

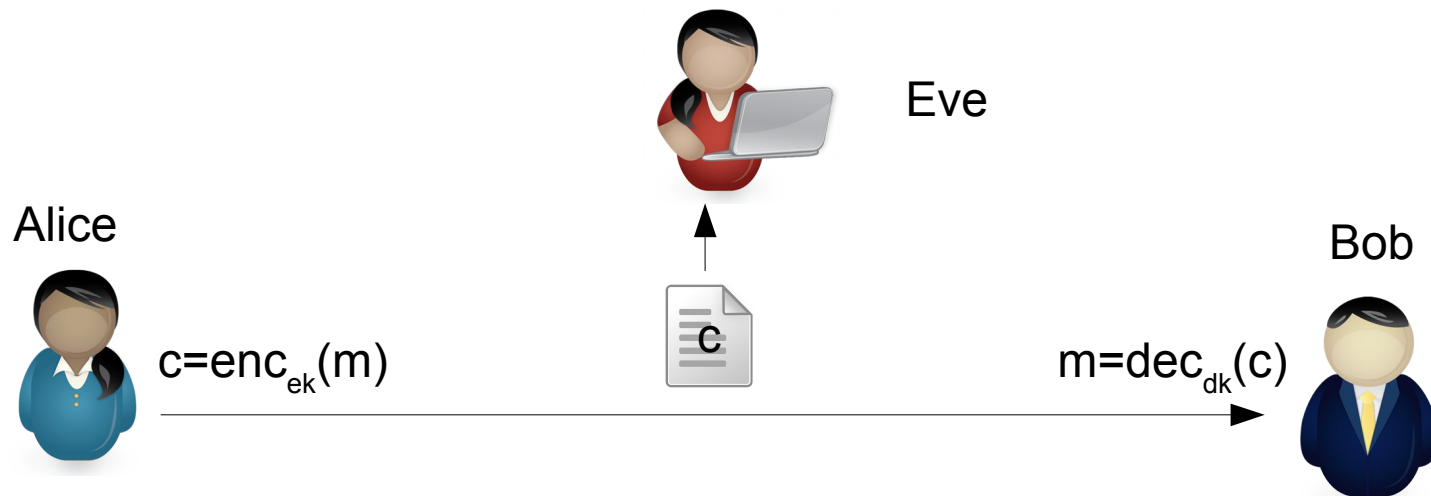
Themen zur Computersicherheit

Verschlüsselung

PD Dr. Reinhard Bündgen
bueundgen@de.ibm.com

Kryptographisches System

- Alphabete Σ_1, Σ_2
- Verschlüsselungsverfahren / Chiffre / englisch: cipher
 - $\text{enc}: eK \times \Sigma_1^* \rightarrow \Sigma_2^*$, $\text{dec}: dK \times \Sigma_2^* \rightarrow \Sigma_1^*$
 - Schlüssel $(ek, dk) \in eK \times dK$ mit $\text{dec}_{dk}(\text{enc}_{ek}(m)) = m$ für alle $m \in \Sigma_1^*$
 - verschlüsselt Klartext (plain text) zu Geheimtext (cipher text)



Angriffe auf ein kryptographisches System

- Ziele eines Angriffs
 - finde Klartext zu gegebenem Geheimtext
 - finde Schlüssel
- oft erreicht ein Angriff nur ein Teilziel
 - Eigenschaften des Geheimtextes
 - Eigenschaften des Schlüssels
- das allerdings von einem weiteren Angriff genutzt werden kann
- I.A braucht man zusätzliche Informationen über den Klartext oder Geheimtext um entscheiden zu können, ob ein Angriff erfolgreich war
- Seitenkanalangriffe
 - nutzen ein Informationsleck des Verschlüsselungsverfahrens
 - z.B Dauer der Berechnung, Strahlung, ...

