# Kryptographie und ihr Bezug zur Komplexitätstheorie

Gerrit Gruben
Freie Universität Berlin
Fachbereich Mathematik und Informatik
Institut für Informatik

14. Juni 2010

# Inhaltsverzeichnis

# I Einführung

Die Kryptographie ist eine uralte Wissenschaft, die sich aus dem Bedürfnis heraus entwickelt, Kommunikation gegen Unbefugte zu schützen, also seine Daten zu sichern. In dieser Arbeit werden grundlegende Fragen der Kryptographie beantwortet: Wann ist ein Kryptographiesystem sicher? Was ist Pseudozufall und Einwegfunktionen? Was haben diese mit der Kryptographie zu tun?

Gelöst werden sollen von der Kryptographie unter anderem folgende Problemstellungen

1. **Datensicherheit** Schutz vor unbefugtem Lesen von Daten.
2. **Datenintegrität** Schutz vor ungewollter Modifikation der Daten.
3. **Authentifikation** Nachweis einer Identität in dem man
   (a) Etwas weiß
   (b) oder etwas besitzt
   (c) oder etwas ist.

Als grundlegendes mathematisches Objekt betrachten wir Kryptosysteme, die formale Definition:

**Definition 1 (Kryptosystem).**
Ein Paar $(\mathbf{Enc}, \mathbf{Dec})$ von Funktionen

$$\mathbf{Enc} : \mathcal{K} \times \mathcal{M} \to \mathcal{C},\ (k, x) \mapsto \mathbf{Enc}_k(x)$$

und

$$\mathbf{Dec} : \mathcal{K} \times \mathcal{C} \to \mathcal{M},\ (k, y) \mapsto \mathbf{Dec}_K(y)$$

mit den Mengen $\mathcal{M}, \mathcal{C}, \mathcal{K}$ (**Klartext-**,**Chiffretext-** und **Schlüsselmenge**) heißt **Kryptosystem**, wenn zusätzlich die Bedingung gilt

$$\mathbf{Dec}_k \circ \mathbf{Enc}_k = \mathrm{Id}_{\mathcal{M}}\ (\forall k \in \mathcal{K})$$

Im ersten Teil wird die von Shannon eingeführte perfekte Geheimhaltung in Shannon (1949) dargestellt. Hier wird mit Hilfe der Informationstheorie versucht Sicherheit eines Kryptosystems zu definieren. Das Hauptergebnis dieses Paragraphen ist, dass das One-Time-Pad diese Definition erfüllt und dabei minimale Schlüssellänge besitzt - diese ist jedoch immer noch so lang wie der Nachrichtentext.

Im zweiten Paragraphen wird mit Hilfe der Komplexitätstheorie versucht Sicherheit eines Kryptosystems zu beschreiben. Dabei werden die Forderungen gelockert um effizientere Kryptosysteme zu ermöglichen. Es werden Einwegfunktionen und die Rolle der $\mathcal{P} = \mathcal{NP}$ Frage für die Kryptographie erörtert. Zwei Beispiele für vermutete Einwegfunktionen - RSA und die Rabinfunktion - werden gezeigt.

Der dritte Paragraph diskutiert den Pseudozufall und seine Rolle für die Kryptographie. Dabei werden die für die Kryptographie wichtigen Pseudozufallsgeneratoren eingeführt und bewiesen, dass aus der Existenz von Einwegfunktionen sich Pseudozufallsgeneratoren konstruieren lassen. Unter anderem wird dafür das Lemma von Yao und das Goldreich-Levin-Theorem bewiesen. Schliesslich wird der Beweis für eine Abschwächung des Satzes gegeben: Aus Einwegpermutationen lassen sich Pseudozufallsgeneratoren konstruieren.

Schliesslich werden im vierten Teil Zero-Knowledge Beweise geführt. Diese Systeme ermöglichen es Aussagen zu beweisen ohne Informationen preiszugeben, die ein polynomieller Algorithmus eh hätte ermitteln können.

# II Perfekte Geheimhaltung

**Definition 2.**
Ein Kryptosystem $(\mathbf{Enc}, \mathbf{Dec})$ über $\mathbb{B}^n$ bietet **perfekte Geheimhaltung**, falls für jede Verteilung $\mathcal{D}$ von Nachrichten, jeder Nachricht $m$ und jeden auftretenden Chiffretext $c$ gilt

$$\mathbf{Pr}\left[M = x \mid C = c\right] = \mathbf{Pr}\left[M = x\right]$$

Das heisst unter der Kenntnis eines beliebigen Chiffretextes erhalten wir keine Informationen über den Klartext im wahrscheinlichkeitstheoretischen Sinne. Äquivalent dazu ist, dass die Verteilung der Klartexte unabhängig von der der Chiffretexte ist. D. h. $\mathbf{Enc}_{U_n}(x) = \mathbf{Enc}_{U_n}(x')$

---

**Lemma 1** Für ein Kryptosystem $\Pi = (\mathbf{Enc}, \mathbf{Dec})$ sind die folgenden Aussagen äquivalent

1. $\Pi$ bietet perfekte Geheimhaltung.
2. Für jede Verteilung auf $\mathcal{M}$ und allen $x \in \mathcal{M}$, $y \in \mathcal{C}$ gilt:

$$\mathbf{Pr}\left[C = y \mid M = x\right] = \mathbf{Pr}\left[C = y\right].$$

3. Für jede Verteilung auf $\mathcal{M}$ und allen $x_0, x_1 \in \mathcal{M}$ und $c \in \mathcal{C}$ gilt

$$\mathbf{Pr}\left[C = y \mid M = x_0\right] = \mathbf{Pr}\left[C = y \mid M = x_1\right]$$

---

Beweis
1. $\Leftrightarrow$ 2.: Einfache Umformung und Bayes.
1., 2. $\Leftrightarrow$ 3.:
'$\Rightarrow$': Folgt direkt aus 2..

'$\Leftarrow$': Sei eine beliebige Verteilung auf $\mathcal{M}$ und $x_0 \in \mathcal{M}$ und $y \in \mathcal{C}$ gewählt. Definiere $p := \mathbf{Pr}[C = y \mid M = m_0]$. Dann gilt

$$\begin{aligned}\mathbf{Pr}[C = y] &= \sum_{x \in \mathcal{M}} \mathbf{Pr}[C = y \mid M = x] \cdot \mathbf{Pr}[M = x] \\ &= \sum_{x \in \mathcal{M}} p \, \mathbf{Pr}[M = x] \\ &= p \\ &= \mathbf{Pr}[C = y \mid M = x_0]\end{aligned}$$

■

---

**Lemma 2** Ein Kryptosystem $(\mathbf{Enc}, \mathbf{Dec})$ mit Klartextmenge $\mathcal{M}$ und Schlüsselmenge $\mathcal{K}$ bietet perfekte Geheimhaltung, dann gilt

$$|\mathcal{K}| \geq |\mathcal{M}|$$

---

Beweis
Annahme: $|\mathcal{K}| < |\mathcal{M}|$. Sei $M$ gleichverteilt und $y \in \mathcal{C}$ ein Chiffretext, der auftritt. Definiere

$$M(c) := \{x \mid x = \mathbf{Dec}_k(y), k \in \mathcal{K}\}$$

Es gilt $|M(y)| \leq |\mathcal{K}|$ da wir jedem Element in $M(c)$ ein Schlüssel zuordnen können. Da $|\mathcal{K}| < |\mathcal{M}|$ existiert ein $x \in M \backslash M(y)$ für das gilt

$$\mathbf{Pr}[M = x \mid C = y] = 0 < \mathbf{Pr}[M = x]$$

im Widerspruch zur perfekten Geheimhaltung. ■

Shannon lieferte auch eine Charakterisierung für Kryptosysteme mit perfekter Geheimhaltung

