

---

# Kryptographie praktisch erlebt

Dr. G. Weck  
INFODAS GmbH  
Köln



---

## Inhalt

- Klassische Kryptographie
- Symmetrische Verschlüsselung
- Asymmetrische Verschlüsselung
- Digitale Signaturen
- Erzeugung gemeinsamer Schlüssel



# Die Cäsar-Chiffre

- Jedes Zeichen wird ersetzt durch ein Zeichen, das im Alphabet um eine bestimmte Anzahl von Stellen später / voran steht:

abcdefghijklmnopqrstuvwxyz

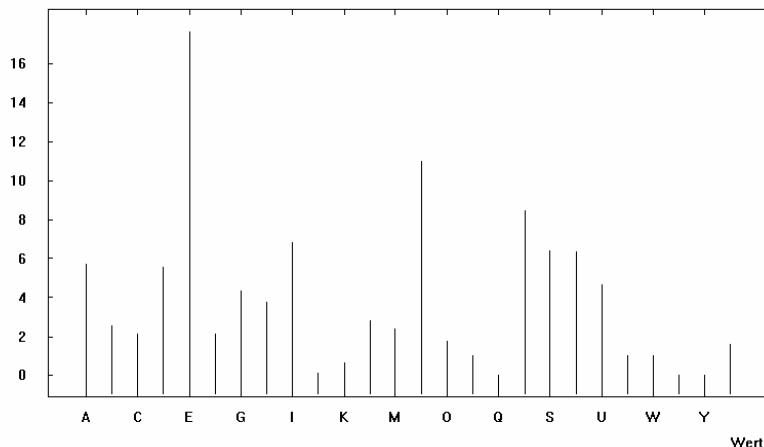


DEFGHIJKLMNOPQRSTUVWXYZABC

- Beispiel einer „monoalphabetischen Substitution“
- Durch Häufigkeitsanalyse der Buchstaben leicht zu rechen

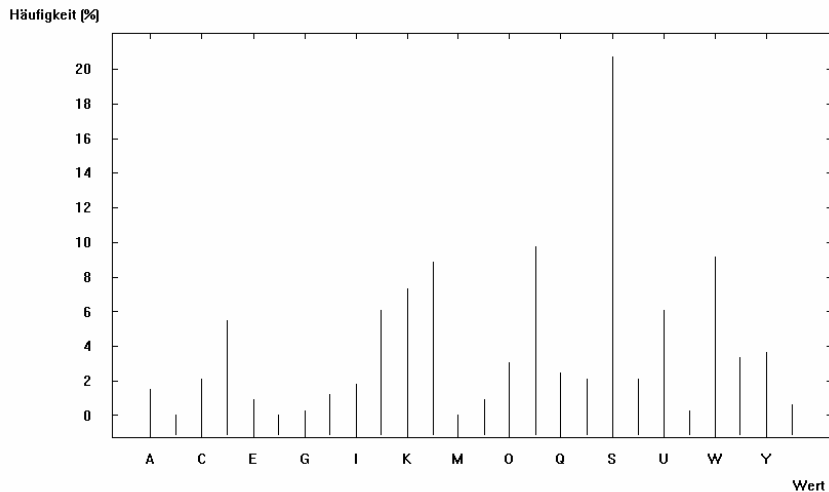
# Häufigkeit der Zeichen in deutschem Text

Häufigkeit (%)





# Häufigkeit der Zeichen im Text



05.11.2003

Kryptographie praktisch erlebt

Folie6



# Vigenère-Verschlüsselung

- Basiert auf Matrix verschobener Alphabete („polyalphabetische Substitution“)
- Buchstaben des Schlüsselwortes wählen (zyklisch) die zur Verschlüsselung verwendeten Zeilen aus
- Schlüssellänge kann durch Autokorrelation der Zeichenhäufigkeiten ermittelt werden

