

Einführung in die Kryptologie

Johannes Köbler



Institut für Informatik
Humboldt-Universität zu Berlin

SS 2020

Informationstheoretische Sicherheit

- Claude E. Shannon untersuchte die Sicherheit kryptografischer Systeme auf informationstheoretischer Basis
- Seinen Untersuchungen liegt das Modell einer **Nachrichtenquelle** X zugrunde, die einzelne Klartextnachrichten x unter einer bestimmten Wahrscheinlichkeitsverteilung $p(x) = \Pr[X = x]$ generiert
- Zudem nehmen wir an, dass der Schlüssel $k \in K$ von einem **Schlüssel-generator** S unter einer Verteilung $p(k) = \Pr[S = k]$ erzeugt wird
- Da der Schlüssel unabhängig vom Klartext gewählt wird, ist $p(k, x) = p(k)p(x)$ die Wahrscheinlichkeit, dass X den Klartext x generiert und dieser mit dem Schlüssel k verschlüsselt wird
- Dabei gehen wir davon aus, dass für jede Nachricht $x \in M$ ein **neuer Schlüssel** gewählt wird
- Dies bedeutet, dass wir beispielsweise bei der additiven Chiffre den Klartextrraum auf $M = A^n$ vergrößern müssen, falls der Schlüssel nach n Zeichen gewechselt wird

Informationstheoretische Sicherheit

- Die Zufallsvariablen X und S induzieren eine Verteilung auf dem Kryptotextraum, die wir durch die Zufallsvariable Y beschreiben
- Die Wahrscheinlichkeit eines Kryptotextes y berechnet sich zu

$$p(y) = \Pr[Y = y] = \sum_{k,x:E(k,x)=y} p(k, x)$$

und für einen beobachteten Kryptotext y (mit $p(y) > 0$) ist

$$p(x|y) = \frac{p(x, y)}{p(y)} = \sum_{k:E(k,x)=y} \frac{p(k, x)}{p(y)}$$

die (bedingte) Wahrscheinlichkeit, dass sich hinter dem Kryptotext y der Klartext x verbirgt

- Da der Schlüsselgenerator für die Sicherheit eine wichtige Rolle spielt, nehmen wir bei Sicherheitsbetrachtungen die Schlüsselverteilung S als sechste Komponente eines Kryptosystems hinzu

Definition

Ein Kryptosystem $KS = (M, C, E, D, K, S)$ mit Schlüsselverteilung S heißt **informationstheoretisch** (oder **absolut**) **sicher**, falls jede Klartextverteilung X auf M unabhängig von der zugehörigen Kryptotextverteilung Y auf C ist

- Bei einem absolut sicheren Kryptosystem ist also die **A-posteriori-Wahrscheinlichkeit** $p(x|y)$ von x gleich der **A-priori-Wahrscheinlichkeit** $p(x)$
- Die Wahrscheinlichkeit von x bleibt somit gleich, ob nun der Kryptotext y bekannt ist oder nicht, d.h. die Kenntnis von y erlaubt keinerlei Rückschlüsse auf die gesendete Nachricht
- Dies bedeutet, dass es dem Gegner nicht möglich ist, das System zu brechen; auch nicht mit **unbegrenzten** Rechenressourcen
- Wie wir sehen werden, lässt sich dieses Maß an Sicherheit nur mit einem sehr hohen Aufwand erreichen

- Sind $p(x), p(y) > 0$, so gilt wegen

$$p(x|y)p(y) = p(x, y) = p(y|x)p(x)$$

die Gleichheit

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \quad (\text{Satz von Bayes})$$

- Daher ist die Bedingung $p(x) = p(x|y)$ äquivalent zu $p(y) = p(y|x)$, was wiederum mit der Unabhängigkeit der Ereignisse $X = x$ und $Y = y$ gleichbedeutend ist

Beispiel

- Sei $KS = (M, C, E, D, K)$ ein Kryptosystem mit $M = \{x_1, \dots, x_4\}$, $K = \{k_1, \dots, k_4\}$, $C = \{y_1, \dots, y_4\}$ und Verschlüsselungsfunktion

E	x_1	x_2	x_3	x_4
k_1	y_1	y_4	y_3	y_2
k_2	y_2	y_1	y_4	y_3
k_3	y_3	y_2	y_1	y_4
k_4	y_4	y_3	y_2	y_1

- Zudem betrachten wir folgende Schlüssel- und Klartextverteilung

k_i	k_1	k_2	k_3	k_4
$p(k_i)$	$1/2$	$1/4$	$1/8$	$1/8$

bzw.

x_i	x_1	x_2	x_3	x_4
$p(x_i)$	$1/2$	$1/6$	$1/6$	$1/6$

Beispiel

- Zudem betrachten wir folgende Schlüssel- und Klartextverteilung

k_i		k_1	k_2	k_3	k_4
$p(k_i)$		$1/2$	$1/4$	$1/8$	$1/8$

bzw.

x_i		x_1	x_2	x_3	x_4
$p(x_i)$		$1/2$	$1/6$	$1/6$	$1/6$

- Wegen $p(y) = \sum_{k,x:E(k,x)=y} p(k,x)$ ergibt sich die Kryptotextverteilung

$$p(y_1) = 1/2 \cdot 1/2 + (1/4 + 1/8 + 1/8) \cdot 1/6 = 1/3$$

$$p(y_2) = 1/4 \cdot 1/2 + (1/8 + 1/8 + 1/2) \cdot 1/6 = 1/4$$

$$p(y_3) = 1/8 \cdot 1/2 + (1/8 + 1/2 + 1/4) \cdot 1/6 = 5/24$$

$$p(y_4) = 1/8 \cdot 1/2 + (1/2 + 1/4 + 1/8) \cdot 1/6 = 5/24$$

- Die bedingten Wahrscheinlichkeiten $p(x|y_1)$ berechnen sich wie folgt:

$$p(x_1|y_1) = p(k_1, x_1)/p(y_1) = p(k_1)p(x_1)/p(y_1) = 1/2 \cdot 1/2 \cdot 3 = 3/4$$

$$p(x_2|y_1) = p(k_2, x_2)/p(y_1) = p(k_2)p(x_2)/p(y_1) = 1/4 \cdot 1/6 \cdot 3 = 1/8$$

$$p(x_3|y_1) = p(k_3, x_3)/p(y_1) = p(k_3)p(x_3)/p(y_1) = 1/8 \cdot 1/6 \cdot 3 = 1/16$$

$$p(x_4|y_1) = p(k_4, x_4)/p(y_1) = p(k_4)p(x_4)/p(y_1) = 1/8 \cdot 1/6 \cdot 3 = 1/16$$

- Wegen $p(x_1) = 1/2 \neq 3/4 = p(x_1|y_1)$ ist das System nicht absolut sicher \triangleleft

Frage

Lässt sich das Kryptosystem KS aus obigem Beispiel durch Verwendung eines anderen Schlüsselgenerators absolut sicher machen?