**Satz 1 (von Shannon).**
Sei $(\mathbf{Enc}, \mathbf{Dec})$ ein Kryptosystem mit $|\mathcal{K}| = |\mathcal{M}| = |\mathcal{C}|$. Dann bietet das Kryptosysteme perfekte Geheimhaltung genau dann wenn folgende Bedingungen gelten:

1. Die Schlüssel werden gleichverteilt gewählt, d. h. $K \sim U_{\mathcal{K}}$.
2. Für alle Nachrichten $x \in M$ und Chiffretexten $y \in C$ existiert genau ein $k \in K$ mit $y = \mathbf{Enc}_k(x)$.

Beweis
'⇒': Sei $(\mathbf{Enc}, \mathbf{Dec})$ mit perfekter Geheimhaltung. Für alle $x \in \mathcal{M}$ und $y \in \mathcal{C}$ muss ein $k \in \mathcal{K}$ existieren, sodass $y = \mathbf{Enc}_k(x)$. Für ein festes $x \in M$[1] gilt also für die Menge $E(x) := \{\mathbf{Enc}_k(x)\}_{k\in\mathcal{K}}$, dass $|E(x)| \geq |\mathcal{C}|$ und damit $|E(x)| = |\mathcal{C}| = |\mathcal{K}|$. Damit ist $\mathbf{Enc}(\cdot, x)$ injektiv, also existiert maximal ein Schlüssel um $x \mapsto y$ für alle $y \in \mathcal{C}$. Variation von $x \in \mathcal{M}$ und die Existenz eines Schlüssels geben die zweite Aussage.

Nun sei $k \in K$ und das Ziel ist es $\mathbf{Pr}[K = k] = \frac{1}{n}$ mit $n := |\mathcal{K}|$ zu zeigen. Sei $\mathcal{M} = \{x_1, \ldots, x_n\}$ und $y \in \mathcal{C}$ fest. Sei $k_1, \ldots, k_n \in \mathcal{K}$, sodass

$$\mathbf{Enc}_{k_i}(x_i) = y \ (1 \leq i \leq n)$$

Nun gilt wegen der perfekten Geheimhaltung für beliebige $i = 1, \ldots, n$:

$$\begin{aligned}
\mathbf{Pr}[M = x_i] &= \mathbf{Pr}[M = x_i \mid C = y] \\
&= \frac{\mathbf{Pr}[C = y \mid M = x_i] \cdot \mathbf{Pr}[M = x_i]}{\mathbf{Pr}[C = y]} \\
&= \frac{\mathbf{Pr}[K = k_i] \cdot \mathbf{Pr}[M = x_i]}{\mathbf{Pr}[C = y]} \\
&\Leftrightarrow \mathbf{Pr}[C = y] = \mathbf{Pr}[K = k_i]
\end{aligned}$$

$\Rightarrow K \sim U_n$.
'⇐': Es folgt direkt

$$\mathbf{Pr}[C = y \mid M = x] = \frac{1}{|\mathcal{K}|}$$

unabhängig von der Verteilung auf $\mathcal{M}$. Also haben wir für alle Verteilungen auf $\mathcal{M}$, allen $x, x' \in \mathcal{M}$ und jedem $y \in \mathcal{C}$:

$$\mathbf{Pr}[C = y \mid M = x] = \frac{1}{|\mathcal{K}|} = \mathbf{Pr}[C = y \mid M = x']$$

so dass unser Kryptosystem perfekte Geheimhaltung bietet. ∎

Und ein Kryptosystem welches perfekte Geheimhaltung bietet ist:

**Definition 3 (One-Time-Pad).**
Seien $\mathcal{K} = \mathcal{M} = \mathcal{C} = \{0,1\}^n$ für ein $n \in \mathbb{N}$. Dann heisst $(\mathbf{Enc}, \mathbf{Dec})$

$$\mathbf{Enc}_k(x) := x \oplus k, \ \mathbf{Dec}_k(y) := y \oplus k$$

**One-Time-Pad** (Vernam Chiffre).

Der Satz von Shannon zeigt, dass das One-Time-Pad perfekte Geheimhaltung bietet.

[1] Welches auftritt, aber diese Annahme machen wir immer.

# III Berechnungssicherheit

Zuerst noch ein Ergebnis, welches uns ein Strich durch sichere Kryptographie macht.

---

**Lemma 3** Sei $\mathcal{P} = \mathcal{NP}$ und $(\mathbf{Enc}, \mathbf{Dec})$ in Polynomialzeit berechenbar mit einer Schlüssellänge von $n = n(m) < m$ bei Nachrichtenlänge $m$. Dann existiert ein Polynomialzeitalgorithmus $A$, so dass für alle Eingabelängen $m$ gilt: es gibt es ein Paar $x_0, x_1 \in \{0,1\}^m$ mit

$$\mathop{\mathbf{Pr}}_{\substack{b \in_R \{0,1\} \\ k \in_R \{0,1\}}} [A(\mathbf{Enc}_k(x_b)) = b] \geq \frac{3}{4}$$

---

Beweis
Sei $(\mathbf{Enc}, \mathbf{Dec})$ mit $m \in \mathbb{N}$ und $n = n(m) < m$ fest. Definiere $S := \mathbf{Enc}_{U_n}(x_0)$ mit $x_0 := 0^m$. Da $\mathcal{P} = \mathcal{NP}$ ist gibt es polynomiellen Algorithmus $A$ der $S$ entscheidet. Das heisst $A(y)$ ist gleich $[y \in S]$.

Es fehlt nun noch ein geeignetes $x_1 \in \mathbb{B}^m$. Sei dazu $D_x \sim \mathbf{Enc}_{U_n}(x)$ die Verteilung der Verschlüsselungen von $x \in \mathbb{B}^m$. Per Konstruktion ist dann

$$\mathbf{Pr}\left[A(D_{x_0}) = 0\right] = 1.$$

Ferner gilt

$$\begin{aligned} \mathop{\mathbf{Pr}}_{b \in_R B}\left[A(D_{x_b}) = b\right] &= \sum_{b=0}^{1} \mathbf{Pr}\left[b\right] \mathbf{Pr}\left[A(D_{x_b}) = b\right] \\ &= \frac{1}{2} + \frac{1}{2}\,\mathbf{Pr}\left[A(D_{x_1}) = 1\right] \end{aligned}$$

sodass wenn $x_1$ die Ungleichung $\mathbf{Pr}\left[A(D_{x_1}) = 1\right] \geq \frac{1}{2}$ erfüllt der Beweis vollbracht ist. Dies ist aber zu

$$\mathbf{Pr}\left[D_{x_1} \in S\right] \leq \frac{1}{2}$$

äquivalent. Für $x \in \mathbb{B}^m$ und $k \in \mathbb{B}^n$ sei die Zufallsvariable $S(x,k)$ definiert als:

$$S(x,k) := \begin{cases} 1, & E_k(x) \in S \\ 0, & \text{sonst} \end{cases}$$

Zum Zwecke eines Widerspruchsbeweises $\mathbf{Pr}\left[D_{x_1} \in S]\right] > \frac{1}{2}$ für alle $x_1 \in \mathbb{B}^m$. Daraus folgt eine Abschätzung für den Erwartungswert von $S(x,k)$:

$$\mathop{\mathbf{E}}_{\substack{x \in \mathbb{B}^n \\ k \in B^m}} [S(x,k)] > \frac{1}{2} \tag{1}$$

