Enigma: Drei Rotoren, jeder führt zu einer einstelligen monoalphabetischen Substitution. Erster Rotor wird bei jedem Buchstaben um eine Stellung weitergeschaltet, weiteren Rotoren folgen dann wie bei einem Zahlwerk

polyalphabetische Substitution

<u>allgemein</u>: Schlüssel besteht hier nicht mehr nur aus einer Permutation, sondern aus einer geordneten Folge von Permutationen

$$E_e(m) = (p_1(m_1)\, p_2(m_2)\, p_3(m_3) \ldots\ldots p_t(m_t))$$

<u>Beispiel</u>: Sei A Alphabet der Großbuchstaben, $t = 3$. dann sei:

$e = (p_1, p_2, p_3)$ mit
p_1: Rechtsverschiebung +3
p_2: " +7
p_3: " +10

m = THI SCI PHE RIS CER TAI
c = WOS VJS SOO UPC FLB WHS

Ferner möglich: <u>Transposition</u>: Permutation aller Symbole in einem Zeichenblock der Länge t. K ist damit die Menge aller Permutation auf der Menge $\{1, 2, \ldots, t\}$

Weder einfache Substitution noch die Transposition erbringen einen hohen Schutz. Ihre Kombination jedoch ist jedoch die Grundlage sehr sicherer Verfahren.

Beispiel (Produktchiffrierung)

Sei $M = C = K$ die Menge aller 6-bit-Wörter. Dann ist $\text{card}(M) = 2^6 = 64$ und ein Wort $m = (m_1\, m_2\, m_3\, m_4\, m_5\, m_6)$

1. Schritt: Polyalphabetische Substitution

$$E_k^{(1)}(m) = m \oplus k$$

wo '$\oplus$' Addition modulo 2 bzw. $\oplus \hat{=}$ XOR und $k \in K$

2. Schritt Transposition (mit oder ohne Schlüsselabhängigkeit)

$$E^{(2)}(m) = (m_4\, m_5\, m_6\, m_3\, m_2\, m_1)$$

Produkt $E_k^{(1)}(m)\, E^{(2)}(m)$ wird als 'Runde' bezeichnet.

Heise iX 12/2002 Artikel 'Angebrochen'

→ AES

Algorithmen und Textverarbeitungssysteme

- Ein Algorithmus heißt für eine Eingabe terminierend, wenn er stets für alle zulässigen Schrittfolgen und endlich vielen

Schritte endet

deterministisch, wenn in der Ausführung der Verarbeitungsschritte keine Freiheit besteht

determiniert, wenn das Resultat eindeutig bestimmt ist

sequentiell, wenn die Verarbeitungsschritte stets hintereinander ausgeführt werden

parallel wenn gewisse Verarbeitungsschritte neben einander ausgeführt werden.

Die Beschreibung eines Algorithmus in einer formalen (formal beschreibbaren) Sprache heißt ein Programm, die formale Sprache Programmiersprache

Ein solches Konzept zur ~~Transformation~~ Verarbeitung: Ersetzung (Substitution) von Teilworten durch andere Wörter ⇒ Ersetzungssysteme auf Zeichenketten.

Def.: Sei V ein Zeichenvorrat, dann ist ein Ersetzungsregel über V gegeben durch ein Paar $(v, w) \in V^* \times V^*$, man schreibt

$v \rightarrow w$

Ein solches Beispiel für Ersetzungsregel:

'ae' → 'ä'

Eine endliche Menge R von Ersetzungsregeln heißt ~~Textersetzungssystem~~ über V

Anwendung: Ersetzungsregeln sollen auf beliebigen Teiltermen (d.h. in beliebigen 'Kontexten' angewendet werden. Also:
Eine Ersetzung $s \to t$ heißt Anwendung der Regel $v \to w$, falls es Wörter $a, v, w, z \in V^*$ gibt, so daß gilt:

$$s = a \circ v \circ z, \quad t = a \circ w \circ z$$

Ein Wort $s \in V^*$ heißt terminal, falls es kein Wort $t \in V^*$ gibt, so daß gilt: Die Ersetzung $s \to t$ ist Anwendung einer Ersetzungsregel. M.a.W.: Auf das terminale Wort kann keine Regel mehr angewendet werden.

Durch iterierte Anwendung von Regeln wird ausgehend von einem Wort t_0 eine Berechnung erzeugt. Berechnungen werden geschrieben als:

$$t_0 \to t_1 \to t_2 \to \cdots \to t_n$$

t_0: Eingabe, t_n: Ausgabe

Beispiel für Berechnung von Textersetzungssystem:

a) Ersetzen Q über Zeichenmenge $\{L, 0\}$, das die beiden Regeln umfaßt:

$$LL \to E, \quad 0 \to E \quad \text{mit } E = \text{'Leerstring'}$$

ist eine Berechnung für die Eingabe

$LOLL \to LO \to L$

b) System Q' über $\{L, O\}$ mit Regeln:

$O \to OO$, $O \to L$

ist die Berechnung basierend auf Eingabe '0': $O \to OO \to OL \to LL$

terminierend, während z.B. die Berechnung

$O \to OO \to OOO \to \ldots$

nicht terminierend ist.

Ein Ersetzungssystem definiert ein folgt einen Algorithmus, der Wörter über V als Ein- und Ausgabe verwendet: Das Eingabewort $t \in V^*$ wird ein folgt verarbeitet.

"Ist eine beliebige der Regeln aus R auf das Wort t anwendbar, es existiert also ein Wort $s \in V^*$, so daß gilt: $t \to s$ ist eine Anwendung einer Regel aus R), so wendet man diese Regel auf t an und setzt dann den Algorithmus mit dem Wort s fort, andernfalls: Abbruch"

Beispiele für Textersetzungsalgorithmen:

a) Addition von zwei Zahlen in Strichdarstellung:

Zahl $n \in \mathbb{N}$ wird durch das Wort $<|||\ldots|>$ dargestellt. Der Algorithmus besteht dann aus einer Regel:

$$>+< \rightarrow \varepsilon$$

Für Eingabe $<||\ldots|>+<|\ldots|>$ liefert Algorithmus Summe der Striche als Ausgabe.

b) Multiplikation zweier nat. Zahlen in Strichdarstellung. Ersetzungsregeln:

$|>*< \rightarrow >*<d$

$d| \rightarrow |md$

$dm \rightarrow md$

$d> \rightarrow >$

$<>*< \rightarrow <e$

$e| \rightarrow e$

$em \rightarrow |e$

$e> \rightarrow >$