

Kryptographie Shannons Theorie

Benjamin R. Andres

2. November 2004

Gliederung

1. Begriffsklärung: Sicherheit
2. Unkonditionelle Sicherheit und ihre Erreichbarkeit
3. Absolute Geheimhaltung
4. Entropie
5. Produkte von Kryptosystemen

1. Begriffsklärung: Sicherheit

Es wird zwischen drei Kategorien der Sicherheit unterschieden.

Berechenbare Sicherheit

Beschreibt wieviel Rechenleistung mindestens aufgewendet werden muss um das Kryptosystem zu knacken.

- Leider nur für jeweils bestimmte Angriffe berechenbar.

Beweisbare Sicherheit

Das Kryptosystem wird auf ein bekanntes (NP-vollständiges) Problem zurückgeführt und ist somit ebenso sicher wie die Nicht-Lösbarkeit des Problems.

Unkonditionelle Sicherheit

Das Kryptosystem kann nicht geknackt werden, ganz gleich welche Methoden der Angreifer nutzt und wieviel Rechenleistung ihm zur Verfügung steht.

2. Unkonditionelle Sicherheit und ihre Erreichbarkeit

- Wie können wir der unkonditionellen Sicherheit möglichst nahe kommen?
- Was kann uns der Geheimtext über den Klartext verraten?

Definition eines Kryptosystems

Annahme: Bei einem *Kryptosystem* $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$, wobei

\mathcal{P} - Klartext (Plaintext)

\mathcal{C} - Geheimtext (Ciphertext)

\mathcal{K} - Schlüssel (Key)

\mathcal{E} - Verschlüsselungsfunktion (Encrypt Function)

\mathcal{D} - Entschlüsselungsfunktion (Decrypt Function)

wird ein Schlüssel $k \in \mathcal{K}$ für genau eine Verschlüsselung verwendet. Ferner soll der Schlüssel unabhängig vom Klartext gewählt worden sein.

Wobei $p \in \mathcal{P}$, $c \in \mathcal{C}$, $k \in \mathcal{K}$, $e_k(x) \in \mathcal{E}$ und $d_k(x) \in \mathcal{D}$ sein soll.

Berechnung der Wahrscheinlichkeit von C

Die Wahrscheinlichkeit mit dem man einen bestimmten Geheimtext erhält berechnet sich mit:

$$Pr(c) = \sum_{\{k: c \in C(k)\}} Pr(k) * Pr(p = d_k(c)).$$

Wenn der Klartext schon bekannt ist mit:

$$Pr(c|p) = \sum_{\{k: p = d_k(c)\}} Pr(k).$$

Dank Bayes Theorem

$$Pr(x|y) = \frac{Pr(x) * Pr(y|x)}{Pr(y)},$$

kann die Wahrscheinlichkeit für einen Klartext berechnet werden wenn der Geheimtext bekannt ist:

$$Pr(p|c) = \frac{Pr(p) * \sum_{\{k:p=d_k(c)\}} Pr(k)}{\sum_{\{k:c \in C(k)\}} Pr(k) * Pr(p = d_k(c))}.$$

3. Absolute Geheimhaltung

Definition:

Ein Kryptosystem hat eine *absolute Geheimhaltung* (perfect secrecy) genau dann, wenn $Pr(p|c) = Pr(p)$ für alle $p \in \mathcal{P}$ und $c \in \mathcal{C}$ gilt.

Die Wahrscheinlichkeit des Klartextes und des Geheimtextes sollen voneinander unabhängig sein.

Theorem:

Es sei $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ ein Kryptosystem in dem $|\mathcal{K}| = |\mathcal{C}| = |\mathcal{P}|$ gilt. Das Kryptosystem hat genau dann absolute Geheimhaltung, wenn jeder mögliche Schlüssel mit der gleichen Wahrscheinlichkeit $\frac{1}{|\mathcal{K}|}$ benutzt wird. Und wenn es für jedes $p \in \mathcal{P}$, $c \in \mathcal{C}$ genau einen Schlüssel k gibt, so dass $e_k(p) = c$.