- KS ist genau dann absolut sicher, wenn $p(y_j) = p(y_j|x_i)$ für alle $(x_i, y_j) \in M \times C$ gilt
- Da es in KS für jedes Paar (x_i, y_j) genau einen Schlüssel $k = k_{i,j} \in K$ mit $E(k, x_i) = y_j$ gibt, gilt $p(y_j|x_i) = p(k_{i,j})$
- Somit gilt die Äquivalenz $p(y_j) = p(y_j|x_i) \Leftrightarrow p(y_j) = p(k_{i,j})$
- Für $j = 1$ und $i = 1, \dots, 4$ implizieren die Gleichungen $p(y_1) = p(k_{i,1})$, dass alle vier Schlüssel $k_{i,1} (= k_i)$ gleichwahrscheinlich sein müssen
- Wegen

$$p(y_j) = \sum_{i=1}^4 p(x_i)p(y_j|x_i) = 1/4 \sum_{i=1}^4 p(x_i) = 1/4 = p(k_{i,j}) = p(y_j|x_i)$$

ist das System in diesem Fall tatsächlich absolut sicher

In Verallgemeinerung dieses Beispiels lässt sich für eine wichtige Klasse von Kryptosystemen die absolute Sicherheit wie folgt charakterisieren

Satz

- Sei $KS = (M, C, E, D, K, S)$ ein Kryptosystem mit $\|M\| = \|C\| = \|K\|$, dessen Schlüsselraum K für jedes Paar $(x, y) \in M \times C$ genau einen Schlüssel k mit $E(k, x) = y$ enthält
- Dann ist KS genau dann absolut sicher, wenn S auf K gleichverteilt ist

Beweis.

- Bezeichne $k_{x,y}$ den eindeutigen Schlüssel, der den Klartext x auf den Kryptotext y abbildet
- Falls S auf K gleichverteilt ist, folgt wegen $p(k_{x,y}) = \|K\|^{-1}$ für alle x, y mit $p(x) > 0$ zunächst

$$p(y|x) = \sum_{k:E(k,x)=y} p(k) = p(k_{x,y}) = \|K\|^{-1}$$

und

$$p(y) = \sum_{x,p(x)>0} p(x)p(y|x) = \|K\|^{-1} \sum_x p(x) = \|K\|^{-1}$$

- Somit folgt $p(x, y) = p(x)p(y|x) = p(x)p(y)$, d.h. KS ist absolut sicher
- Die Umkehrung wird in den Übungen gezeigt □

Ist der One-Time-Pad absolut sicher

- Verwendet man beim One-Time-Pad nur Klartexte einer festen Länge, so ist dieser nach obigem Satz genau dann absolut sicher, wenn die Schlüssel gleichwahrscheinlich sind
- Variiert die Klartextlänge, so kann ein Gegner aus y nur die Länge des zugehörigen Klartextes x ableiten
- Wird jedoch derselbe Schlüssel k zweimal verwendet, so kann aus den Kryptotexten die Differenz der zugehörigen Klartexte ermittelt werden:

$$\left. \begin{array}{l} y_1 = E(x_1, k) = x_1 + k \\ y_2 = E(x_2, k) = x_2 + k \end{array} \right\} \rightsquigarrow y_1 - y_2 = x_1 - x_2$$

- Sind die Klartexte natürlichsprachig, so können aus $y_1 - y_2$ die beiden Nachrichten x_1 und x_2 ähnlich wie bei der Analyse einer Lauftextverschlüsselung rekonstruiert werden

- In einem absolut sicheren Kryptosystem muss der Schlüsselraum mindestens so groß wie der Klartextrraum sein (siehe Übungen)
- Daher erfordert die absolute Sicherheit einen extrem hohen Aufwand
- Vor der Kommunikation muss ein Schlüssel, der mindestens so lang wie der zu übertragende Klartext ist, zufällig generiert und zwischen den Partnern auf einem sicheren Kanal ausgetauscht werden
- Für die meisten Anwendungen ist jedoch keine absolute Sicherheit erforderlich
- Wie wir bei der Betrachtung von Stromsystemen gesehen haben, kann der Schlüsselstrom auch von einem **Pseudo-Zufallsgenerator** erzeugt werden
- Dieser erhält als Eingabe eine Zufallszahl s_0 (den sogenannten **Keim**) und erzeugt daraus eine lange Folge $v_0 v_1 \dots$ von Pseudo-Zufallszahlen
- Als Schlüssel muss jetzt nur noch der Keim ausgetauscht werden