Nun gilt für einen festen Schlüssel $k$, dass die Funktion $E_k(\cdot) : x \mapsto \mathbf{Enc}_k(x)$ wegen $\mathbf{Dec} \circ \mathbf{Enc} = \mathrm{Id}$ injektiv sein muss. Nun gilt die Ungleichungskette

$$|S| \leq 2^n \leq 2^{m-1} < |\operatorname{im} E_k(\cdot)| = 2^m$$

woraus folgt

$$\mathop{\mathbf{E}}_{x\in\mathbb{B}^n}[S(x,k)] \leq \frac{1}{2}$$

und damit im Widerspruch zu (1) folgt

$$\frac{1}{2} < \mathop{\mathbf{E}}_{x,k}[S(x,k)] \leq \mathop{\mathbf{E}}_{k}\left[\frac{1}{2}\right] = \frac{1}{2}$$

■

**Definition 4 (Vernachlässigbare Funktionen).**
Sei $\epsilon : \mathbb{N} \to [0,1] \subseteq \mathbb{R}$, dann heisst $\epsilon$ **vernachlässigbar**, wenn $\epsilon(n) = n^{-\omega(1)}$. D. h. für alle $c > 0$ existiert ein $N \in \mathbb{N}$, sodass $\epsilon(n) < n^{-c}$ für alle $n \geq N$ gilt.

Es gilt

$$\lim_{n\to\infty} \epsilon(n) = 0$$

und dieser Ausdruck konvergiert so schnell, dass man es in praktischen gelangen vernachlässigen kann. Mit vernachlässigbaren Funktionen wollen wir Ereignisse modellieren, welche praktisch nie eintreten.

**Definition 5 (Einwegfunktionen).**
Eine in Polynomialzeit berechenbare Funktion $f : \mathbb{B}^* \to \mathbb{B}^*$ heißt **Einwegfunktion**, falls für alle $A \in \mathcal{PPT}$ gilt:

$$\mathop{\mathbf{Pr}}_{\substack{x\in\mathbb{B}^n \\ y=f(x)}}[A(y) = x' \text{ mit } f(x') = y]$$

ist vernachlässigbar in $n \in \mathbb{N}$.

Gilt $|f(x)| = |x|$ für $x \in \mathbb{B}^*$, so heisst $f$ **längenerhaltend**. Ist $f$ eine injektive , längenerhaltende Einwegfunktion, dann heisst $f$ **Einwegpermutation**.

**Vermutung**: Es existiert eine Einwegfunktion.

**Satz 2.**
Wenn $P = \mathcal{NP}$, dann existieren keine Einwegfunktionen.

Beispiel (Faktorisierung):
Betrachtet man die Multiplikation $a \cdot b =: N$ zweier Zahlen $a, b \in \mathbb{N}$, so ist die Umkehrung im gewissen Sinne die Faktorisierung einer Zahl in ihren Primfaktoren. Der naive Algorithmus, die Probedivision von Teilern bis $\sqrt{N}$ ist exponentiell in der Eingabegröße $\log(N)$. Es gibt einen Faktorisierungsalgorithmus (siehe (Lenstra u. a., 1990)) mit einer oberen Laufzeitschranke von $2^{\mathcal{O}(\log^{1/3} N \sqrt{\log\log N})}$.

Das nächste Beispiel benötigt die **eulersche $\varphi$-Funktion** für diese gilt

$$\varphi(n) := |\{1 \leq j < n \mid \mathrm{ggT}(j,n) = 1\}| = \left|\left(\mathbb{Z}/n\,\mathbb{Z}\right)^*\right|$$

Beispiel (RSA):
Für ein $n \in \mathbb{N}$ sei $N = N(n) \in \mathbb{N}$ eine zusammengesetzte Zahl und $e \in \mathbb{N}$ mit $\mathrm{ggT}(\varphi(n), e) = 1$. Im Regelfall ist $N$ das Produkt zweier verschiedener Primzahlen $p, q \neq 2$. Es ist dann für $x \in \left(\mathbb{Z}/N\,\mathbb{Z}\right)^*$:

$$\mathbf{Enc}_{(N,e)}(x) = RSA_{(N,e)}(x) = \rho_N(x^e)$$

die Chiffrierungsfunktion vom **RSA-Kryptosystem** mit $\rho_N : \mathbb{Z} \twoheadrightarrow \mathbb{Z}/N\,\mathbb{Z}$ der Reduktion modulo $N$. Dechiffriert werden kann ein Chiffretext $y$ mit einem geheim gehaltenen $d$, welches die Gleichung

$$ed \equiv 1 \mod \varphi(N)$$

erfüllt und zwar mit

$$\mathbf{Dec}_{(N,e)}(y) = \rho_N(y^d)$$

Kennt man den Wert von $\varphi(N)$, so lässt sich $d$ effizient mit dem euklidischen Algorithmus bestimmen. Es lässt sich $\varphi(N)$ effizient unter Kenntniss der Primfaktorzerlegung berechnen. Im allgemeinen Fall gilt für $n = \prod_{i=1}^{r} p_i^{\nu_i}$ mit verschiedenen Primzahlen $p_i$:

$$\varphi(n) = \prod_{i=1}^{r} p_i^{\nu_i - 1}(p_i - 1)$$

wie man z. B. mit dem Inklusion-Exklusionsprinzip zeigen kann.

Beispiel (Rabin):
Sei $N = PQ$ wieder das Produkt zweier verschiedener Primzahlen größer 2. Zusätzlich gelte $P, Q \equiv 3 \mod 4$. Wir betrachten die Menge der quadratischen Reste

$$QR_N := \{z \in \mathbb{Z}_N^* \mid \exists y \in \mathbb{Z}_n^* : y^2 = z\}$$

wobei $\mathbb{Z}_n := \mathbb{Z}/n\,\mathbb{Z}$ für $n \in \mathbb{N}$. Das Verschlüsseln ist nun das quadrieren einer Zahl mod $N$. Die Dechiffrierung ist das Quadratwurzel ziehen in $\mathbb{Z}_N$. Da der Empfänger die Faktorisierung von $N$ kennt, kann er mit Hilfe des chinesischen Restsatzes das Problem auf die simultane Quadratwurzelbestimmung in $\mathbb{Z}_P$ und $\mathbb{Z}_Q$ reduzieren. Wegen $P, Q \equiv 3 \mod 4$ gilt aber:

$$z^{(P+1)/4} = \sqrt{z} \mod P$$

und analog für $Q$. Mit Hilfe des schnellen Exponentierens lässt sich so die Quadratwurzel effizient bestimmen.