Angriffsformen

- Art des Zusatzinformationsgewinns
 - Angriff mit bekanntem Geheimtext (cipher text only attack, COA)
 - Angreifer kennt einen oder mehrere Geheimtexte
 - Angriff mit bekanntem Klartext (known plaintext attack, KPA)
 - Angreifer kennt die zu einem oder mehreren Geheimtexten gehörigen Klartexte
 - Angriff mit gewähltem Klartext (chosen plain text attack, CPA)
 - Angreifer kann sich zu gewählten Klartexten die dazugehörigen Geheimtexte berechnen lassen
 - Angriff mit gewählten Geheimtext (chosen cipher text attack, CCA)
 - Angreifer kann sich zu gewählten Geheimtexten die dazugehörigen Klartexte berechnen lassen
- Zeitpunkt des Zusatzinformationsgewinns
 - direkter Angriff (offline, batch)
 - vor bekannt werden der Herausforderung
 - adaptiv (online, adaptive)
 - nach bekannt werden der Herausforderung

Orakel

- Angriffs-Orakel: Funktion die Verschlüsselungsaufgaben als „black box“ löst
 - COA-Orakel erzeugt zufälligen Geheimtext
 - KCA-Orakel erzeugt zufälligen Klartext mit zugehörigem Geheimtext
 - CPA-Orakel berechnet zu gegebenem Klartext den Geheimtext
 - CCA-Orakel berechnet zu gegebenem Geheimtext den dazugehörigen Klartext
- IND-Orakel verschlüsselt zufällig einen von zwei gegebenen Klartexten und gibt den Geheimtext aus
- Orakel berechnet *nicht* die Lösung der Herausforderung

Ununterscheidbarkeit (indistinguishability)

Ununterscheidbarkeitstest:

- 1) beim direkten Angriff: Angreifer befragt das Angriffs-Orakel
- 2) Angreifer wählt 2 Nachrichten m_1, m_2
- 3) IND-Orakel wählt ein $m \in \{ m_1, m_2 \}$
- 4) IND-Orakel gibt $c = \text{enc}_k(m)$ aus
- 5) bei adaptivem Angriff: Angreifer befragt das Angriffs-Orakel
- 6) Angreifer gibt als Resultat seiner Angriffs: $b \in \{ 1, 2 \}$ aus

Der Test ist (im Sinne des Angreifers) bestanden, wenn $c = \text{enc}_k(m_b)$

Definition: Ein kryptografisches System ist gegen einen Angriff *sicher im Sinne der polynomiellen Ununterscheidbarkeit*, wenn der Angriff den Ununterscheidbarkeitstest nur mit einer Wahrscheinlichkeit von 0,5 besteht.

Definition: Ein kryptografisches System ist *perfekt sicher*, wenn es gegen beliebige Angriffe sicher im Sinne der polynomiellen Ununterscheidbarkeit ist.

Alternative Definition: Ein kryptografisches System ist *perfekt sicher*, wenn nach jedem Angriff, die *a priori Wahrscheinlichkeit*, dass ein beliebiger Klartexte das Ergebnis der Entschlüsselung eines Geheimtextes ist, gleich der *a posteriori Wahrscheinlichkeit*, dass der Klartexte das Ergebnis der Entschlüsselung eines Geheimtextes ist.

One Time Pad

- **Definition** Das folgende Kryptosystem heißt *one time pad*:
 - Sei $\Sigma = \Sigma_1 = \Sigma_2 = \{0,1\}$, $eK = dK = \Sigma^*$,
 - $m \in \Sigma^*$
 - $k \in \Sigma^*$ ist ein Zufallsbitstring mit $|k| = |m|$
 - und $enc_k(m) = dec_k(m) = k \oplus m$ (\oplus bit-weises xor)
- **Satz** Das one time pad ist perfekt sicher, wenn jeder Schlüssel genau einmal zur Verschlüsselung einer Nachricht verwendet wird.
- **Achtung:** Ein Zufallsbitstring darf nur einmal als one time pad Schlüssel verwendet werden!
 - seien $m_1, m_2, k \in \Sigma^*$, $|m_1| = |m_2| = |k|$, $c_1 = m_1 \oplus k$, $c_2 = m_2 \oplus k$
 - dann ist $c_1 \oplus c_2 = m_1 \oplus m_2$
 - wenn m_2 bekannt: $m_1 = c_1 \oplus c_2 \oplus m_2$