Ein Maß für den Informationsgehalt

- In der Informationstheorie wird die Unsicherheit über eine Nachrichtenquelle X nach ihrer Entropie bemessen
- Dabei entspricht die Unsicherheit über X genau dem Informationsgewinn, der sich aus der Beobachtung der Quelle X ergibt
- Intuitiv ist die in einer einzelnen Nachricht x steckende Information umso größer, desto unwahrscheinlicher sie ist
- Tritt eine Nachricht x mit der Wahrscheinlichkeit $p(x) = \Pr[X = x] > 0$ auf, dann ist ihr **Informationsgehalt** definiert als

$$\mathit{Inf}_X(x) = \log_2(1/p(x)) = -\log_2 p(x)$$

- Im Fall $p(x) = 0$ sei $\mathit{Inf}_X(x) = 0$

Ein Maß für den Informationsgehalt

- Diese Definition des Informationsgehalts ergibt sich zwangsläufig aus den beiden folgenden Axiomen:
 - Der (gemeinsame) Informationsgehalt $Inf_{X,Y}(x,y)$ von zwei Nachrichten x und y , die aus unabhängigen Quellen X und Y stammen, ist $Inf_X(x) + Inf_Y(y)$
 - Eine Nachricht x , die mit Wahrscheinlichkeit $\Pr[X = x] = 1/2$ auftritt, hat den Informationsgehalt $Inf_X(x) = 1$
- Die Einheit des Informationsgehalts ist bit (basic indissoluble information unit)
- Die Entropie von X ist nun der erwartete Informationsgehalt einer von X generierten Nachricht

Der Entropiebegriff

Definition

- Sei X eine Zufallsvariable mit Wertebereich $W(X) = \{x_1, \dots, x_n\}$ und sei $p_i = \Pr[X = x_i]$
- Dann ist die **Entropie** von X definiert als

$$\mathcal{H}(X) = \sum_{i=1}^n p_i \ln f_X(x_i) = \sum_{i=1}^n p_i \log_2(1/p_i) = - \sum_{i=1}^n p_i \log_2(p_i)$$

Beispiel

- Sei X eine Zufallsvariable mit der Verteilung

x_i	sonnig	leicht bewölkt	bewölkt	stark bewölkt	Regen	Schnee	Nebel
p_i	1/4	1/4	1/8	1/8	1/8	1/16	1/16

- Dann ergibt sich die Entropie von X zu

$$\mathcal{H}(X) = 1/4 \cdot (2 + 2) + 1/8 \cdot (3 + 3 + 3) + 1/16 \cdot (4 + 4) = 2,625 \quad \triangleleft$$

Der Entropiebegriff

- Die Entropie nimmt im Fall der Gleichverteilung $p_1 = \dots = p_n = 1/n$ den Wert $\log_2(n)$ an, während sie für jede andere Verteilung auf einer n -elementigen Menge einen Wert $\mathcal{H}(X) < \log_2(n)$ hat (siehe unten)
- Die Unsicherheit über eine Zufallsvariable X ist um so größer, je größer der Wertebereich und je gleichmäßiger die Verteilung von X ist
- Bringt X zum Beispiel nur einen einzigen Wert mit positiver Wahrscheinlichkeit hervor, dann (und nur dann) hat $\mathcal{H}(X)$ den Wert 0
- Für den Nachweis von oberen Schranken für die Entropie benutzen wir folgende Hilfsmittel aus der Analysis

Definition

- Sei $I \subseteq \mathbb{R}$ ein Intervall. Eine Funktion $f : I \rightarrow \mathbb{R}$ heißt **konkav** auf I , falls für alle $x \neq y \in I$ und $0 \leq t \leq 1$ gilt:

$$f(tx + (1-t)y) \geq tf(x) + (1-t)f(y)$$

- Gilt sogar „ $>$ “ anstelle von „ \geq “, so heißt f **streng konkav** auf I

Der Entropiebegriff

Beispiel

Die Funktion $f(x) = \log_2(x)$ ist streng konkav auf $(0, \infty)$



Für den Beweis des nächsten Satzes benötigen wir die **Jensensche Ungleichung**, die wir ohne Beweis angeben

Satz

- Sei f eine streng konkave Funktion auf I und seien $0 < a_1, \dots, a_n < 1$ reelle Zahlen mit $\sum_{i=1}^n a_i = 1$
- Dann gilt für alle $x_1, \dots, x_n \in I$,

$$f\left(\sum_{i=1}^n a_i x_i\right) \geq \sum_{i=1}^n a_i f(x_i).$$

- Hierbei tritt Gleichheit genau dann ein, wenn alle x_i den gleichen Wert haben

Satz

- Sei X eine Zufallsvariable auf einer n -elementigen Menge $\{x_1, \dots, x_n\}$ mit der Verteilung $p_i = \Pr[X=x_i]$ für $i = 1, \dots, n$
- Dann gilt $H(X) \leq \log_2(n)$, wobei Gleichheit genau im Fall $p_i = 1/n$ für $i = 1, \dots, n$ eintritt

Beweis.

- Aufgrund der Jensenschen Ungleichung gilt

$$\mathcal{H}(X) = \sum_{i=1}^n p_i \log_2(1/p_i) \leq \log_2 \sum_{i=1}^n (p_i/p_i) = \log_2 n,$$

wobei Gleichheit genau im Fall $1/p_1 = \dots = 1/p_n$ eintritt

- Letzteres ist mit der Bedingung $p_i = 1/n$ für $i = 1, \dots, n$ gleichbedeutend



- Die Entropie liefert eine sehr gute untere Schranke für die mittlere Codewortlänge von Binärcodes
- Ein **Binärcode** für X ist eine (geordnete) Menge $C = \{y_1, \dots, y_n\}$ von binären Codewörtern y_i für die Nachrichten x_i mit der Eigenschaft, dass die Abbildung $c : W(X)^* \rightarrow \{0, 1\}^*$ mit $c(x_{i_1} \cdots x_{i_k}) = y_{i_1} \cdots y_{i_k}$ injektiv ist
- Die Injektivität von c stellt sicher, dass jede Folge $y_{i_1} \cdots y_{i_k}$ von Codewörtern eindeutig decodierbar ist
- Die **mittlere Codewortlänge** von C unter X ist

$$L(C) = \sum_{i=1}^n p_i \cdot |y_i|$$

- Ein Binärcode C heißt **optimal**, wenn kein anderer Binärcode für X eine kürzere mittlere Codewortlänge besitzt
- Für einen optimalen Binärcode C für X gilt (ohne Beweis)

$$\mathcal{H}(X) < L(C) < \mathcal{H}(X) + 1$$

Beispiel

- Sei X die Zufallsvariable aus dem letzten Beispiel mit der Verteilung

i	1	2	3	4	5	6	7
p_i	$1/4$	$1/4$	$1/8$	$1/8$	$1/8$	$1/16$	$1/16$

- Betrachte die Codes $C = \{y_1, \dots, y_7\}$ und $C' = \{y'_1, \dots, y'_7\}$ mit