**Definition 6 (Levins universelle Einwegfunktion).**
Es ist $f_{\mathcal{U}}$ **Levins universelle Einwegfunktion** wie folgt definiert:

Die Eingabe wird geeignet zerteilt: $x = x_1 \cdots x_{\log n}$ ($n := |x|$) mit $|x_i| = \frac{n}{\log n}$ für $i = 1, \ldots, \log n$. Sei $(M_i)_{i \in \mathbb{N}}$ eine Abzählung aller Turingmaschinen. Dann gilt

$$f_{\mathcal{U}}(x) = M_1^{n^2}(x_1) \cdots M_{\log n}^{n^2}(x_{\log n})$$

wobei

$$M_i^t(x) = \begin{cases} M_i(x), & M_i \text{ terminiert nach } \leq t \text{ Schritten.} \\ 0^{|x|}, & \text{sonst} \end{cases}$$

**Satz 3.**
Falls es eine Einwegfunktion gibt, dann ist $f_{\mathcal{U}}$ eine Einwegfunktion.

Beweis
Falls eine Einwegfunktion $f : \mathbb{B}^* \to \mathbb{B}^*$ existiert, dann existiert auch eine Einwegfunktion $g$ mit Laufzeit $n^2$ für große $n$. Denn sei $p(n)$ ein Polynom, welches die Laufzeit von $f$ beschränkt, dann kann eine neue Funktion $g$ definiert werden, welche die ersten $\lceil n^2/p(n) \rceil n$ vielen Bits der Eingabe auf $f$ ausführt. Angenommen $g$ ist keine Einwegfunktion, dann kann auch $f$ durch Auffüllen von Bits in der Eingabe (der Teil der verworfen wird) effizient (Polynome sind unter Komposition abgeschlossen) invertiert werden.
Sei $i_0$ so dass $M_{i_0}$ so ein $g$ berechnet. Das heisst Einwegfunktion und $n^2$ Laufzeit. Betrachte genügend große $n$, sodass $i_0 < \lceil \log n \rceil$ (dann wird die Eingabe nicht aufgefüllt) und das invertieren von $M_{i_0}$ wird von jedem $A \in \mathcal{PPT}$ mit in $n/\log n$ vernachlässigbarer Wahrscheinlichkeit gefunden. Wegen

$$n/\log n < n^{-c} \Leftrightarrow n < n^{-c} \log n < n^{-c-\epsilon}$$

für alle $c > 0$ und geeignetem $\epsilon > 0$, sodass $-c - \epsilon > 0$ für alle $n \geq N$ mit geeignetem $N$. ∎

Auf Einwegfunktionen lässt sich 'berechnungssichere' Kryptographie aufbauen, es gilt:

**Satz 4.**
Wenn Einwegfunktionen existieren, dann gibt es ein $c \in \mathbb{N}$, sodass es ein berechnungssicheres Kryptosystem $(\mathbf{Enc}, \mathbf{Dec})$ gibt, welches eine Schlüssellänge von $n$ und Nachrichtenlänge von $n^c$ hat.

**Berechnungssicherheit** bedeutet für ein Kryptosystem $(\mathbf{Enc}, \mathbf{Dec})$ mit Schlüssellänge $n$ und Eingabelänge $m$, dass wir für alle $A \in \mathcal{PPT}$ eine vernachlässigebare Funktion $\epsilon$ haben, so dass ($x_i$ bezeichne das $i$-te Bit von der Nachricht $x$):

$$\mathop{\mathbf{Pr}}_{\substack{k \in_R \{0,1\}^n \\ x \in_R \{0,1\}^m}} [A(E_k(x)) = (i, b) \text{ s. d. } x_i = b] \geq \frac{1}{2} + \epsilon(n)$$

gilt. Das heisst über kein Bit der Nachricht $x$ kann ein Angreifer mit nicht vernachlässigbarer Wahrscheinlichkeit Informationen in Polynomialzeit errechnen.

# IV Pseudozufall

Um sichere Kryptosysteme konstruieren zu können, benötigt man bei der Schlüsselwahl den Zufall. Beim One-Time-Pad ist der Schlüssel so lang wie der Klartext - eine praktisch nicht immer günstige Lösung. Eine Idee ist es aus einem kleinen Schlüssel einen Strom an Pseudozufallszahlen zu erzeugen und dieses als Pad für das One-Time-Pad zu benutzen. Der Vorteil in dieser Konstruktion besteh darin, dass man einen viel kürzeren Schlüssel als beim One-Time-Pad benötigt. Um diese Pseudozufallszahlen zu erzeugen benutzt man Pseudozufallsgeneratoren, -Funktionen und -Permutationen. Diese Begriffe werden in diesem Abschnitt eingeführt und untersucht.

Was ist nun Zufall? **Kolmogorow** definiert eine Zeichenkette der Länge $n$ als zufällig, wenn keine Turingmaschine mit einer Codierungslänge von $< \frac{99}{100}n$ die Zeichenkette $n$ bei einer leeren Eingabe ausgibt. Dies führt zum Begriff der **Kolmogorow-Komplexität**. Leider ist die Frage i. A. unentscheidbar, daher für die Zwecke der Kryptographie ungeeignet.

Ein alternativer Ansatz stamm aus der Statistik. Dort müssen zufälligen Zeichenketten bzw. deren Teilmuster die Gesetze der Statistik genügen. Jedoch existiert eine Verteilung die diese Definition erfüllt, aber für Zwecke der Kryptographie ungeeignet ist.

Die Definition von Pseudozufall die der heutigen Kryptographie genügt ist: Eine Verteilung ist pseudozufällig, wenn sie nicht effizient vom wahren Zufall zuverlässig unterscheidbar ist. Genauer:

**Definition 7 (Pseudozufallsgenerator (PZG)).**
Seien $G : \mathbb{B}^* \to \mathbb{B}^*, \ell : \mathbb{N} \to \mathbb{N}$ in Polynomialzeit berechenbar und gelte $\ell(n) > n$ für alle $n \in \mathbb{N}$. $(G, \ell)$ heisst **Pseudozufallsgenerator** (kurz: **PZG**) mit Dehnung $\ell$, wenn gilt $|G(x)| = \ell(|x|)$ für alle $x \in \mathbb{B}^*$ und für alle $A \in \mathcal{PPT}$ existiert ein vernachlässigbares $\epsilon$, sodass

$$\left|\mathbf{Pr}\left[A(G(U_n)) = 1\right] - \mathbf{Pr}\left[A(U_{\ell(n)}) = 1\right]\right| < \epsilon(n) \ (n \in \mathbb{N})$$

gilt.

Häufig wird im folgendem auch nur $G$ als PZG bezeichnet. Es gilt der bemerkenswerte Satz:

**Satz 5 (Einwegfunktionen $\Rightarrow$ PZG).**
Existiert eine Einwegfunktion, dann existieren für alle Polynome $\ell$ mit $\ell(n) > n$ für $n \in \mathbb{N}$ ein PZG $(G, \ell)$.

Der Beweis findet sich im Katz u. Lindell (2008) oder siehe die Originalpublikation Astad u. a. (1999).

