Einführung in die Kryptographie
3. Vorlesung

11. 5. 2006

- Sicherheit
- Affine Chiffren
- Moderne Blockchiffren

Sicherheit eines Kryptosystems

- Perfekte Geheimhaltung ist in der Regel eine unrealistische Forderung.

- Die Sicherheit der meisten modernen Kryptosysteme ist nicht beweisbar.

- Man beurteilt ihre Sicherheit nach allgemeinen Regeln, z.B.:

1. Regel

Die Sicherheit eines K.-Systems beruht alleine darauf, dass der Angreifer den geheimen Schlüssel nicht kennt.

Konsequenzen:

- Annahme: der Angreifer kennt das Kryptosystem und die Wahrscheinlichkeitsverteilung von Klartexten und Schlüsseln.

- Die Wahl des geheimen Schlüssels erfolgt zufällig.

Problem: Was heisst *zufällig*?

2. Regel

Die Sicherheit eines K.-Systems beruht auf der Schwierigkeit eines wohldefinierten mathematischen Problems.

Beispiele:

- Faktorisieren (RSA)
- Diskreter Logarithmus

3. Regel

Ein Kryptosystem sollte sicher gegen alle bekannten Angriffe sein.

Typen von Angriffen:

- Ciphertext-Only
- Known-Plaintext
- Chosen-Plaintext
- Chosen-Ciphertext

Ciphertext-Only-Angriff

Der Angreifer kennt nur das Kryptosystem und den Chiffretext.

Er versucht, den Klartext und/oder den Schlüssel zu ermitteln.

Beispiele:

- Angriff durch Häufigkeitsanalyse bei Substitutions- oder Vigenère-Chiffre.
- Vollständige Suche im Schlüsselraum.

Known-Plaintext-Angriff

Der Angreifer kennt andere Klartexte und die zugehörigen Chiffretexte.

- Klartexte enthalten oft vorhersehbare Sätze/Wörter.
- Derselbe Schlüssel wird mehrmals verwendet (Abhilfe: *Randomisierung*).

Chosen-Plaintext-Angriff

Der Angreifer kann Chiffretexte zu selbst gewählten Klartexten erzeugen.

Chosen-Plaintext-Angriff

Dies ist z.B. bei einem Verfahren mit öffentlichem Schlüssel immer möglich:

- Alice verschlüsselt Klartext p mit Bobs öffentlichem Schlüssel e:
$$c := E_e(p).$$
- Bob entschlüsselt Chiffretext c mit seinem geheimen Schlüssel d:
$$p = D_d(c).$$
- Eve kennt c und e. Sie kann also beliebige Klartexte p' verschlüsseln:
$$c' := E_e(p').$$

Chosen-Ciphertext-Angriff

Der Angreifer kann selbst gewählte Chiffretexte entschlüsseln.

Chosen-Ciphertext-Angriff

Dies ist z.B. möglich, wenn ein Verschlüsselungsverfahren zur *Identifizierung* verwendet wird:

- Alice und Bob kennen geheimen Schlüssel k
- Bob möchte Alice gegenüber beweisen, dass er k kennt
- Alice verschlüsselt Klartext p, $c = E_k(p)$, und sendet c an Bob
- Bob entschlüsselt c, $p' = D_k(c)$, und sendet p' an Alice
- Gilt $p = p'$, so weiss Alice, dass Bob k kennt
- Eve gibt sich gegenüber Alice als Bob aus und kann sich so beliebige Chiffretexte c' entschüsseln lassen

4. Regel

Der beste Angriff sollte die vollständige Suche im Schüsselraum sein.

Beispiel: DES

- symmetrische Blockchiffre
- 64-Bit-Klartextblöcke, 56-Bit-Schüssel
- Von 1976 bis 1997 US-Verschlüsselungsstandard
- Kann mittlerweile von speziellen Rechnern durch *brute force* gebrochen werden

Beispiel: affine Chiffren

Definition: Eine Blockchiffre mit Blocklänge n und

$$\mathcal{P} = \mathcal{C} = (\mathbb{Z}/m)^n,$$

$m \geq 2$, heisst *affin linear* falls für alle $k \in \mathcal{K}$ gilt:

$$E_k : \begin{cases} (\mathbb{Z}/m)^n & \to & (\mathbb{Z}/m)^n, \\ v & \mapsto & A_k \cdot v + b_k \pmod{m}, \end{cases}$$

mit $A \in \mathrm{GL}_n(\mathbb{Z}/m)$ und $b \in (\mathbb{Z}/m)^n$.

In der Praxis oft $m = 2$.

Affine Chiffren

Beispiele:

- Vigenère-Chiffre: $m = 26$ und $A = I_n$:

$$v \mapsto v + b \pmod{26}$$

- *Permutationschiffre*: $b = 0$, A ist eine Permutationsmatrix, z.B.