Historische Chiffren I

- Caesar Chiffre

a	b	c	d	...	v	w	x	y	z
↓	↓	↓	↓		↓	↓	↓	↓	↓
d	e	f	g	...	y	z	a	b	c

- Verschiebechiffren („Caesar Chiffren“)

- $\Sigma = \Sigma_1 = \Sigma_2 = \{s_0, s_1, \dots, s_{n-1}\}$, $ek = dk \in \{0, \dots, n-1\}$, $|enc_k(m)| = |m|$

- $enc_k: s_i || m \mapsto s_{(i+k) \bmod n} || enc_k(m)$ für $s_i \in \Sigma$, $m \in \Sigma^*$

- $dec_k: s_i || m \mapsto s_{(i-k) \bmod n} || dec_k(m)$ für $s_i \in \Sigma$, $m \in \Sigma^*$

- Schlüsselraum: n verschiedene Schlüssel

- Angriff: vollständige Suche

- Monoalphabetische Chiffren

- $\Sigma = \Sigma_1 = \Sigma_2 = \{s_0, s_1, \dots, s_{n-1}\}$

- $ek=dk \in \{\pi \in \Sigma \rightarrow \Sigma \mid \pi \text{ ist Permutation}\}$, $|enc_\pi(m)| = |m|$

- $enc_\pi: s || m \mapsto \pi(s) || enc_\pi(m)$ für $s \in \Sigma$, $m \in \Sigma^*$

- $dec_\pi: s || m \mapsto \pi^{-1}(s) || dec_\pi(m)$ für $s \in \Sigma$, $m \in \Sigma^*$

- Schlüsselraum: $|\Sigma|!$ verschiedene Schlüssel

- Angriff: statistische Analyse

Historische Chiffren II

Polyalphabetische Chiffren

- Vigenère Chiffre (um 1500)
 - Sei $|\Sigma| = n$, $\text{ord}: \Sigma \rightarrow \{0,1,\dots,n-1\}$ bijektiv
 - $eK = dK = \Sigma^*$ mit $e_k = d_k = k \in \Sigma^*$ für alle Schlüssel
 - seien $a, s \in \Sigma$ und $k, m \in \Sigma^*$ dann ist
 - $\text{enc}_{a||k}(s||m) = \text{enc}_{\text{ord}(a)}(s) || \text{enc}_{k||a}(m) = \text{ord}^{-1}((\text{ord}(s) + \text{ord}(a)) \bmod n) || \text{enc}_{k||a}(m)$
- Angriffe
 - Vigenère Chiffre galt lange Zeit als unentzifferbar
 - Bestimmung der Schlüssellänge (Kasiski)
 - gleiche Geheimtextstücke (die zu gleichen Klartextstücken gehören) haben einen Abstand, der ein vielfaches der Schlüssellänge beträgt
 - Bestimmung von sprachlicher Redundanz im Geheimtext (Friedman)
 - Friedmanscher Koinzidenzindex: Wahrscheinlichkeit dass zwei beliebig aus einem Text gewählte Buchstaben gleich sind.
 - Für jede Sprache gibt es charakteristische Wahrscheinlichkeit, dass zwei aus einem Text gewählte Buchstaben gleich sind (folgt aus statistischer Buchstabenhäufigkeit)
 - Sei $|k|$ die gesuchte Schlüssellänge, f der sprachtypische Koinzidenzindex, g der Koinzidenzindex eines Zufallstextes (Gleichverteilung der Buchstaben), $|m|$ die Länge des Textes, p die Anzahl der gezogenen Paare aus gleichen Buchstaben
 - theoretisch: $p = f \cdot |m| \cdot (|m|/|k| - 1) / 2 + g \cdot |m| \cdot (|m| - |m|/|k|) / 2$
 - $F = p \cdot 2 / (|m| \cdot (|m|-1))$ ist Approximation des Friedmanschen Koinzidenzindexes
 - $\Rightarrow |k| = (f - g) \cdot |m| / ((|m|-1) \cdot F - g \cdot |m| + f)$

