

One-time-pad (Vernam)

Wähle k_0, k_1, \dots, k_{n-1} zufällig
und verwende Schlüssel für
nur einen Klartext

Eigenschaften:

- Das Kryptogramm ist zufällige Folge von n Bits
- Ohne Kenntnis des Schlüssels kann das Kryptogramm nicht entschlüsselt werden, denn:
für jeden Klartext der Länge n gibt es einen Schlüssel der Länge n , der denselben Geheimtext erzeugt.

Perfekte Sicherheit

One-time-pad

k_i , $1 \leq i \leq n$ zufällig

$c_i = b_i \oplus k_i$, $1 \leq i \leq n$

- Schlüssel muß
 - vorher übermittelt worden sein
 - zufällig gewählt worden sein
 - sicher aufbewahrt werden
 - mindestens so lang wie die Nachricht sein
- sehr aufwendiges Verfahren,
das aber benutzt wurde

Produktchiffren

Hintereinander ausführen von
t Chiffren F_1, \dots, F_t

(Substitutions- oder
Transpositionschiffre)

DES (Data Encryption Standard)

64 - Bit - Blöcke von Daten

mit einem 56 - Bit - Schlüssel chiffriert
(in 16 Runden)

Suchraum: $2^{56} \sim 2.7 \cdot 10^{16}$ Schlüssel

DES gilt nicht mehr als sicher

Weiterentwicklung

a) Triple - DES

DES 3 mal hintereinander anwenden
mit 2 Schlüsseln (a 56 Bit)

$$c = E_{k_1}(D_{k_2}(E_{k_1}(m)))$$

Schlüsselraumgröße: 2^{112}

- Warum nicht nur 2 mal anwenden?

$$c = E_{k_2}(E_{k_1}(m))$$

→ ist "leichter" zu brechen
(Known-Plaintext-Angriff)

b) AES (Advanced Encryption Standard)

Blockgröße 128 Bit

Schlüssellängen 128, 192, u. 256 Bit
sind möglich und können eingestellt
werden

Sieger im Wettbewerb 2000

Rijndael (nach J. Daemen u. V. Rijmen,
Belgien)

Telefonbuch verschlüsselung

- Wähle für $\alpha \in \{A, B, \dots, Z\}$ zufällig Namen, die mit α beginnt, aus Telefonbuch aus.
Verschlüssele α mit entsprechender Telefonnummer

Klartext	gewählter Name	Telefonnr.
K	Kunze	42103
L	Logisch	11432
A	Amstel	94822
R	Reschke	75001

$$A = (a_1, a_2, \dots, a_n)$$

$$x = x_1 x_2 \dots x_n, \quad x_i \in \{0, 1\}$$

$$f_A(x) = \sum_{i=1}^n x_i a_i$$

Verschlüsselung
von x

$f(x)$ kann leicht mit Hilfe von
 $= f_A(x)$ A berechnet werden!

1. Ansatz

"Einwegfunktion" $f: \mathbb{N} \rightarrow \mathbb{N}$

a) $f(x)$ kann in polynomieller Zeit berechnet werden
($O(\log^k x)$, k konstant)

b) $f^{-1}(x)$ benötigt mindestens exponentielle Zeit, d.h. $\Omega(2^{\log x})$ z.B.
(außer man hat Zusatzinformation)

- bisher: keine Beweise, dass es solche Funktionen gibt; aber der starke Verdacht auf ihre Existenz ist da

- Man begnügt sich mit:

b') $f^{-1}(x)$ benötigt nach heutigem Kenntnisstand exponentielle Zeit

Achtung:

bezieht sich auf Worst-Case-Komplexität