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix},$$

entspricht der Permutation

$$\sigma = (\,1\,4\,2\,3\,).$$

Affine Chiffren

Vorteile:

- Leicht zu implementieren.
- Sind A, b hinreichend allgemein und zufällig, so ist die Chiffre sicher gegen Angriff durch Häufigkeitsanalyse.

Nachteile:

- Grosser Schlüsselraum
- Kann durch Known-Plaintext-Angriff gebrochen werden (benötigt $n+1$ Plaintext-Chiffretext-Paare).

Moderne Blockchiffren

Anforderungen (Shannon, 1949):

- **Konfusion**: Der funktionale Zusammenhang zwischen Klartext, Chiffretext und Schlüssel sollte möglichst komplex sein.

- **Diffusion**: Jedes Chiffretextzeichen sollte von möglichst vielen Klartext- und Schlüsseltextzeichen abhängen.

Beide Prinzipien sollen garantieren, dass die statistischen Eigenschaften eines typischen Klartextes im Chiffretext verwischt sind.

Diffusion

Optimale Diffusion:

Strict avalanche criterion: ändert man man *ein* Bit im Klartext, so ändert sich *jedes* Bit im Chiffretext mit Wahrscheinlichkeit $1/2$.

Beispiel affine Chiffren: ist A hinreichen allgemein (z.B. eine beliebige Permutationsmatrix), so hat das Verfahren eine hohe Diffusion.

Konfusion

Kein klar definierter Begriff.

Praxisorientierte Auslegung: affine Chiffren haben geringe Konfusion, da sie leicht durch einen Known-Plaintext-Angriff gebrochen werden können.

In modernen Blockchiffren versucht man, Konfusion durch S-**Boxen** zu realisieren:

Ein Klartext wird in kleinere Blöcke $s_1, \ldots, s_m$ aufgeteilt. Auf jeden dieser Blöcke wendet man eine *nichtlineare* Funktion f_i an. Der Chiffretext ist dann

$$f_1(s_1), \ldots, f_m(s_m).$$

Produktchiffren

Eine *Produktchiffre* entsteht aus der Hintereinanderausführung mehrerer Blockchiffren:

$$c = \tilde{E}_k(p) = E_{k_1}(E_{k_2}(\ldots E_{k_r}(p)\ldots)).$$

Strategie:

- Durch Verknüpfung von einfachen Chiffren entstehen komplexe Chiffren
- Man realisiert abwechselnd Konfusion (z.B. durch S-Boxen) und Diffusion (z.B. durch Permutationen).
- Die Schlüsselfolge $k_1, \ldots, k_r$ ist eine Pseudozufallsfolge mit Startwert k (*Rundenschlüssel*.

Feistel-Chiffren

Ausgangspunkt: eine Blockchiffre mit Alphabet $\Sigma = \{0, 1\}$ und Blocklänge t. Sei

$$f_k : \Sigma^t \to \Sigma^t$$

die Verschlüsselungsfunktion zum Schlüssel $k \in \mathcal{K}$.

Resultat: Eine Blockchiffre mit Alphabet Σ, Blocklänge $2t$ und Verschlüsselungsfunktion

$$E_k : \mathcal{P} = \Sigma^{2t} \to \mathcal{C} = \Sigma^{2t}.$$

Feistel-Chiffren

- Schreibe $p = (L_0, R_0)$.
- Ausgehend vom Schlüssel k, konstruiere Folge

$$k_1, k_2, \ldots, k_r.$$

- Konstruiere Folge (L_i, R_i), $i = 1, \ldots, r$:

$$(L_i, R_i) := (R_{i-1}, L_{i-1} \oplus f_{k_i}(R_{i-1}))$$

  ($\oplus$: exklusives Oder).

- Setze

$$E_k(p) := (L_r, R_r).$$

DES

- Leicht modifizierte Feistel-Chiffre
- Klar- und Chiffretextblöcke: 64 Bit
- Schlüssel: 64 Bit mit einer Quersummenbedingung, daher $2^{56} \sim 7,2 \cdot 10^{16}$ Möglichkeiten
- 16 Verschlüsselungsrunden
- zusätzlich eine (vom Schlüssel unabhängige) *initiale Permutation*

DES

Die interne Blockchiffre

$$f_k : \{0,1\}^{32} \to \{0,1\}^{32}$$

zu $k \in \{0,1\}^{48}$:

$$f_k(R) := P(S(E(R) \oplus k)),$$

mit einer *Expansionsfunktion*

$$E : \{0,1\}^{32} \to \{0,1\}^{48},$$

einer S-Box-Funktion

$$S : \{0,1\}^{48} \to \{0,1\}^{32}$$

mit 8 Blöcken und einer Permutation

$$P : \{0,1\}^{32} \to \{0,1\}^{32}.$$