i	1	2	3	4	5	6	7
y_i	001	010	011	100	101	110	111
y'_i	00	01	100	101	110	1110	1111

- Dann hat C die mittlere Codewortlänge $L(C) = 3$
- Dies ist nicht optimal, während der Code C' wegen

$$|y'_i| = \log_2(1/p_i) \text{ für } i = 1, \dots, 7$$

den minimalen Wert $L(C') = \mathcal{H}(X) = 2,625$ erreicht

- Die Redundanz eines Codes für X ist um so höher, je größer seine mittlere Codewortlänge im Vergleich zur Entropie von X ist
- Um auch Codes über unterschiedlichen Alphabeten vergleichen zu können, sollte die Codewortlänge das Alphabet berücksichtigen
- Hierzu definiert man die **Bitlänge** eines Wortes x über einem Alphabet A mit $m > 2$ Buchstaben zu $|x|_2 = |x| \log_2(m)$
- Beispielsweise ist die Bitlänge von GOLD (über dem lateinischen Alphabet) $|\text{GOLD}|_2 = 4 \log_2(26) = 18,8$
- Entsprechend berechnet sich für einen Code $C = \{y_1, \dots, y_n\}$ unter einer Verteilung p_1, \dots, p_n die mittlere Codewortlänge (in bit) zu

$$L_2(C) = \sum_{i=1}^n p_i \cdot |y_i|_2$$

- Damit können wir die Redundanz eines Codes als den mittleren Anteil der Codewortbuchstaben definieren, die keine Information tragen

Definition

Die **(relative) Redundanz** eines Codes C für X ist definiert als

$$\mathcal{R}(C) = \frac{L_2(C) - \mathcal{H}(X)}{L_2(C)}.$$

Beispiel (Fortsetzung)

- Während eine von X generierte Nachricht im Durchschnitt $\mathcal{H}(X) = 2,625$ bit an Information enthält, haben alle Codewörter von C die Bitlänge $L_2(C) = 3$
- Der Anteil an „überflüssigen“ Zeichen pro Codewort beträgt also

$$\mathcal{R}(C) = \frac{3 - 2,625}{3} = 12,5\%,$$

wogegen C' wegen $L_2(C') = \mathcal{H}(X)$ keine Redundanz besitzt

Redundanz von Sprachen

- Auch Schriftsprachen wie Deutsch oder Englisch und Programmiersprachen wie C können als eine Art Code aufgefasst werden
- Eine solche Sprache enthält umso mehr Redundanz, desto restriktiver die Gesetze sind, unter denen in ihr Worte und Sätze gebildet werden
- Um die statistischen Eigenschaften einer Sprache L zu erforschen, erweist es sich als zweckmäßig, die Textstücke der Länge n (n -Gramme) von L für unterschiedliche n getrennt voneinander zu betrachten
- Sei also L_n die Zufallsvariable, die die Verteilung aller n -Gramme in L beschreibt
- Interpretieren wir diese n -Gramme als Codewörter einer festen Codewortlänge n , so ist

$$\mathcal{R}(L_n) = \frac{n \log_2 m - \mathcal{H}(L_n)}{n \log_2 m}$$

die Redundanz dieses Codes

Definition

- Für eine Sprache L über einem Alphabet A mit $\|A\| = m$ ist $\mathcal{H}(L_n)/n$ die **n -Gramm-Entropie** von L (pro Buchstabe)
- Falls dieser Wert für n gegen ∞ von oben gegen einen Grenzwert

$$\mathcal{H}(L) = \lim_{n \rightarrow \infty} \mathcal{H}(L_n)/n$$

konvergiert, so wird dieser Grenzwert als die **Entropie von L** bezeichnet

- In diesem Fall konvergiert $\mathcal{R}(L_n)$ von unten gegen den Grenzwert

$$\mathcal{R}(L) = \lim_{n \rightarrow \infty} \mathcal{R}(L_n) = \frac{\log_2 m - \mathcal{H}(L)}{\log_2 m},$$

der als die **(relative) Redundanz** von L bezeichnet wird

- Der Zähler $\mathcal{R}_{abs}(L) = \log_2 m - \mathcal{H}(L) = \mathcal{R}(L) \log_2 m$ wird auch als **absolute Redundanz** von L bezeichnet (gemessen in bit/Zeichen)

Beispiel

- Die Redundanz von (natürlichen) Sprachen lässt sich näherungsweise bestimmen, indem man die Entropien $\mathcal{H}(L_n)$ ihrer n -Gramme empirisch ermittelt
- Im Deutschen hat die Einzelzeichenverteilung eine Entropie von $\mathcal{H}(L_1) = 4,1$ bit
- Im Vergleich hierzu hat eine auf A_{lat} gleichverteilte Zufallsvariable U den maximalen Entropiewert von $\mathcal{H}(U) = \log(26) = 4,7$ bit
- Für die Bigramme ergibt sich ein Entropiewert von $\mathcal{H}(L_2)/2 = 3,5$ bit pro Buchstabe

Beispiel (Fortsetzung)

- Mit wachsender Länge sinkt die Entropie weiter ab und strebt gegen einen Grenzwert $\mathcal{H}(L)$ von 1,5 bit pro Buchstabe

n	$\mathcal{H}(L_n)$	$\mathcal{H}(L_n)/n$	$\mathcal{R}_{abs}(L_n)/n$	$\mathcal{R}(L_n)$
1	4,1	4,1	0,6	13%
2	7,0	3,5	1,2	26%
3	9,6	3,2	1,5	32%
6	12,2	2,0	2,7	57%
15	27,6	1,8	2,9	62%
\vdots	\vdots	\vdots	\vdots	\vdots
∞	∞	$\mathcal{H}(L) = 1,5$	$\mathcal{R}_{abs}(L) = 3,2$	$\mathcal{R}(L) = 67\%$