Im folgenden wird ein Lemma von Yao und das Goldreich-Levin-Theorem gezeigt um daraufhin eine Abschwächung des Satzes zu zeigen: Aus der Existenz von Einwegpermutationen folgt die Existenz von Pseudozufallsgeneratoren. Zuerst wird noch eine Charakterisierung für Pseudozufallsgeneratoren eingeführt.

**Definition 8.**
Sei $G : \mathbb{B}^* \to \mathbb{B}^*$ mit Dehnung $\ell$. $G$ und $\ell$ sind in Polynomialzeit berechenbar. $G$ heißt **unvorhersehbar**, wenn für alle $B \in \mathcal{PPT}$ gilt:

$$\underset{\substack{x\in_r\mathbb{B}^n\\ y=G(x)\\ i\in_R[\ell(n)]}}{\mathbf{Pr}} [B(1^n, y_1, \ldots, y_{i-1}) = y_i] \leq 1/2 + \epsilon(n) \ (n \in \mathbb{N})$$

mit vernachlässigbaren $\epsilon$.

Anschaulich bedeutet die Unvorhersehbarkeit von $G$, dass unter Vorlage der ersten $i-1$ Bits das nächste nicht effizient berechnet werden kann. Die Übergabe von $n$ 1 an $B$ gewährleistet, dass $B$ in $n$ polynomielle Laufzeit haben kann. Dies ist ein typischer technischer Trick. Das Ziel ist es nun die Äquivalenz der Begriffe des Pseudozufallsgenerators und der Unvorhersehrbarkeit zu beweisen.

---

**Lemma 4 (Yao)** Sei $(G, \ell)$ ein PZG, dann existieren für alle $A \in \mathcal{PPT}$ ein $B \in \mathcal{PPT}$, so dass für alle $n \in \mathbb{N}$ und $\epsilon > 0$ aus $\mathbf{Pr}[A(G(U_n))] - \mathbf{Pr}\left[A(U_{\ell(n))}\right] \geq \epsilon$, folgt

$$\underset{\substack{x\in_R\{0,1\}^n\\ y=G(x)\\ i\in_R[\ell(n)]}}{\mathbf{Pr}} [B(1^n, y_1, \ldots, y_{i-1}) = y_i] \geq 1/2 + \epsilon/\ell(n).$$

---

Beweis
Seien $A \in \mathcal{PPT}$, $\epsilon > 0$, $n \in \mathbb{N}$ und $\ell := \ell(n)$ und es gelte

$$\mathbf{Pr}[A(G(U_n))] - \mathbf{Pr}[A(U_\ell] \geq \epsilon,$$

das heisst $A$ gibt eher bei 'Pseudozufallszahlen' eine 1 aus. Es ist nun ein $B \in \mathcal{PPT}$ zu konstruieren.

**Algorithmus** $B$:
**Eingabe**: $1^n$, $i \in [\ell(n)]$ und $y_1, \ldots, y_{i-1} \in \mathbb{B}$.
**Vorgehen**: Wähle $z_i, \ldots, z_{\ell(n)} \in_R B$ und berechne

$$a = A(y_1, \ldots, y_{i-1}, z_i, \ldots, z_{\ell(n)}).$$

Falls $a = 1$ ist, dann gebe $z_i$ aus, sonst $1 - z_i$. $B$ fragt also quasi $A$ ob er $z_i$ richtig geraten hat.

**Analyse von** $B$: Setze für $i = 0, \ldots, \ell$

$$\mathcal{D}_i := \{y_1, \ldots, y_i, z_{i+1}, \ldots, z_\ell \mid z_j \in_R \mathbb{B},\ j = i+1, \ldots, \ell, x \in_R \mathbb{B}^n,\ y = G(x)\}$$

als Verteilungen welche die ersten $i$ Bits von $y \in \operatorname{im} G$ festlassen. Zum Beispiel ist $\mathcal{D}_0 = U_\ell$ und $\mathcal{D}_l = G(U_n)$. Mit $p_i := \mathbf{Pr}[A(\mathcal{D}_i) = 1]$ für $i = 0, \ldots, \ell$ gilt:

$$\epsilon \leq \mathbf{Pr}[A(\mathcal{D}_\ell)] - \mathbf{Pr}[A(\mathcal{D}_0)] = p_\ell - p_0 = \sum_{j=1}^{\ell} (p_j - p_{j-1})$$

und damit

$$\mathop{\mathbf{E}}_{i\in_R[\ell]}[p_i - p_{i-1}] \geq \frac{\epsilon}{\ell}. \tag{2}$$

$B$ sagt $y_i$ korrekt voraus wenn entweder $a = 1$ und $y_i = z_i$ oder $a \neq 1$ und $y_i = 1 - z_i$ ist. Damit ist die Wahrscheinlichkeit einer richtigen Antwort gleich

$$\mathop{\mathbf{Pr}}_{\substack{x\in_R\mathbb{B}^n\\ y=G(x)}}[B(1^n, y_1, \ldots, y_{i-1}) = y_i] = \frac{1}{2}\left(\mathbf{Pr}[a = 1 \mid z_i = y_i] + \mathbf{Pr}[a \neq 1 \mid y_i = 1 - z_i]\right) \tag{3}$$

Wenn $z_i = y_i$ ist, dann ist es als ob $B$ $A$ mit der Verteilung $\mathcal{D}_i$ aufgerufen hätte. Stellt man keine Bedingung an $z_i$, dann ist der Aufruf äquivalent zu $\mathcal{D}_{i-1}$. Damit lässt sich schreiben

$$\begin{aligned} p_{i-1} &= \mathbf{Pr}[a = 1] \\ &= \frac{1}{2}\left(\mathbf{Pr}[a = 1 \mid z_i = y_i] + \mathbf{Pr}[a = 1 \mid z_i = 1 - y_i]\right) \\ &= \frac{1}{2}\left(p_i + \mathbf{Pr}[a = 1 \mid z_i = 1 - y_i]\right) \end{aligned}$$

und damit wird 3 zu

$$\frac{1}{2}\left(p_i + 1 - \mathbf{Pr}[a = 1 \mid y_i = 1 - z_i]\right) = \frac{1}{2}(p_i + 1 + p_i - 2p_{i-1}) = 1/2 + (p_i - p_{i-1})$$

für $i \in [\ell]$. Man erhält durch Bilden des Erwartungswertes über die Wahlen von $i \in_R [\ell]$ dann mit 2

$$\begin{aligned} \mathop{\mathbf{Pr}}_{\substack{x\in_R\mathbb{B}^n\\ y=G(x)\\ i\in_R[\ell]}}[B(1^n, y_1, \ldots, y_{i-1}) = y_i] &= \mathop{\mathbf{E}}_{i\in_R[\ell]}\left[\mathop{\mathbf{Pr}}_{\substack{x\in_R\mathbb{B}^n\\ y=G(x)}}[B(1^n, y_1, \ldots, y_{i-1}) = y_i]\right] \\ &\geq \mathop{\mathbf{E}}_{i\in[\ell]}[1/2 + (p_i - p_{i-1})] \\ &\geq 1/2 + \epsilon/\ell \end{aligned}$$

■

Die Form der Argumentation in der von $D_{i-1}$ auf $D_i$ und insgesamt von $D_0$ auf $\mathcal{D}_l$ geschlossen wurde, heisst **Hybridargument**.

