv oder beide negativ sein. Auch ist $a = 0$ wegen $m \geq 1$ nicht möglich, ebenso $b = 0$ wegen $e \geq 1$. Es gilt daher entweder $a > 0$ und $b < 0$ oder $a < 0$ und $b > 0$. In jedem Fall gibt es $c, r \in \mathbb{Z}$ mit $|a| = cm + r$ und $0 \leq r < m$. Allerdings ist $r = 0$ nicht möglich, denn eine Gleichung $(ce + b)m = 1$ ist für ganze Zahlen und $m > 1$ nicht möglich.

Zunächst der Fall $a > 0$. Man hat hier $1 = ae + bm = cme + er + bm$ oder $er = (|b| - ce)m + 1$, was natürlich $\varrho_m(er) = 1$ oder $r = e^{-1}$ bedeutet.

Im anderen Fall, bei $a < 0$ und $b > 0$, erhält man zunächst $-er = (ce - b)m + 1$. Das ergibt durch Addition von $0 = me - me$ auf der linken Seite $(m - r)e = (ce - b + 1)m + 1$. Es gilt daher $\varrho_m((m - r)e) = 1$, was wegen $0 < r < m$, also $m - r < m$, bedeutet, daß $m - r = e^{-1}$.

Es gibt nun viele Möglichkeiten, das RSA-System in das Schema von Abschnitt 2.1 einzuordnen. Die nachfolgende Darstellung ist davon angeregt worden, daß das RSA-System nur zur Verschlüsselung kleiner Texte geeignet ist. Das ist natürlich darin begründet, daß die Verschlüsselungsdauer von RSA in einer ganz anderen Größenordnung liegt als etwa die von AES.

Zur Darstellung im Schema sind einige Vorbereitungen nötig. Es sei dazu $k \in \mathbb{N}_+$. Dann ist $\{0, 1, \dots, 2^8 - 1\}^k$ die Menge aller k -Tupel $(t_0, \dots, t_{k-1})$ mit $t_\kappa \in \{0, \dots, 2^8 - 1\}$. Man hat

$$\{0, \dots, 2^8 - 1\}^k \subset \mathcal{E}_{\{0, \dots, 2^8 - 1\}}$$

wenn das Tupel $(t_0, \dots, t_{k-1})$ als Funktionswert einer Folge von Elementen aus $\{0, \dots, 2^8 - 1\}$ mit der Indexmenge $\{0, \dots, k - 1\}$ interpretiert wird. Weiter sei die Abbildung Δ_k definiert durch

$$\Delta_k : \{0, \dots, 2^8 - 1\}^k \longrightarrow \mathbb{Z}_{2^{8k}} \quad \Delta_k(t_0, \dots, t_{k-1}) = \sum_{\kappa=0}^{k-1} t_\kappa 2^{8\kappa}$$

Ein Funktionswert dieser Abbildung ist natürlich nichts anderes als die Darstellung einer natürlichen Zahl u im Bereich $0 \leq u < 2^{8k}$ zur Zahlenbasis 2^8 . Diese Darstellung ist bekanntlich eindeutig, folglich ist die Abbildung Δ_k bijektiv. Dem Leser ist sicher auch bekannt, wie bei gegebenem $u \in \mathbb{Z}_{8k}$ die Entwicklungskoeffizienten u_κ berechnet werden können.

Es seien nun p und q *verschiedene* Primzahlen, mit der Forderung, nicht nahe beieinander liegen zu sollen (für die Verschlüsselungspraxis gibt es noch eine Reihe weiterer Anforderungen). Das von p und q abhängige RSA-System wird hier mit $\text{RSA}\langle p, q \rangle$ bezeichnet. Weiterhin sei $n = pq$ und $m = (p-1)(q-1)$. Schließlich sei k diejenige natürliche Zahl mit $2^{8k} < n < 2^{8k+1}$. Damit kann das folgende Schema formuliert werden:

- (i) $\mathbf{T} = \{0, \dots, 2^8 - 1\}$
- (ii) $\mathcal{T} = \{0, \dots, 2^8 - 1\}^k$
- (iii) $\mathbf{G} = \mathbb{Z}_n$
- (iv) $\mathcal{G} = \mathbb{Z}_n^1$
- (v) $\mathcal{K} = \{e \in \mathbb{Z}_m^* \mid \text{ggT}(e, m) = 1\}$
- (vi) Die Abbildung $\mathbf{\Omega}: \{e \in \mathbb{Z}_m^* \mid \text{ggT}(e, m) = 1\} \longrightarrow \mathbf{I}\langle \{0, \dots, 2^8 - 1\}^k, \mathbb{Z}_n \rangle$ ist gegeben als

$$\mathbf{\Omega}(e) = \Psi_e \circ \Delta_k \quad \text{d.h.} \quad \mathbf{\Omega}(e)(t_0, \dots, t_{k-1}) = \left(\sum_{\kappa=0}^{k-1} t_{\kappa} 2^{8\kappa} \right)^e$$

mit der Chiffrierabbildung Ψ_e von RSA.

Darin wird $\mathbb{Z}_n$ als die Menge aller 1-Tupel aus $\mathbb{Z}_n^1$ aufgefasst. Die Abbildung $\mathbf{\Omega}(e)$ ist injektiv, denn die Abbildungen Δ_k und Ψ_e sind injektiv, doch ist $\mathbf{\Omega}(e)$ nicht surjektiv und deshalb nicht bijektiv, weil Δ_k nicht surjektiv ist. Die Dechiffrierabbildung $\mathbf{\Omega}(e)^{-1}$ ist daher auf der Menge

$$\mathbf{Bild}(\Psi_e \circ \Delta_k) = \Psi_e[\mathbb{Z}_{8k}]$$

definiert. In der Chiffrierpraxis sind (unverfälschte) Geheimtexte ganz selbstverständlich Elemente von $\mathbf{Bild}(\Psi_e \circ \Delta_k)$.

Für ein Beispiel zum Einsatz von $\text{RSA}\langle p, q \rangle$ mit etwas größeren Zahlen seien die beiden Primzahlen gewählt als $p = 2^{24} - 3 = 16777213$ und $q = 2^{25} - 39 = 33554393$. Das ergibt

$$\begin{aligned} n &= pq = 562949198446709 \\ m &= (p-1)(q-1) = 562949148115104 \end{aligned}$$

Der Wert von k kann an der folgenden Tabelle als $k = 6$ abgelesen werden:

$$\begin{aligned} 2^{47} &= 140737488355328 \\ 2^{48} &= 281474976710656 \\ n &= 562949198446709 \\ 2^{49} &= 562949953421312 \end{aligned}$$

Die Klartexte von $\text{RSA}\langle 16777213, 33554393 \rangle$ sind daher alle 6-Tupel $(t_0, t_1, t_2, t_3, t_4, t_5)$ mit Koeffizienten $t_{\kappa} \in \{0, \dots, 2^8 - 1\}$. Interpretiert man die Elemente $t \in \{0, \dots, 2^8 - 1\}$ als ASCII-Zeichen, dann sind die Klartexte tatsächlich Texte. Beispielsweise entspricht der Datumstext 1.1.16 dem Klartext $\mathbf{t} = (31_{16}, 2\text{E}_{16}, 31_{16}, 2\text{E}_{16}, 31_{16}, 36_{16})$ mit

$$\Delta_k(\mathbf{t}) = 31_{16} + 2\text{E}_{16} \cdot 2^8 + 31_{16} \cdot 2^{16} + 2\text{E}_{16} \cdot 2^{24} + 31_{16} \cdot 2^{32} + 36_{16} \cdot 2^{40} = 59584856272433$$

wobei die Elemente des Klartextalphabets $\{0, \dots, 2^8 - 1\}$ in ihrer Hexadezimaldarstellung verwen-

2. Eine kurze Einführung in Chiffriersysteme

det werden. Zur Rückrechnung von $\Delta_k(\mathbf{t})$ auf $\mathbf{t}$ bestimmt man einfach die Hexadezimaldarstellung von 59584856272433.