- Deutsche Texte hinreichender Länge besitzen also eine durchschnittliche Redundanz von ca. 67%, so dass ihre Länge bei optimaler Kodierung auf ca. 1/3 komprimierbar ist

Die Eindeutigkeitsdistanz eines Kryptosystems

- Wir betrachten nun den Fall, dass Klartexte einer variablen Länge n verschlüsselt werden, ohne den Schlüssel zu wechseln
- Die Chiffrierfunktion hat also die Form

$$E_n : K \times A^n \rightarrow C_n$$

- Dabei nehmen wir an, dass die Menge C_n der zugehörigen Kryptotexte die gleiche Kardinalität $\|C_n\| = \|A^n\| = m^n$ wie der Klartextrraum hat
- Ist y ein abgefangener Kryptotext, so ist

$$K(y) = \{k \in K \mid \exists x \in A^n : E_n(k, x) = y \wedge p(x) > 0\}$$

die Menge aller infrage kommenden Schlüssel

- Die Menge $K(y)$ besteht aus einem „echten“ Schlüssel (mit dem y erzeugt wurde) und $\|K(y)\| - 1$ so genannten „unechten“ Schlüsseln

- Aus informationstheoretischer Sicht ist das Kryptosystem unter einer Klartextverteilung X umso sicherer, desto größer die erwartete Anzahl

$$\bar{s}_n = \sum_{y \in C_n} p(y) \cdot (\|K(y)\| - 1) = \sum_{y \in C_n} p(y) \cdot \|K(y)\| - 1$$

der unechten Schlüssel ist

- Im besten Fall kommen für jeden Kryptotext alle Schlüssel infrage, d.h. $\bar{s}_n = \|K\| - 1$
- Ist dagegen \bar{s}_n gleich 0, so liefert der abgefangene Kryptotext dem Gegner genügend Information, um den benutzten Schlüssel und somit den zugehörigen Klartext eindeutig bestimmen zu können (sofern er über genügend Ressourcen verfügt)

Die Eindeutigkeitsdistanz

Definition

Die **Eindeutigkeitsdistanz** n_0 eines Kryptosystems unter einer Klartextverteilung X ist der kleinste Wert von n , für den $\bar{s}_n = 0$ wird

- Als nächstes wollen wir eine untere Schranke für \bar{s}_n (und damit für n_0) herleiten
- Hierzu benötigen wir den Begriff der bedingten Entropie $\mathcal{H}(X|Y)$ von X , wenn der Wert von Y bereits bekannt ist

Definition. Seien X, Y Zufallsvariablen

Die **bedingte Entropie** von X unter Y ist definiert als

$$\mathcal{H}(X|Y) = \sum_{y \in W(Y)} p(y) \mathcal{H}(X|y) = \sum_y p(y) \sum_x p(x|y) \log_2(1/p(x|y)),$$

wobei $X|y$ die Zufallsvariable mit der Verteilung $p_y(x) = p(x|y)$ ist

Satz. Es gilt

- 1 $\mathcal{H}(X, Y) = \mathcal{H}(Y) + \mathcal{H}(X|Y)$ und
- 2 $\mathcal{H}(X, Y) \leq \mathcal{H}(X) + \mathcal{H}(Y)$, wobei Gleichheit genau dann eintritt, wenn X und Y unabhängig sind

(Beweis siehe Übungen)

Korollar

Es gilt $\mathcal{H}(X|Y) \leq \mathcal{H}(X)$, wobei Gleichheit genau dann eintritt, wenn X und Y unabhängig sind.

Satz

In jedem Kryptosystem gilt für die Klartextentropie $\mathcal{H}(X)$, die Schlüsselentropie $\mathcal{H}(S)$ und die Kryptotextentropie $\mathcal{H}(Y)$ die Gleichung

$$\mathcal{H}(S|Y) = \mathcal{H}(S) + \mathcal{H}(X) - \mathcal{H}(Y)$$

Satz

In jedem Kryptosystem gilt die Gleichung

$$\mathcal{H}(S|Y) = \mathcal{H}(S) + \mathcal{H}(X) - \mathcal{H}(Y)$$

Beweis.

- Zunächst ist $\mathcal{H}(S|Y) = \mathcal{H}(S, Y) - \mathcal{H}(Y)$
- Es reicht also zu zeigen, dass $\mathcal{H}(S, Y) = \mathcal{H}(S) + \mathcal{H}(X)$ ist
- Bei Kenntnis des Schlüssels ist der Wert von X eindeutig durch Y und der Wert von Y eindeutig durch X festgelegt
- Da X und S unabhängig sind, folgt daher

$$\begin{aligned}\mathcal{H}(S, Y) &= \mathcal{H}(S, X, Y) - \underbrace{\mathcal{H}(X|S, Y)}_{=0} = \mathcal{H}(S, X) + \underbrace{\mathcal{H}(Y|S, X)}_{=0} \\ &= \mathcal{H}(S) + \mathcal{H}(X)\end{aligned}$$



Jetzt verfügen wir über alle Hilfsmittel, um die erwartete Anzahl

$$\bar{s}_n = \sum_{y \in \mathcal{C}_n} p(y) \cdot \|K(y)\| - 1$$

der unechten Schlüssel nach unten abschätzen zu können

Lemma

Seien X_n und Y_n die Zufallsvariablen, die die Verteilungen der n -Gramme der Klartextsprache und der zugehörigen Kryptotexte beschreiben.

Dann gilt

- $\mathcal{H}(S|Y_n) \leq \log_2(\bar{s}_n + 1)$
- $\mathcal{H}(S|Y_n) \geq \mathcal{H}(S) - n\mathcal{R}(L_n) \log_2 m$

Lemma. Es gilt

- $\mathcal{H}(S|Y_n) \leq \log_2(\bar{s}_n + 1)$
- $\mathcal{H}(S|Y_n) \geq \mathcal{H}(S) - n\mathcal{R}(L_n) \log_2 m$

Beweis.