**Satz 6 ($G$ PZG $\Leftrightarrow$ unvorhersehbar).**
Seien $G : \mathbb{B}^* \to \mathbb{B}^*$ und $\ell : \mathbb{N} \to \mathbb{N}$ in Polynomialzeit berechenbar und $G$ habe die Dehnung $\ell$. Dann ist $(G, \ell)$ ein PZG genau dann wenn $G$ unvorhersehbar ist.

Beweis
"⇒": Angenommen $(G, \ell)$ ist PZG und $n \in \mathbb{N}$. Wenn $y = (y_1, \ldots, y_{\ell(n)}) \in_R \mathbb{B}^{\ell(n)}$ zufällig gleichverteilt gewählt wurde, kann kein Bit vorhergesagt werden. Ist $G$ vorhersehbar, dann

kann $y = G(x)$ von $y \in_R \mathbb{B}^{\ell(n)}$ unterschieden werden. Damit ist $G$ kein PZG.
"$\Leftarrow$": Angenommen $(G, \ell)$ ist kein PZG. Dann existiert ein $A \in \mathcal{PPT}$ mit

$$\mathbf{Pr}\left[A(G(U_n))\right] - \mathbf{Pr}\left[A(U_{\ell(n)})\right] \geq n^{-c}$$

für eine Konstante $c$ und $\infty$-vielen $n$. Die Betragsstriche in der Definition vom Pseudozufallsgenerator lassen sich ggf. durch Übergang von $A$ zu $1 - A$ entfernen. Mit dem Lemma von Yao gibt es für solche $n$ ein $B \in \mathcal{PPT}$ welches mit Wahrscheinlichkeit $\geq 1/2 + n^{-c}/\ell(n)$ ein Bit vorhersagen kann. Da $n^{-c}/\ell(n)$ nicht vernachlässigbar ist für große $n$ folgt, dass $G$ vorhersehbar ist. ∎

**Satz 7.**
Existiert eine Einwegpermutation, dann existiert ein PZG $G$ mit Dehnung $n + 1$.

Beweis
$G(x, r) := f(x), r, x^t r$ ist ein Pseudozufallsgenerator mit Dehnung $2n + 1$. Denn $G$ ist unvorhersehbar: die ersten $2n$ Bits von $G(U_{2n})$ sind zufällig unabhängig voneinander und das $2n + 1$ Bit kann wegen des Goldreich-Levin-Theorems nicht zuverlässig vorhergesagt werden. ∎

**Satz 8.**
Seien $a_1, \ldots, a_n \in [0, 1]$ und $\sum a_i/n = \rho$ der Durchschnitt. Dann gilt für mindestens $\rho/2$ der $a_i$, dass sie größer gleich $\rho/2$ sind.

Beweis
Sei $\gamma$ der Anteil der $i$ mit $a_i \geq \rho/2$. Dann gilt:

$$\begin{aligned}\rho &\leq \gamma \cdot 1 + (1 - \gamma)\frac{\rho}{2} \\ &\leq \gamma + \frac{\rho}{2} \\ &\Rightarrow \rho/2 \leq \gamma\end{aligned}$$

∎

**Satz 9 (Goldreich-Levin Theorem).**
Sei $f : \mathbb{B}^* \to \mathbb{B}^*$ eine Einwegpermutation. Dann gibt es für alle $A \in \mathcal{PPT}$ ein vernachlässigbares $\epsilon$ mit

$$\Pr_{\substack{x \in_R \mathbb{B}^n \\ r \in_R \mathbb{B}^n}}\left[A(f(x), r) = x^t \cdot r = \sum_{i=1}^{n} x_i r_i\right] \leq 1/2 + \epsilon(n)$$

Beweis
Angenommen es gibt $A \in \mathcal{PPT}$, $\epsilon > 0$ und $n \in \mathbb{N}$ mit

$$\mathop{\mathbf{Pr}}_{\substack{x \in_R \mathbb{B}^n \\ r \in_R \mathbb{B}^n}} \left[ A(f(x), r) = x^t \cdot r = \sum_{i=1}^{n} x_i r_i \right] \geq 1/2 + \epsilon$$

Im folgendem wird die Einwegpermutation mit Hilfe von $A$ invertiert.

Mindestens $\epsilon/2 \cdot 2^n$ der $x \in \mathbb{B}^n$ erfüllen $A(f(x), r) = x^t r$ mit einer Wahrscheinlichkeit (nach Wahl von $r$ und festem $x$) $\geq 1/2 + \epsilon/2$. Dies gilt wegen Satz 8. Diese $x$ heißen **gut** und sie werden zur Umkehrung der Einwegpermutation genutzt.

Es ist quasi eine schwarze Box gegeben, die $x \mapsto x^t r$ für $1/2 + \epsilon/2$ der Eingaben $r$ berechnet. Daraus soll in polynomieller Zeit (in $|x|$ und $1/\epsilon$) $x$ rekonstruiert werden.

Wenn $\mathbf{Pr}_{r \in_R \mathbb{B}^n} [A(f(x), r) = x^t r] = 1$, dann gilt

$$A(f(x), e^i) = x^t e^i = x_i$$

für $i = 1, \ldots, n$ und damit kann $x$ effizient bestimmt werden.

Sei nun die Wahrscheinlichkeit, dass $A(f(x), r) = x^t r$ bei $\frac{9}{10}$ für einen $\Omega(\epsilon)$ Anteil der $x$. Es gilt

$$\begin{aligned}
&\mathop{\mathbf{Pr}}_{r \in_R \mathbb{B}^n} \left[ A(f(x)), r) \neq x^t r \vee A(f(x), r \oplus e^i) \neq x^t (r \oplus e^i) \right] \\
\leq &\mathop{\mathbf{Pr}}_{r} \left[ A(f(x), r) \neq x^t r \right] + \mathop{\mathbf{Pr}}_{r} \left[ A(f(x), r \oplus e^i) \neq x^t (r \oplus e^i) \right] \\
\leq &\frac{2}{10}
\end{aligned}$$

Und es gilt mit einer Wahrscheinlichkeit von $\geq \frac{8}{10}$

$$A(f(x), r) \oplus A(f(x), r \oplus e^i) = x^t r \oplus x^t (r \oplus e^i) = x^t e^i = x_i$$

Dies lässt sich über eine **Majoritätswahl** noch weiter verbessern:
**Algorithmus** $B$:
1. Wähle $r^1, \ldots, r^m$ aus $U_{\mathbb{B}^n}$.
2. Für alle $i = 1, \ldots, n$: Rate $x_i$ nach der Majorität in $(A(f(x), r^j) \oplus A(f(x), r^j \oplus e^i))_{1 \leq j \leq m}$.