05.11.2003

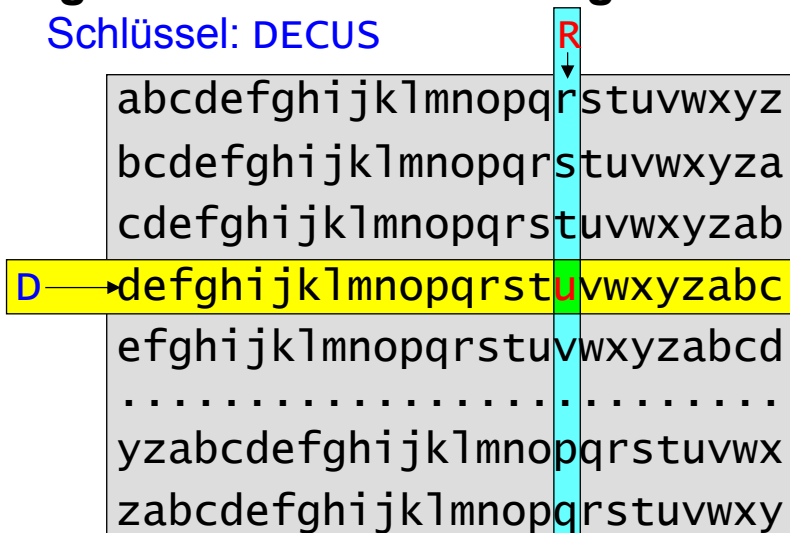
Kryptographie praktisch erlebt

Folie7



# Vigenère-Verschlüsselung

Schlüssel: DECUS



05.11.2003

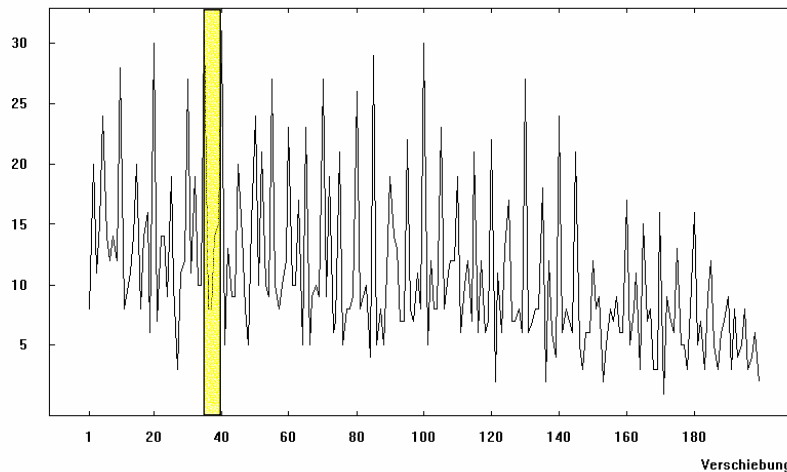
Kryptographie praktisch erlebt

Folie 8



# Autokorrelation der Häufigkeiten

Anzahl der übereinstimmenden Zeichen



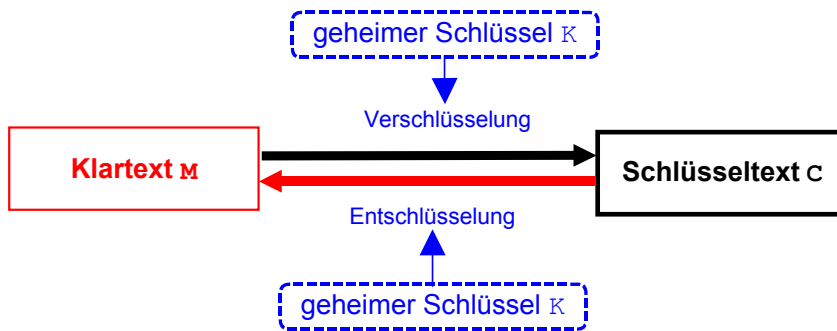
05.11.2003

Kryptographie praktisch erlebt

Folie 9



# Symmetrische Verschlüsselung



- Beziehungen:

$$C = \kappa(M) \text{ und } M = \kappa(C)$$

# Verschlüsselungs-Standard DES

- Von IBM entwickelt und (nach Modifikation) als US-Standard (FIPS PUB 46) festgelegt
- Zahlreiche Hardware-Verschlüsselungs-Geräte verfügbar
- Verschlüsselungsraten bis  $> 100$  MBit/Sekunde
- In den USA für nicht-militärische Datenkommunikation zwischen Regierungsstellen vorgeschrieben