- Unter Verwendung der Jensenschen Ungleichung folgt

$$\begin{aligned}\mathcal{H}(S|Y_n) &= \sum_{y \in \mathcal{C}_n} p(y) \cdot \mathcal{H}(S|y) \\ &\leq \sum_{y \in \mathcal{C}_n} p(y) \cdot \log_2 \|K(y)\| \\ &\leq \log_2 \sum_{y \in \mathcal{C}_n} p(y) \cdot \|K(y)\| \\ &= \log_2(\bar{s}_n + 1)\end{aligned}$$

Lemma. Es gilt

- $\mathcal{H}(S|Y_n) \leq \log_2(\bar{s}_n + 1)$
- $\mathcal{H}(S|Y_n) \geq \mathcal{H}(S) - n\mathcal{R}(L_n) \log_2 m$

Beweis (Fortsetzung)

- Wir wissen bereits dass $\mathcal{H}(S|Y_n) = \mathcal{H}(S) + \mathcal{H}(X_n) - \mathcal{H}(Y_n)$ ist
- Wegen $X_n = L_n$ folgt $\mathcal{H}(X_n) = (1 - \mathcal{R}(L_n))n \log_2 m$, wobei $m = \|A\|$ ist
- Wegen $W(Y_n) = C_n$ und $\|C_n\| = m^n$ gilt für die Kryptotextentropie

$$\mathcal{H}(Y_n) \leq n \log_2 m$$

- Somit folgt

$$\mathcal{H}(S|Y_n) = \mathcal{H}(S) + \mathcal{H}(X_n) - \mathcal{H}(Y_n) \geq \mathcal{H}(S) - n\mathcal{R}(L_n) \log_2 m \quad \square$$

Die Eindeutigkeitsdistanz

- Zusammen ergibt sich also

$$\log_2(\bar{s}_n + 1) \geq \mathcal{H}(S) - n\mathcal{R}(L_n) \log_2 m \geq \mathcal{H}(S) - n\mathcal{R}(L) \log_2 m$$

- Im Fall eines gleichverteilten Schlüssels erreicht $\mathcal{H}(S)$ den maximalen Wert $\log_2 \|K\|$, was auf die gesuchte Abschätzung für \bar{s}_n führt

Satz

Werden mit einem Kryptosystem (M, C, E, D, K, S) mit $M = A^n$ und $\|C\| = m^n$ Klartexte einer Sprache L der festen Länge n mit gleichverteiltem Schlüssel $k \in K$ verschlüsselt, so gilt für die erwartete Anzahl \bar{s}_n der unechten Schlüssel,

$$\bar{s}_n \geq \frac{\|K\|}{m^{n\mathcal{R}(L_n)}} - 1$$

Die Eindeutigkeitsdistanz

Setzen wir in obiger Abschätzung $\bar{s}_n = 0$, so erhalten wir folgende untere Schranke für die Eindeutigkeitsdistanz n_0 eines Kryptosystems

Korollar

Unter den Bedingungen des obigen Satzes gilt

$$n_0 \geq \frac{\log_2 \|K\|}{\mathcal{R}(L_n) \log_2 m} \geq \frac{\log_2 \|K\|}{\mathcal{R}(L) \log_2 m} = \frac{\log_2 \|K\|}{\mathcal{R}_{abs}(L)}$$

- Man beachte, dass wir die Mindestmenge an Kryptotext zur eindeutigen Bestimmung des Schlüssels nur nach unten abgeschätzt haben und die tatsächlich benötigte Menge deutlich größer sein kann
- Natürlich erlaubt die eindeutige Bestimmung des Schlüssels auch die eindeutige Bestimmung des Klartexts
- Unter Umständen kann der Klartext aber auch schon mit einer wesentlich geringeren Menge an Kryptotext rekonstruierbar sein

Beispiel (Untere Schranken für n_0 bei klassischen Chiffren)

Für Substitutionen bei deutschsprachigem Klartext ergeben sich folgende Werte als untere Schranke für die Eindeutigkeitsdistanz n_0 :

Kryptosystem	Schlüsselanzahl $\ K\ $	$\log_2 \ K\ $	$\log_2 \ K\ / \mathcal{R}_{abs}(L)$
additive Chiffre	26	4.7	$\frac{4.7}{3.2} \approx 1.5$
affine Chiffre	$12 \cdot 26 = 312$	8.3	2.6
einfache Substitution	$26!$	88.4	27.6
Vigenère-Chiffre	26^d	$4.7 \cdot d$	$1.5 \cdot d$

Dagegen erhalten wir für Blocktranspositionen folgende unteren Schranken für die Mindestmenge an Kryptotext, die zur eindeutigen Schlüsselbestimmung benötigt wird

Die Eindeutigkeitsdistanz

Beispiel (Untere Schranken für n_0 bei Blocktranspositionen)

Untere Schranken für n_0 bei einer Häufigkeitsanalyse auf der Basis von		Blocklänge ℓ				
		10	20	50	100	1 000
Einzelzeichen	$(\mathcal{R}(L_1) = 0, 6)$	59	165	578	1415	22 986
Bigrammen	$(\mathcal{R}(L_2) = 1, 2)$	40	111	390	954	15 502
Trigrammen	$(\mathcal{R}(L_3) = 1, 5)$	24	65	226	553	9 473
n -Grammen, $n \rightarrow \infty$	$(\mathcal{R}(L) = 3, 2)$	7	19	67	164	2 665

- Hierbei unterscheiden wir zusätzlich nach der Länge der bei der Häufigkeitsanalyse benutzten n -Gramme
- Dies entspricht der Situation, dass die Wahrscheinlichkeit jedes Zeichens im Klartext höchstens von den $n - 1$ vorausgehenden bzw. nachfolgenden Zeichen abhängt

- Da die Benutzung eines informationstheoretisch sicheren Kryptosystems einen immensen Aufwand erfordert, begnügt man sich in der Praxis meist mit schwächeren Sicherheitsanforderungen
- Ein Kryptosystem gilt als **komplexitätstheoretisch sicher** oder als **berechnungssicher** (*computationally secure*), falls es höchstens mit einem unverhältnismäßig hohen Aufwand zu brechen ist
- Die Kosten für einen erfolgreichen Angriff (sofern er überhaupt möglich ist) übersteigen also den potentiellen Nutzen bei weitem
- Ein Kryptosystem gilt als **nachweisbar sicher**, wenn seine Sicherheit auf der Basis von bekannten komplexitätstheoretischen Annahmen bewiesen werden kann, die gemeinhin als zutreffend gelten
- Als **praktisch sicher** wird ein Kryptosystem eingestuft, wenn es über mehrere Jahre allen Versuchen einer erfolgreichen Kryptoanalyse widerstehen konnte, obwohl es weit verbreitet ist und allein schon deshalb ein attraktives Ziel für einen Angriff darstellt