One-Time Pad:

- 1917 von Gilbert Vernam erstmals beschrieben,
- Galt lange als unknackbar, aber es fehlte ein Beweis,
- Beweis durch das Konzept der perfekten Geheimhaltung von Shannon,
- Verwendung beim Militär, aber nur mäßigen Erfolg in der Wirtschaft.

Es sei $1 \leq n$, $n \in \mathbb{N}$ und $\mathcal{P} = \mathcal{C} = \mathcal{K} = (\mathbb{Z}_2)^n$. Für $k \in (\mathbb{Z}_2)^n$ sei $e_k(p)$ als die Vektorsumme modulo 2 von k und p definiert. Wenn also $p = (p_1, \dots, p_n)$ und $k = (k_1, \dots, k_n)$ ist, dann gilt:

$$e_k(p) = (p_1 + k_1, \dots, p_n + k_n) \text{ mod } 2.$$

Wobei die Dekodierung identisch mit der Codierung ist:

$$d_k(c) = (c_1 + k_1, \dots, c_n + k_n) \text{ mod } 2.$$

4. Entropie

- Was passiert wenn ein Schlüssel mehrmals verwendet wird?
- Entropie wird 1948 von Shannon als Begriff eingeführt,
- mathematisches Mittel zur Bestimmung von Information oder Ungenauigkeit.

Definition:

Angenommen x sei eine beliebige Variable, die mit einer bestimmten Wahrscheinlichkeit aus der Menge \mathcal{X} gewählt wird. Dann lässt sich die *Entropie* der Variable x mit

$$H(x) = - \sum_{x \in \mathcal{X}} Pr(x) \log_2 Pr(x)$$

berechnen.

Eigenschaften der Entropie

Theorem:

Angenommen x ist eine beliebige Variable, die mit einer Wahrscheinlichkeit von p_1, \dots, p_n bestimmte Werte annimmt, wobei $p_i > 0$ und $1 \leq i \leq n$.

Dann ist $H(x) \leq \log_2 n$, mit Gleichheit genau dann, wenn $p_i = \frac{1}{n}$ mit $1 \leq i \leq n$.

Theorem:

$$H(x, y) \leq H(x) + H(y),$$

mit Gleichheit genau dann, wenn x und y unabhängig sind.

Corollary:

$$H(x|y) \leq H(x),$$

mit Gleichheit genau dann, wenn x und y unabhängig sind.

Theorem:

Es sei $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ ein Kryptosystem, dann ist

$$H(K|C) = H(K) + H(\mathcal{P}) - H(\mathcal{C}).$$

Definition:

x und y seien zwei beliebige Variablen. Dann erhalten wir für jedes fest gewählte $y \in \mathcal{Y}$ die konditionelle Wahrscheinlichkeit für x . Es gilt:

$$H(x|y) = - \sum_x Pr(x|y) * \log_2 Pr(x|y).$$

Die *konditionelle Entropie* $H(x|y)$ sei als der gewichtete Durchschnitt aller möglichen Entropien $H(x|y)$ definiert:

$$H(x|y) = - \sum_y \sum_x Pr(y) * Pr(x|y) * \log_2 Pr(x|y).$$

Wir wollen nun unser Wissen auf ein Kryptosystem anwenden von dem wir wissen, dass der Klartext in englischer Sprache geschrieben und der gleiche Schlüssel mehrmals verwendet worden ist.

Spurious keys:

Wenn der gleiche Schlüssel mehrmals verwendet worden ist, dann fällt es leicht bestimmte Schlüssel als nicht mögliche Schlüssel auszusortieren. Alle anderen, mit Ausnahme des richtigen, werden *spurious keys* genannt.

Nun gibt es eine Formel mit der wir die Anzahl der *spurious keys* abschätzen können.