Behauptung: Für $m = 200n$ wird für alle $i \in [n]$ $B$ $x_i$ mit einer Wahrscheinlichkeit von mindestens $1 - \frac{1}{10n}$ richtig erraten und damit $x$ mit einer Wahrscheinlichkeit von $\geq \frac{9}{10}$ korrekt berechnet. Definiere für festes $i \in [n]$:

$$Z_j = \begin{cases} 0, & A(f(x)), r) \neq x^t r \vee A(f(x), r \oplus e^i \\ 1, & \text{sonst} \end{cases}$$

für $j = 1, \ldots, m$. Die $Z_j$ sind unabhängig und es wurde gezeigt $\mathbf{E}[Z_j] \geq \frac{8}{10}$. Setze $Z = \sum_{j=1}^{m} Z_j$. $Z$ zählt die Anzahl der falschen Berechnungen und es gilt $\mathbf{E}[Z] \geq 0.8m$ wegen der Linearität des Erwartungswertes. Ferner ist $\mathbf{Var}(Z) = \sum_{j=1}^{m} \mathbf{Var}(Z_j) \leq m$ und damit mit Tschebyschow

$$\mathbf{Pr}\left[ |Z - \mathbf{E}[Z]| \geq \frac{3m}{10} \right] \leq \frac{\mathbf{Var}[Z]}{\left(\frac{3m}{10}\right)^2} \leq \frac{9}{100m} = \frac{9}{100 \cdot 200n} < \frac{1}{10n}$$

Wenn $|Z - \mathbf{E}[Z]| \geq \frac{3m}{10}|$ ist, heisst dies nichts anderes als $Z \leq m/2$. In dem Fall wird die Mehrheitswahl ein falsches $x_i$ wählen. Insgesamt bedeutet dies, dass $B$ mit einer Wahrscheinlichkeit von $> 0.9$ richtig raten wird.

Dieser Beweis scheitert bei der Abschätzung von $\mathbf{E}[Z_j]$ wenn die Erfolgswahrscheinlichkeit von $A$ unter $3/4$ sinkt. Es kann nur garantiert werden, dass für gute $x$ die Wahrscheinlichkeit besser als $1/2 + \epsilon/2$ ist - dies könnte kleiner als $3/4$ sein. Selbst wenn die $r^i$ nur paarweise unabhängig sind gilt noch

$$\mathbf{Var}\left[\sum_j Z_j\right] = \sum_j \mathbf{Var}[Z_j]$$

dies kann dazu benutzt werden den allgemein Fall zu zeigen.

Wie kann $r^1, \ldots, r^m$ gewählt werden, so dass dies ausgenutzt wird? Sei $k$ (minimal) mit $m \leq 2^k - 1$:
1. Wähle $s^1, \ldots, s^k \in_R \mathbb{B}^n$.
2. Wähle $T_1, \ldots, T_m \subseteq [k]$ nichtleer und paarweise verschieden. Setze

$$r^j = \left[\sum_{t \in T_j} s^t\right]_2$$

Es kann gezeigt werden, dass die $r_j$ paarweise unabhängig sind. Für $x \in \mathbb{B}^n$ gilt

$$x^t \cdot r^j = \sum_{i \in T_j} x^t \cdot s^i$$

Das heisst aus $x^t s^1, \ldots, x^t s^k$ lassen sich $x^t r^1, \ldots, x^t r^m$ berechnen. Da $2^k = O(m)$ kann man alle möglichen Werte für $x^t s^1, \ldots, x^t s^k$ in polynomieller Zeit durchtesten. Genauer:
**Algorithmus** $B'$:
Eingabe: $y \in \mathbb{B}^n$ mit $y = f(x)$ für ein unbekanntes $x$. Dabei sind nur die Fälle interessant, wo $x$ gut ist.
Sei $m = 200\frac{n}{\epsilon^2}$ und $k$ minimal mit $m \leq 2^k - 1$. Wähle $s^1, \ldots, s^k \in_R \mathbb{B}^k$ und definiere $r^1, \ldots, r^m$ wie oben. Für alle $w \in \mathbb{B}^k$: Starte $B$ mit der Annahme $x \odot s^j = w_j$ für alle $j \in [k]$. Wenn $x = x_1, \ldots, x_n$ $f(x) = y$ erfüllt, dann halten und $x$ ausgeben.

Die Analyse geht wie vorher auch nur dass der Fall abgewartet werden muss, wo $x \odot s^j$ richtig geraten wird. ∎

**Satz 10 (PZGs mit polynomieller Dehnung).**
Sei $f$ eine Einwegpermutation, $c \in \mathbb{N}$ und $x, r \in \mathbb{B}^n$, setze:

$$G(x, r) := r, f(x)^t \cdot r, f^2(x)^t \cdot r, \ldots, f^l(x)^t \cdot r$$

mit $l = n^c$. Dann ist $G$ ein PZG mit Dehnung $l(2n) = n + n^c$.

Beweis
Wir führen einen Widerspruch zur Unvorhersehbarkeit des PZG. Sei also $A \in \mathcal{PPT}$, so dass für $x, r \in_R \mathbb{B}^n$ und $i \in_R [N]$ gilt:

$$\mathbf{Pr}\left[A(r, f(x)^t \cdot r, \ldots, f^{i-1}(x)^t \cdot r) = f^i(x)^t \cdot r\right] \geq \frac{1}{2} + \epsilon$$

Es wird nun ein $B \in \mathcal{PPT}$ konstruiert, welches $x^t \cdot r$ aus $f(x)$ und $r$ mit Wahrscheinlichkeit $\geq 1/2 + \epsilon$ berechnet im Widerspruch zum Goldreich-Levin-Theorem. $B$ bekommt die Eingabe $y := f(x)$ und $r$, wählt $i \in [N]$ und berechnet $f(y), \ldots, f^{l-i}(y)$ und gibt

$$a = A(r, y^t \cdot r, f(x)^t \cdot r, \ldots, f^{l-i-1}(y)^t \cdot r)$$

aus.

Da $f$ eine Permutation ist, ist es die gleiche Verteilung als wenn wir $x' \in_R \mathbb{B}^n$ und $x = f^i(x')$ wählen. $A$ sieht $f^i(x')^t \cdot r$ mit Wahrscheinlichkeit $\geq 1/2 + \epsilon$ vorraus und damit $B$ auch $x^t \cdot r$. $B$ trägt einfach das Problem an $A$ heran und bringt es in das Format von $A$. Damit kann $B$ mit nicht vernachlässigbarer Abweichung von $1/2$ im Widerspruch zum Goldreich-Levin-Theorem $x^t \cdot r$ berechnen. ∎

# V Zero-Knowledge Beweise

In mathematischen Beweisen von Aussagen wird mehr Information preisgegeben als nur die Wahrheit der bewiesenen Aussage. Es gibt Fälle in denen diese Preisgabe an zusätzlichen Informationen nicht gewollt ist, dies führt zum Begriff der Zero-Knowledge Beweise. Modelliert wird dies mit einer Interaktion zwischen einem Beweiser $P$ (für Prover) und Verifizier $V$.

Zur Authentizifizerung ist es interessant, die zur Authentifikation nötigen Informationen nicht preiszugeben, denn diese könnte abgefangen und z. B. wiedergegeben werden zur fälschlichen Authentifikation (Replay-Attacke). Bei einem Zero-Knowledge Beweis zur Authentifikation gibt es dieses Problem nicht mehr. Mathematisch definiert:

**Definition 9 (Zero-Knowledge Beweise).**
Sei $L \in \mathcal{NP}$ und $M$ eine Turingmaschine, die in Polynomialzeit läuft, mit

$$x \in L \Leftrightarrow \exists u \in \{0,1\}^{p(|x|)} : M(x, h) = 1. \ (p \text{ Polynom})$$

$M$ entscheidet also $L$ mit Hilfe eines Zeugen $u$.