- Die komplexitätstheoretische Analyse eines Kryptosystems ist äußerst schwierig, da der Aufwand für einen erfolgreichen Angriff unabhängig von der dabei benutzten Technik abgeschätzt werden muss
- Es reicht also nicht, alle bekannten kryptoanalytischen Ansätze in Betracht zu ziehen, sondern alle **möglichen**
- Dabei kommt als Angriffsziel nicht nur die vollständige Rekonstruktion des Klartextes in Betracht, da bereits kleine Unterschiede zwischen A-posteriori- und A-priori-Wahrscheinlichkeit einen lohnenswerten Vorteil für den Angreifer bedeuten könnten
- Aus diesen Gründen ist noch für kein praktikables Kryptosystem der Nachweis gelungen, dass es komplexitätstheoretisch sicher ist
- Damit ist auch nicht zu rechnen, solange fundamentale komplexitätstheoretische Fragen wie etwa das **$P \stackrel{?}{=} NP$ -Problem** nicht gelöst sind
- Dagegen gibt es eine ganze Reihe praktikabler Kryptosysteme, die als nachweisbar sicher oder praktisch sicher gelten

Sicherheit gegen einen IND-CPA Angriff

- Eine mögliche Präzisierung der komplexitätstheoretischen Sicherheit ist die **Ununterscheidbarkeit unter einem gewählten Klartextangriff** (kurz **IND-CPA**; **indistinguishability under a chosen-plaintext attack**)
- Konkret läuft ein **IND-CPA-Angriff** wie folgt ab
 - ① Zuerst wählt der Gegner zwei Klartexte $x_0 \neq x_1 \in M$
 - ② Dann wird x_0 oder x_1 zufällig ausgewählt und der zugehörige Kryptotext y gebildet
 - ③ Dem Gegner wird der Kryptotext y vorgelegt und er muss raten, welcher der beiden Klartexte sich hinter y verbirgt
 - ④ Der Angriff ist erfolgreich, falls der Gegner richtig rät
- Um die Erfolgsaussichten des Gegners zu formalisieren, stellen wir das gewünschte Maß an Sicherheit über einen Parameter $s \in \mathbb{N}$ ein
- Typischerweise wird für s die Schlüssellänge $s = |k|$ gewählt
- Aus Praktikabilitätsgründen sollten legale Operationen (wie die Chiffrierung oder die Schlüsselgenerierung) effizient (in Zeit $s^{O(1)}$) ablaufen
- Natürlich darf dann auch der Aufwand des Gegners in Abhängigkeit von s steigen, weshalb er zusätzlich den Parameterwert s erhält

Ein Maß für den Erfolg eines IND-CPA Angriffs

Definition

- Sei (M, C, E, D, K, S) ein Kryptosystem mit **Sicherheitsparameter** s
- Ein (**IND-CPA-*Gegner***) ist ein Tripel $G = (X_0, X_1, V)$ von probabilistischen Algorithmen, wobei
 - X_0, X_1 bei Eingabe $s \in \mathbb{N}$ zwei Klartexte aus M generieren und
 - V bei Eingabe $s \in \mathbb{N}$ sowie $x_0, x_1 \in M$ und $y \in C$ ein Bit $V(s, x_0, x_1, y) \in \{0, 1\}$ ausgibt
- Der **Vorteil** von G bei Parameterwert $s \in \mathbb{N}$ ist

$$\alpha_G(s) = 2 (\Pr[V(X_0(s), X_1(s), E(S, X_B(s))) = B] - 1/2),$$

wobei B auf $\{0, 1\}$ gleichverteilt und von S, X_0, X_1, V unabhängig ist

- Ist der Wert des Sicherheitsparameters s irrelevant, fest vorgegeben oder aus dem Kontext ersichtlich, so verzichten wir meist auf seine explizite Angabe

Beispiel

- Im **ECB-Modus** (electronic code book mode) werden Klartextblöcke a_1, a_2, \dots unabhängig voneinander zu einer Folge b_1, b_2, \dots von Kryptotextblöcken $b_i = E(k, a_i)$ verschlüsselt
- In diesem Fall kann ein Gegner wie folgt einen Vorteil von 1 erzielen (d.h. mit Wahrscheinlichkeit 1 den richtigen Klartext raten)
- Er wählt (deterministisch) zwei beliebige Klartexte $x_0 = a_1 a_2 \dots$ und $x_1 = a'_1 a'_2 \dots$ mit der Eigenschaft $a_1 = a_2$ und $a'_1 \neq a'_2$
- Dann kann er bei Vorlage eines Kryptotextes $y = b_1 b_2 \dots$ leicht erkennen, aus welchem Klartext y generiert wurde:

$$V(x_0, x_1, y) = \begin{cases} 0, & b_1 = b_2 \\ 1, & \text{sonst} \end{cases}$$

Erwartungsgemäß sind absolut sichere Kryptosysteme gegen IND-CPA-Angriffe resistent

Satz

Der maximale Vorteil eines IND-CPA-Gegners gegenüber einem absolut sicheren Kryptosystem ist gleich Null

- Ein IND-CPA-Gegner kann also bei einem absolut sicheren Kryptosystem höchstens mit Wahrscheinlichkeit $\frac{1}{2}$ den richtigen Klartext raten, auch wenn er über unbeschränkte Rechenressourcen verfügt
- In den Übungen wird auch die umgekehrte Implikation bewiesen
- Ein Kryptosystem ist somit genau dann absolut sicher, wenn kein Gegner einen Vorteil größer als 0 erzielen kann

Beweis.