Theorem:

Angenommen $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ ist ein Kryptosystem mit $|\mathcal{C}| = |\mathcal{P}|$ und es besteht gleiche Wahrscheinlichkeit für alle Schlüssel. R_L sei die Redundanz der genutzten Sprache. Dann kann mit einem Geheimtext der Länge n die zu erwartende Anzahl der spurious keys mit

$$s_n \geq \frac{|\mathcal{K}|}{|\mathcal{P}|^{n \cdot R_L}} - 1$$

berechnet werden. Diese Formel können wir nach n umstellen und $s_n = 0$ setzen:

$$n_0 \approx \frac{\log_2 * |\mathcal{K}|}{R_L * \log_2 * |\mathcal{P}|}$$

Berechnen wir nun also die Redundanz der englischen Sprache.

Definition:

Angenommen L sei eine Sprache, dann ist die *Entropie* von L definiert als

$$H_L = \lim_{n \rightarrow \infty} \frac{H(\mathcal{P}^n)}{n}$$

und die *Renundanz* von L als

$$R_L = 1 - \frac{H_L}{\log_2 * |\mathcal{P}|}$$

Für große n hat man H_L auf $1,0 \leq H_L \leq 1,5$ berechnet, und damit die Renundanz der englischen Sprache auf ca. $\frac{3}{4}$.

Berechnen wir die von uns benötigte Länge des Geheimtextes wenn der Substitution Cipher ($|\mathcal{P}| = 26$ und $|\mathcal{K}| = 26!$) benutzt wurde:

$$n_0 \approx \frac{88,4}{0,75 * 4,7} \approx 25.$$

Daraus folgt, dass eine eindeutige Entschlüsselung möglich ist, wenn wir einen Geheimtext mit einer Länge von mindestens 25 Zeichen haben.

5. Produkte von Kryptosystemen

Idee:

- Die Sicherheit zu erhöhen, indem zwei Kryptosysteme miteinander verknüpft werden.
- Die Idee ist heute noch von fundamentaler Bedeutung.
- Findet z.B. im Advanced Encryption Standard Verwendung.

Anwendung:

Wir haben zwei endomorphe $(C = \mathcal{P})$ Kryptosysteme $S_1 = (\mathcal{P}, \mathcal{K}_1, \mathcal{E}_1, \mathcal{D}_1)$ und $S_2 = (\mathcal{P}, \mathcal{P}, \mathcal{K}_2, \mathcal{E}_2, \mathcal{D}_2)$, welche die gleiche Klartextmenge besitzen. Das Produkt der Kryptosysteme $S_1 * S_2$ ist dann definiert als $(\mathcal{P}, \mathcal{P}, \mathcal{K}_1 * \mathcal{K}_2, \mathcal{E}, \mathcal{D})$. Wobei $k = k_1 * k_2 = (k_1, k_2)$, mit $k_1 \in \mathcal{K}_1, k_2 \in \mathcal{K}_2$ und

$$e_{(k_1, k_2)}(p) = e_{k_2}(e_{k_1}(p))$$

bzw.

$$d_{(k_1, k_2)}(c) = d_{k_1}(d_{k_2}(c)).$$

Sicherheit:

$$Pr((k_1, k_2)) = Pr(k_1) * Pr(k_2).$$

Kommutativität:

Zwei Kryptosysteme kommutieren genau dann, wenn $S_1 * S_2 = S_2 * S_1$ gilt.

Assoziativität:

Es gilt immer $(S_1 * S_2) * S_3 = S_1 * (S_2 * S_3)$

Idempotente Kryptosysteme:

Ein Kryptosystem ist genau dann *idempotent*, wenn $S^2 = S * S = S$ gilt.

Anmerkung:

Wenn S_1 und S_2 idempotent sind und kommutieren, dann ist auch $(S_1 * S_2)$ idempotent:

$$\begin{aligned}(S_1 * S_2) * (S_1 * S_2) &= S_1 * (S_2 * S_1) * S_2 = S_1 * (S_1 * S_2) * S_2 \\ &= (S_1 * S_1) * (S_2 * S_2) = S_1 * S_2.\end{aligned}$$

Daraus folgt, dass wenn wir $(S_1 * S_2)$ nicht-idempotent haben wollen (z.B. für den Data Encryption Standard Cipher), S_1 und S_2 aber idempotent sind, dann dürfen S_1 und S_2 nicht kommutieren.