# Verschlüsselungs-Standard DES

- Verschlüsselt jeweils 64 Bit lange Datenblöcke
- Schlüssellänge 56 Bit (ursprünglich 128 Bit)
- Separate Verfahren für Ver- und Entschlüsselung
- Nur der Schlüssel braucht geheim zu sein – der Algorithmus ist veröffentlicht
  - Vorteil: öffentlicher Kritik unterworfen
  - Nachteil: Verschlüsselung ist bei Bekanntwerden des Schlüssels gebrochen
- Erhöhte Sicherheit durch mehrfache Anwendung (Triple-DES, 3DES)

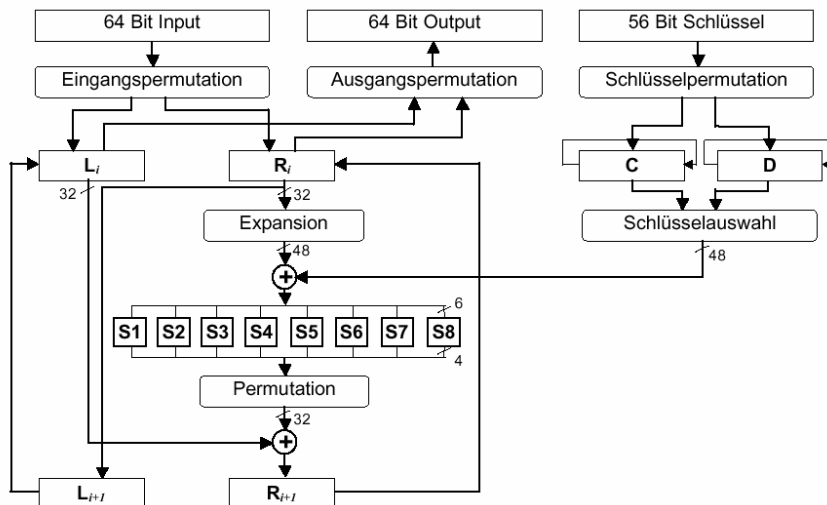
05.11.2003

Kryptographie praktisch erlebt

Folie12



## Der DES-Algorithmus



05.11.2003

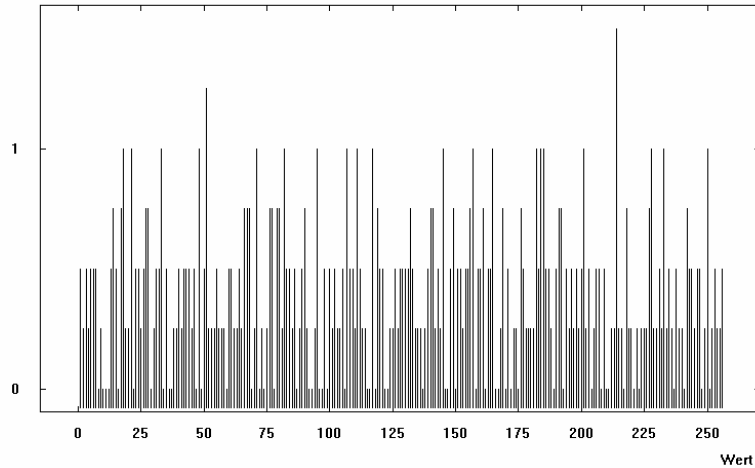
Kryptographie praktisch erlebt

Folie13



# Häufigkeit der Zeichen bei DES/ECB

Häufigkeit (%)