- Sei (X_0, X_1, V) ein beliebiger Gegner und für $b = 0, 1$ sei $Y_b = E(S, X_b)$
- Da bei einem absolut sicheren Kryptosystem der Kryptotext $Y = E(S, X)$ und der Klartext X unabhängig sind, sind auch Y_B und B unabhängig
- Daher folgt

$$\begin{aligned} & \Pr[V(X_0, X_1, Y_B) = B] \\ &= \Pr[V(X_0, X_1, Y_B) = B = 0] + \Pr[V(X_0, X_1, Y_B) = B = 1] \\ &= \Pr[V(X_0, X_1, Y_B) = 0] \cdot \underbrace{\Pr[B = 0 \mid V(X_0, X_1, Y_B) = 0]}_{= \Pr[B=0] = 1/2} \\ &\quad + \Pr[V(X_0, X_1, Y_B) = 1] \cdot \underbrace{\Pr[B = 1 \mid V(X_0, X_1, Y_B) = 1]}_{= \Pr[B=1] = 1/2} \\ &= 1/2 \end{aligned}$$



- Um zu einer formalen Definition zu kommen, müssen wir nun folgende Fragen beantworten:
 - ① Über welche Rechenressourcen verfügt ein Gegner realistischweise?
 - ② Wie groß darf der vom Gegner erzielbare Vorteil höchstens sein, ohne die Vertraulichkeit der verschlüsselten Nachricht zu verletzen?
- Bezüglich Frage 1 geht man typischerweise davon aus, dass der Gegner über probabilistische Schaltkreise polynomieller Größe verfügt (siehe nächste Folie)
- Bezüglich Frage 2 verlangt man, dass der Gegner für jedes Polynom p höchstens für endlich viele Parameterwerte s einen Vorteil größer gleich $1/p(s)$ erzielen darf
- Andernfalls wäre die Sicherheit gefährdet, da er für jedes solche s nach polynomiell vielen Wiederholungen der probabilistischen Berechnung von $V(s, x_0, x_1, y)$ fast sicher den richtigen Klartext ausfindig machen könnte, indem er das mehrheitlich berechnete Bit ausgibt

Definition

- Ein **boolescher Schaltkreis** der Größe m mit n Eingängen x_1, \dots, x_n und l Ausgängen $i_1, \dots, i_l \in [m]$ ist eine Folge $c = (g_1, \dots, g_m)$ von **Gattern**

$$g_l \in \{0, 1, x_1, \dots, x_n, (\neg, j), (\wedge, j, k), (\vee, j, k)\} \text{ mit } 1 \leq j, k < l$$

- Der von c bei Eingabe $a \in \{0, 1\}^n$ am Gatter g_l berechnete Wert $g_l(a) \in \{0, 1\}$ ist induktiv wie folgt definiert:

g_l	0	1	x_i	(\neg, j)	(\wedge, j, k)	(\vee, j, k)
$g_l(a)$	0	1	a_i	$1 - g_j(a)$	$g_j(a)g_k(a)$	$g_j(a) + g_k(a) - g_j(a)g_k(a)$

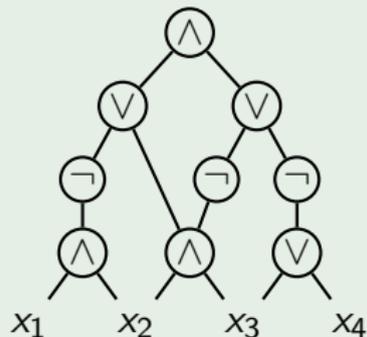
- Die **Ausgabe** von c bei Eingabe $a \in \{0, 1\}^n$ ist die Bitfolge $c(a) = g_{i_1}(a) \dots g_{i_l}(a)$

Beispiel

Der Schaltkreis

$$c = (x_1, x_2, x_3, x_4, (\wedge, 1, 2), (\wedge, 2, 3), \\ (\vee, 3, 4), (\neg, 5), (\neg, 6), (\neg, 7), \\ (\vee, 6, 8), (\vee, 9, 10), (\wedge, 11, 12))$$

mit den Eingängen x_1, x_2, x_3, x_4 und Ausgängen $(11, 12, 13)$ gibt bei Eingabe $a = 0110$ die Bitfolge $c(0110) = 100$ aus



- Ein **probabilistischer Schaltkreis** c hat neben den regulären Eingabegattern x_1, \dots, x_n noch eine Reihe von Zufallsgattern z_1, \dots, z_m
- Hierbei werden die Eingabegatter x_i wie bisher mit den Bits a_i eines Eingabevektors $a = a_1 \dots a_n \in \{0, 1\}^n$ belegt, während die m Zufallsgatter unabhängig gleichverteilte Bits Z_1, \dots, Z_m erzeugen (d.h. es gilt $\Pr[Z_1 \dots Z_m = b] = 2^{-m}$ für alle $b \in \{0, 1\}^m$)
- Dadurch wird die Ausgabe $c(a, Z_1, \dots, Z_m)$ zu einer Zufallsvariablen, die wir auch kurz mit $C(a)$ bezeichnen

Definition

Sei KS ein Kryptosystem mit variablem Sicherheitsparameter $s \in \mathbb{N}$

- Eine Funktion $\varepsilon : \mathbb{N} \rightarrow \mathbb{R}$ heißt **vernachlässigbar**, wenn für jedes Polynom p eine Zahl $n_0 \in \mathbb{N}$ existiert, so dass $\varepsilon(n) < 1/p(n)$ für alle $n \geq n_0$ gilt
- Ein Gegner $G = (X_0, X_1, V)$ heißt **effizient**, wenn probabilistische Schaltkreise c und c' der Größe $s^{O(1)}$ mit $C(s) = (X_0(s), X_1(s))$ und $C'(s, x_0, x_1, y) = V(s, x_0, x_1, y)$ existieren, wobei die Ein- und Ausgaben von c und c' binär kodiert sind
- KS heißt **komplexitätstheoretisch sicher**, wenn jeder effiziente Gegner G nur einen vernachlässigbaren Vorteil erzielen kann (d.h. die Funktion $\alpha_G(s)$ ist vernachlässigbar)