Ein Paar $(P, V)$ von interaktiven Polynomialzeitalgorithmen heißt Zero-Knowledge Beweis für $L$, falls die folgenden Eigenschaften erfüllt sind: **Vollständigkeit** (Completeness): Für jedes $x \in L$ und Zertifikat $u = u(x)$ gilt

$$\mathbf{Pr}\left[\text{out}_V \langle P(x,), V(x)\rangle\right] \geq \frac{2}{3}$$

Wobei $\langle P(x,u), V(x)\rangle$ die Interaktion zwischen $P$ und $V$ mit den gegebenen Eingaben bezeichnet und $\text{out}_V I$ beschreibt die Ausgabe von $V$ am Ende der Interaktion $I$.

**Zuverlässigkeit** (Soundness): Wenn $x \notin L$, dann gilt für jede Strategie $P^*$ und Eingabe $u$, dass

$$\mathbf{Pr}\left[\mathrm{out}_V\langle P^*(x,u), V(x)\rangle\right] \leq \frac{1}{3}$$

dabei ist $P^*$ in keiner Weise beschränkt.

**Perfect-Zero-Knowledge-Eigenschaft**: Für alle Verifizierstrategien $V^* \in \mathcal{PPT}$ existiert ein $S^*$ mit erwarteter probabilistischer Polynomiallaufzeit, so dass für alle $x \in L$ und $u$ Zeuge dafür gilt:

$$\mathrm{out}_{V^*}\langle P(x,u), V^*(x)\rangle \equiv S^*(x)$$

Die Gleichheit bezieht sich auf die Gleichheit der Verteilungen. $S^*$ **simuliert** $V^*$.

Die letzte Eigenschaft gewährleistet, dass kein Verfizier Informationen erlangt, die er eh schon haben könnte z. B. durch Ausführen vom Simulator $S^*$.

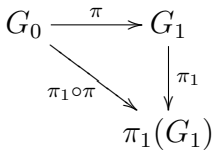
Beispiel (Zero-Knowledge Beweis für Graphenisomorphie (GI)):
Das Entscheidungsproblem der Graphenisomorphie ist es für Graphen $G_0$ und $G_1$ zu entscheiden, ob $G_0 \cong G_1$, d. h. ob es eine Bijektion $\Phi : V(G_0) \to V(G_1)$ gibt, so dass

$$vw \in E(G_0) \Leftrightarrow \Phi(v)\Phi(w) \in E(G_1)$$

oder anders formuliert, ob $V(G_0) = V(G_1)$ und ob eine Permutation (o. E. $V(G_0) = [n]$) $\pi : [n] \to [n]$ existiert, sodass $G_1 = \pi(G_0)$ gilt. Hierfür existiert ein Zero-Knowledge Beweis mit der Interaktion:
*Eingabe*: Graphen $G_0, G_1$ mit $V(G_i) = [n]$ in Adjazenzmatrixform gegeben.
*Eingabe von P*: $\pi : [n] \to [n]$ mit $G_1 = \pi(G_0)$.

*Interaktion*: $P$ wählt Permutation $\pi_1 \in_R S_n$ und sendet $V$ die Adjazenzmatrix von $\pi_1(G_1)$ dieser Graph soll $H$ heißen (dies wird inbesondere dann wichtig, wenn $P$ kein Isomorphismus kennt). $V$ wählt ein $b \in_R \{0,1\}$ zufällig und schickt es zu $P$. Nun antwortet $P$ mit $\pi_1$ falls $b = 1$ und sonst mit $\pi_1 \circ \pi$. Diese Antwort sei mit $\widetilde{\pi}$ bezeichnet.

Jetzt akzeptiert $V$ genau dann wenn $\pi_1(G_1) = \widetilde{\pi}(G_b)$. Als Bild




**Vollständigkeit**: Halten beide Parteien sich an das Protokoll, dann akzeptiert der Verifizier das Ergebnis mit Wahrscheinlichkeit 1.
**Zuverlässigkeit**: Wenn $G_0 \ncong G_1$, dann wird der Verfizierer mit einer Wahrscheinlichkeit von $\geq 1/2$ ablehnen. Denn ein $G_b$ wird nicht isomorph zu $H$ sein.
**Perfect-Zero-Knowledge-Eigenschaft**: Sei $V^*$ ein Verifizierer und $S^*$ der folgende Simulator:

Bei Eingabe von zwei Graphen $G_0, G_1$ $S^*$ wählt ein $b' \in_R \mathbb{B}$ und eine zufällige Permutation $\pi \in_R S_{[n]}$ und berechnet $H = \pi(G_{b'})$. $S^*$ sendet dann $H$ an den Verifizier und erhält $b \in \mathbb{B}$

zurück. Wenn $b = b'$ dann sendet $S^*$ $\pi$ and $V^*$ und gibt aus, was $V^*$ ausgibt. Wenn $b \neq b'$ dann startet $S^*$ einen neuen Anlauf.

Die erste Nachricht von $S^*$ ist identisch verteilt zu der von $P$: Beide verschicken einen zufälligen, zu $G_1$ (und $G_0$) isomorphen Graphen. Wenn $b' = b$, dann sieht $V^*$ die gleiche Interaktion wie in einer richtigen Interaktion mit $P$. Die Wahrscheinlichkeit davon ist $1/2$ also ist die Wahrscheinlichkeit, dass $k$ Runden gebraucht werden $2^{-k}$. Damit ist die erwartete Laufzeit

$$T(n) \sum_{k \geq 1} 2^{-k} = \mathcal{O}(T(n))$$

mit $T(n)$ Laufzeit von $V^*$.

# Literatur

[Arora u. Barak 2009] Arora, Sanjeev ; Barak, Boaz: *Computational Complexity: A Modern Approach*. New York, NY, USA : Cambridge University Press, 2009. – ISBN 0521424267, 9780521424264

[Astad u. a. 1999] Astad, Johan H. ; Impagliazzo, Russell ; Michael, Leonid A. L.: *A Pseudorandom Generator from any One-way Function*. http://citeseer.ist.psu.edu/326477.html;http://www.icsi.berkeley.edu/~luby/PAPERS/hill.ps. Version: April 07 1999

[Katz u. Lindell 2008] Katz, Jonathan ; Lindell, Yehuda: *Introduction to Modern Cryptography*. USA : Chapman and Hall/CRC, 2008. – ISBN 9781584885511

[Lenstra u. a. 1990] Lenstra, A. K. ; Lenstra, H. W. Jr. ; Manasse, M. S. ; Pollard, J. M.: The number field sieve. In: *STOC '90: Proceedings of the twenty-second annual ACM symposium on Theory of computing*. New York, NY, USA : ACM, 1990. – ISBN 0–89791–361–2, 0–89791–361–2, S. 564–572

[Shannon 1949] Shannon, C. E.: Communication Theory of Secrecy Systems. In: *Bell Systen Technicl Journal* 28 (1949), Oktober, S. 656–715