	A	B	C	D	...	X	Y	Z
A	A	B	C	D	...	X	Y	Z
B	B	C	D	E	...	Y	Z	A
C	C	D	E	F	...	Z	A	B
D	D	E	F	G	...	A	B	C
...								
X	X	Y	Z	A	...	U	V	W
Y	Y	Z	A	B	...	V	W	X
Z	Z	A	B	C	...	W	X	Y

Rechnung

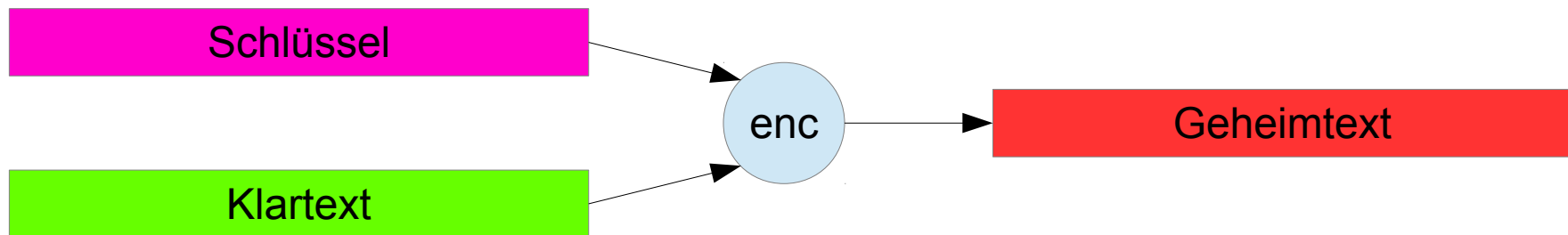
- gesuchte Schlüssellänge: $|k|$
- Textlänge $|m|$
- sprachtypischer Koinzidenzindex: f
- Koinzidenzindex eines Zufalltextes: g
- Anzahl der Paare gleicher Buchstaben in Geheimtext: P
- Buchstaben pro Spalte: $|m| / |k|$
- Buchstabenpaare aus gleicher Spalte: $|m| \cdot (|m|/|k| - 1) / 2$
- Buchstabenpaare aus verschiedenen Spalten: $|m| \cdot (|m| - |m|/|k|) / 2$
- Erwartete Anzahl von Paaren aus gleichen Buchstaben:
 - $p = f \cdot |m| \cdot (|m|/|k| - 1) / 2 + g \cdot |m| \cdot (|m| - |m|/|k|) / 2$
- Approximation des Friedmanschen Koinzidenzindexes:
 - $F = P \cdot 2 / (|m| \cdot (|m| - 1))$ (# gleicher Paare / # der Paare)
- P durch p ersetzen und umformen:
 - $\Rightarrow |k| = (f - g) \cdot |m| / ((|m| - 1) \cdot F - g \cdot |m| + f)$