Es muß nun noch der Schlüssel e bestimmt werden. Die erste erzeugte Zufallszahl ist 48020, sie kann als gerade Zahl nicht teilerfremd mit der geraden Zahl m sein. Der EUKLIDSche Algorithmus bestätigt das natürlich: $13763063304605 \cdot e - 1174 \cdot m = 4$. Der nächste Wurf ist $e = 79031$, und hier gibt EUKLID

$$236659317191687 \cdot e - 33224 \cdot m = 1$$

Es ist also $\text{ggT}(e, m) = 1$ und wegen $236659317191687 < m$ ist

$$e^{-1} = 236659317191687$$

das zu $e = 79031$ inverse Element im Ring $\mathbb{Z}_m$. Es kann jetzt verschlüsselt und entschlüsselt werden. Mit dem obigen Datumstext erhält man den Geheimtext

$$g = \Psi_{79031}(\Delta_6(\mathbf{t})) = 59584856272433^{79031} = 514044458167847$$

wobei die Potenzierung im Ring $\mathbb{Z}_n = \mathbb{Z}_{562949198446709}$ zu erfolgen hat. Man kann dazu den bekannten einigermaßen schnellen Algorithmus verwenden, der die Binärdarstellung von e verwendet, hat aber nach jeder Multiplikation auf das Produkt die Restfunktion ϱ_n anzuwenden.

Zur Entschlüsselung ist nun der Geheimtext mit e^{-1} zu potenzieren, und zwar wieder im Ring $\mathbb{Z}_n$. Wird daher die Multiplikation in $\mathbb{Z}_n$ mit $\otimes$ bezeichnet, dann ist

$$g^{e^{-1}} = \underbrace{g \otimes g \otimes \cdots \otimes g}_{e^{-1}\text{-mal}}$$

zu berechnen. Das führt hier im Beispiel auf

$$\Delta_6(\mathbf{t}) = \Psi_{79031}^{-1}(514044458167847) = 514044458167847^{236659317191687} = 59584856272433$$

Die Hexadezimaldarstellung von 59584856272433 ist 36312E312E31₁₆, womit der Klartext wieder erreicht wäre.

2.4. Nachrede

Wie die umfangreiche Literatur über das Thema Kryptologie zeigt gibt es sehr viele kryptographische Verfahren, oft mit zahllosen Varianten. Folglich konnte das Thema in diesem kurzen einführenden Kapitel nur ganz leicht gestreift werden. Das ist für die weiteren Kapitel des Buches jedoch folgenlos, weil tiefere allgemeine Kenntnisse über Verschlüsselungsverfahren für die Lektüre des Buches nicht erforderlich sind.

So bleiben beispielsweise Sicherheitsfragen nahezu gänzlich unbeachtet. Das geschieht natürlich nicht grundlos. Denn die manchmal recht einfallsreichen Verfahren, die ersonnen wurden, um Geheimschriften zu lesen, ohne den Schlüssel zu kennen, mit dem sie chiffriert wurden, bekannt als das „Knacken“ von Geheimschriften (*codebreaking*), können selten einen Beitrag zur besseren Implementierung eines Verschlüsselungsverfahrens leisten.

Zwar ist es richtig, daß man bei den Versuchen, RSA zu „knacken“, lernen kann, die beiden Primzahlen p und q so zu wählen, daß eben dieses „Knacken“ erschwert wird, aber das ist kein Beitrag zu einer verbesserten Implementierung, denn auf der Implementierungsebene ist eine Primzahl mit einer vorgegebenen Ziffernzahl so gut wie jede andere.

Hinzu kommt, daß manche Attacken gegen ein Chiffrierverfahren gar nicht den Algorithmus des Verfahrens attackieren, sondern ein bestimmtes Anwendungsprotokoll. Es ist klar, daß in solch einem Fall die Implementierung des Verfahrens überhaupt nicht tangiert wird. Was hier mit einer Attacke auf ein Chiffrierverfahren *via* Anwendungsprotokoll gemeint ist erklärt das folgende Beispiel.