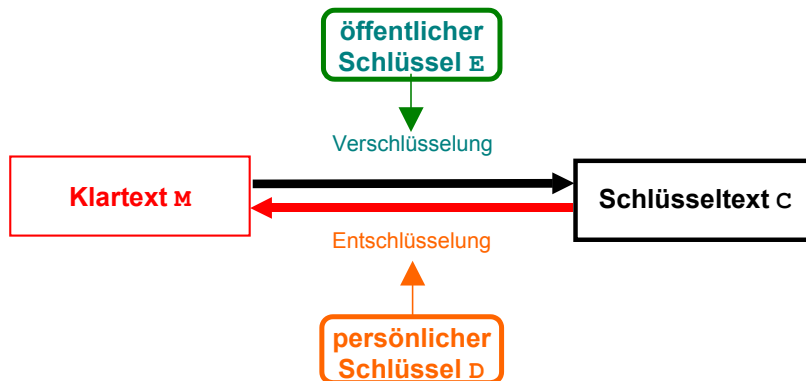
05.11.2003

Kryptographie praktisch erlebt

Folie14



# Asymmetrische Verschlüsselung



- Beziehungen:

$$C = E(M) \text{ und } M = D(C)$$

05.11.2003

Kryptographie praktisch erlebt

Folie15



---

## Öffentliche Schlüssel

- Getrennte Schlüssel für Ver- und Entschlüsselung
- Verschlüsselungs-Schlüssel **E** („öffentlicher Schlüssel, public key“) kann veröffentlicht sein
- Entschlüsselungs-Schlüssel **D** („persönlicher Schlüssel, private key“)
  - muß geheim bleiben
  - kann nicht aus **E** (oder den Nachrichten) abgeleitet werden

---

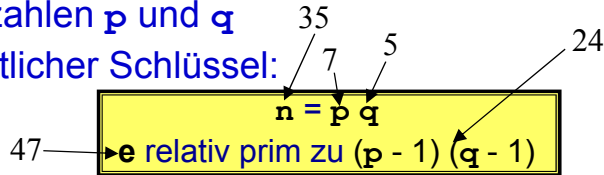
## Öffentliche Schlüssel

- Voraussetzung:  $D(E(x)) = x$
- Digitale Signaturen möglich, falls auch  $E(D(x)) = x$
- Verfahren:
  - RSA-Algorithmus (Rivest/Shamir/Adleman): beruht auf Primzahl-Zerlegung
  - ElGamal-Algorithmus: beruht auf diskreten Logarithmen
  - Diffie-Hellman-Algorithmus: Erzeugung eines gemeinsamen Schlüssels für zwei Kommunikationspartner, der nur diesen bekannt ist
- Hauptnachteil: sehr hoher Rechenaufwand

# RSA-Algorithmus

- Voraussetzung: zwei große (geheimgehaltene) Primzahlen  $p$  und  $q$

- Öffentlicher Schlüssel:



- Privater (geheimer) Schlüssel:

$$d = e^{-1} \text{ mod } ((p-1)(q-1))$$

(with  $d=23$ )

- Verschlüsselung:

$$c = m^e \text{ mod } n$$

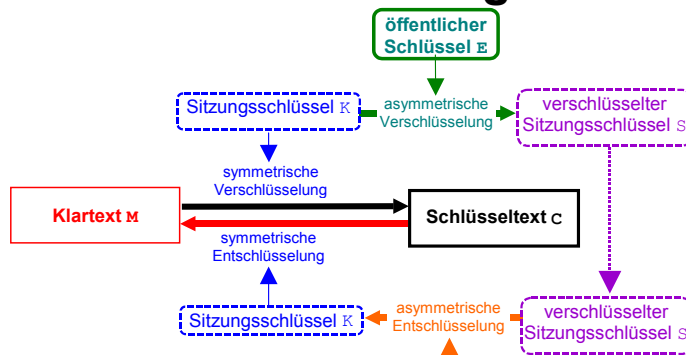
(with  $m=18, e=2, n=175$ , resulting in  $2^{47} \text{ mod } 35$ )

- Entschlüsselung:

$$m = c^d \text{ mod } n$$

(with  $c=18, d=23, n=175$ , resulting in  $18^{23} \text{ mod } 35$ )