Historische Chiffren III

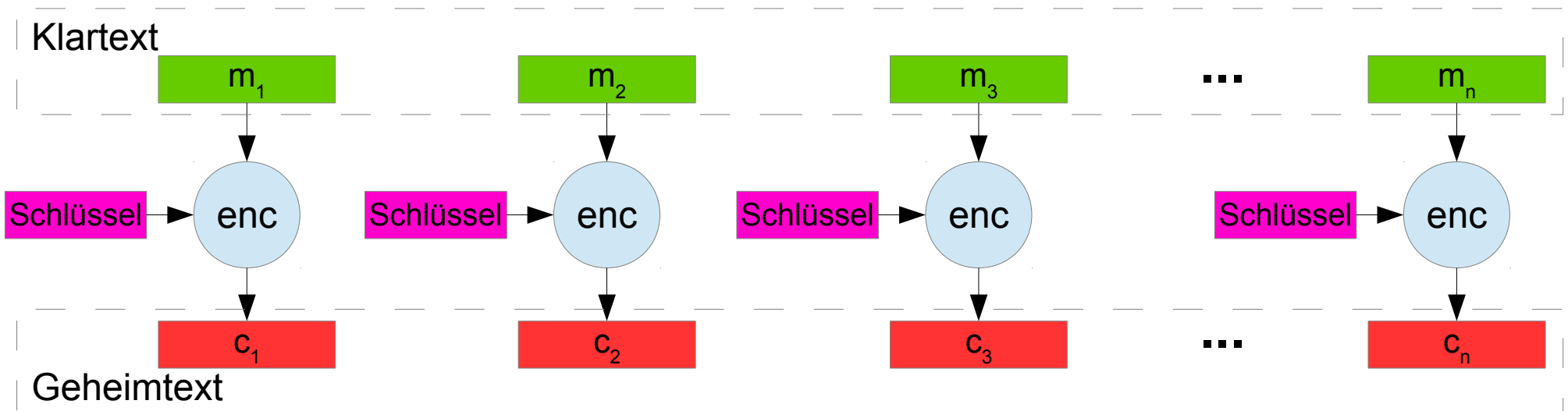
- die Enigma → getrennte Präsentation

Moderne Chiffren

- Stromchiffren
 - Verknüpfung eines Klartextes mit Schlüsselstrom
 - Verknüpfung: oft xor
 - Schlüssel: generiert mit Pseudozufallsgenerator (mit vereinbartem initialen Zustand)
 - Klartext kann beliebige Länge haben



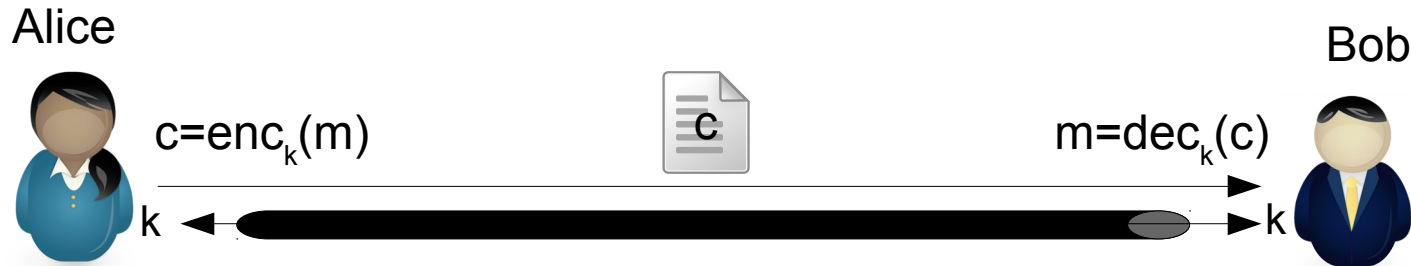
- Blockchiffren
 - Verschlüsselung von Nachrichtenblöcken mit Hilfe eines konstanten Schlüssels
 - monoalphabetische Verschlüsselung auf großem Alphabet
 - Klartextlänge muss Vielfaches der Blocklänge sein



Symmetrische und asymmetrische Chiffren

- **Symmetrische Chiffren**

- $eK=dK (= K)$ und für alle Schlüssel $(ek,dk) \in eK \times dK$ gilt $ek = dk (=k)$



- **Asymmetrische Chiffren**

- für alle Schlüssel $(ek,dk) \in eK \times dK$ gilt
 - $ek \neq dk$ und
 - dk kann nicht effizient aus ek berechnet werden
- ek heißt auch öffentlicher Schlüssel (public key pk)
- dk heißt auch privater Schlüssel (private/secret key sk)