Es wird ein Klartext x mit dem Rucksackverfahren l -mal verschlüsselt, und zwar jeweils mit dem öffentlichen Schlüssel $e^{(\lambda)} = (e_1^{(\lambda)}, \dots, e_{96}^{(\lambda)})$ zu einem Geheimtext g_λ :

$$g_\lambda = \sum_{\nu=1}^{96} x_\nu e_\nu^{(\lambda)} \quad \lambda \in \{1, \dots, l\}$$

Werden die l Summen ausgeschrieben, wird leichter erkennbar, daß ein System von linearen Gleichungen vorliegt, mit den bekannten Gleichungskoeffizienten $e_\nu^{(1)}$ und Gleichungskonstanten g_λ und den Unbekannten x_ν :

$$\begin{aligned} g_1 &= \sum_{\nu=1}^{96} x_\nu e_\nu^{(1)} \\ g_2 &= \sum_{\nu=1}^{96} x_\nu e_\nu^{(2)} \\ &\vdots \\ g_l &= \sum_{\nu=1}^{96} x_\nu e_\nu^{(l)} \end{aligned}$$

Stehen daher $l = 96$ Geheimtexte zur Verfügung, dann ist ein lineares Gleichungssystem mit 96 Gleichungen für 96 Unbekannte gegeben, das garantiert eine Lösung besitzt.

Es ist allerdings sehr fraglich, ob mit den gängigen Fließkommaarithmetiken mit 32 oder 64 Bit eine Lösung ermittelt werden kann. denn die Unbekannten $x_\nu \in \{0, 1\}$ sind gemessen an

2. Eine kurze Einführung in Chiffriersysteme

den Gleichungskoeffizienten und an den g_λ extrem klein. Das bedeutet, daß zur Bestimmung der Unbekannten sehr große Zahlen derselben Größenordnung zu subtrahieren sind. Solche Subtraktionen erzeugen aber unweigerlich bei der Differenz einen beträchtlichen Verlust an signifikanten Bits. Mit anderen Worten: Das Gleichungssystem ist extrem schlecht konditioniert!

Eine Möglichkeit, diese Fehlerquelle zu vermeiden, wäre, eine rationale Arithmetik einzusetzen. Aber auch hier besteht wenig Grund zu hoffen. Denn wer jemals mit einem Programmpaket für rationale Arithmetik gearbeitet hat, der weiß, daß die Zähler und Nenner rationaler Zahlen bei auch nur mäßig umfangreicher Rechnung zu enormer Größe anwachsen. Hier bei einem System mit 96 Gleichungen sind deshalb gigantische Zähler und Nenner zu erwarten.

Zu dem allgemeinen Schema für Verschlüsselungsverfahren, das im Kapitel eingeführt wird, ist noch zu sagen, daß manche Verfahren in dieses Schema ziemlich hineingezwungen werden müssen. Das ist bei der Vielzahl der Verfahren und der vielfachen Mittel und Wege zur Realisierung der Verfahren auch gar nicht anders zur erwarten. Es ist immerhin ein Versuch, etwas Ordnung in den immensen *Zoo* von Verschlüsselungsverfahren zu bringen.

Es scheint so zu sein, daß für die zunehmende Sicherheit von Chiffrierverfahren ein Preis zu zahlen ist, nämlich die Zunahme an mathematischem Gehalt der Verfahren. Waren in den Anfangsgründen der Kryptologie noch einfachste Kenntnisse in Algebra zum theoretischen Nachvollzug der Verfahren ausreichend, erfordert der Nachvollzug aller Schritte von AES ein gewisses Vertrautsein mit der Theorie der endlichen Körper und der Polynome über solchen Körpern. Allerdings geht die Kryptographie hier keinen Sonderweg, sie teilt diesen Effekt mit vielen anderen Themenkreisen.

AES und Rucksackverfahren

Theorie und Praxis mit AVR- und dsPIC-Mikrocontrollern

Schmidt, H.; Schwabl-Schmidt, M.

2017, IX, 241 S., Softcover

ISBN: 978-3-658-19703-2