# Hybride Verschlüsselungsverfahren

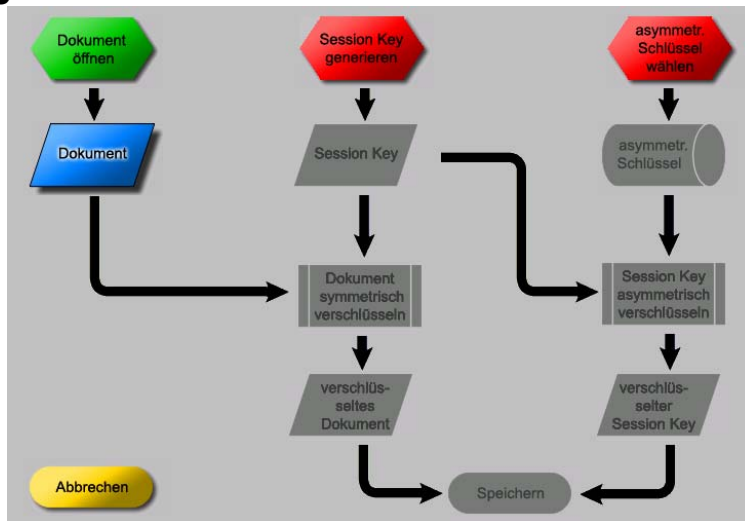


- Beziehungen:

$$S = E(K) \text{ und } C = \kappa(M)$$

$$K = D(s) \text{ und } M = \kappa(C)$$

# Hybrid-Demonstration



05.11.2003

Kryptographie praktisch erlebt

Folie 20



# Digitale Signaturen

- Schutz gegen Verlust von Integrität und Authentizität
- Beweiswert digitaler Signaturen
  - einsetzbar im Wege des Augenscheins-/Sachverständigenbeweises
  - beweishebliche Tatbestände:
    - Nachweis der Authentizität und Integrität - Nachricht ist unverändert
    - Nachweis der Urheberschaft der Nachricht / der Identität des Senders
    - Nachweis, daß die beabsichtigte Nachricht signiert wurde (offenes Problem!)

05.11.2003

Kryptographie praktisch erlebt

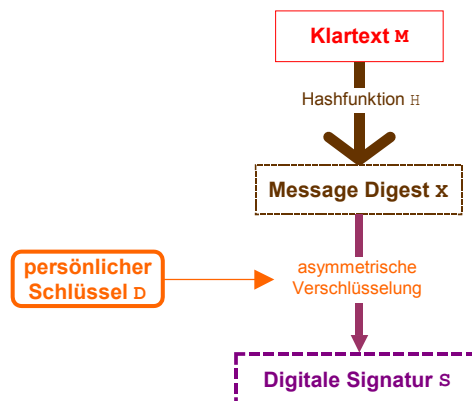
Folie 21



# Digitale Signaturen

- Anforderungen an digitale Signaturen
  - Abhängigkeit von der signierten Nachricht und vom Unterzeichner
  - Erzeugbarkeit nur durch den Unterzeichner
  - Überprüfbarkeit auf Echtheit durch jeden (autorisierten) Dritten
- Anforderungen an die verwendeten Mechanismen
  - hinreichende Stärke der eingesetzten Algorithmen
  - Authentizität der öffentlichen Schlüssel

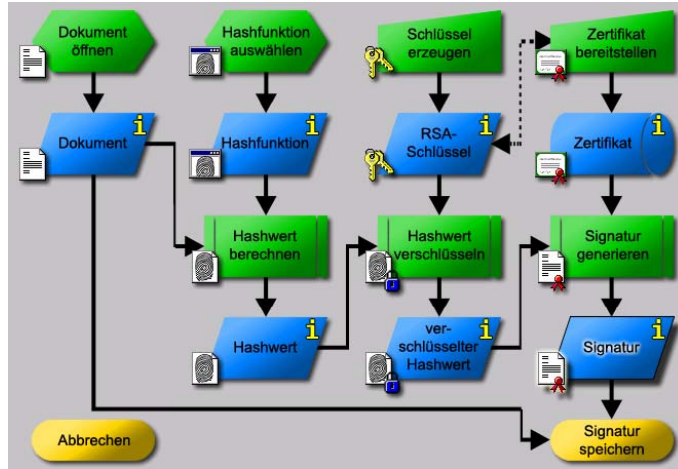
# Erzeugung digitaler Signaturen



- Beziehungen:

$$x = H(M) \text{ und } s = D(x)$$

# Erzeugung digitaler Signaturen



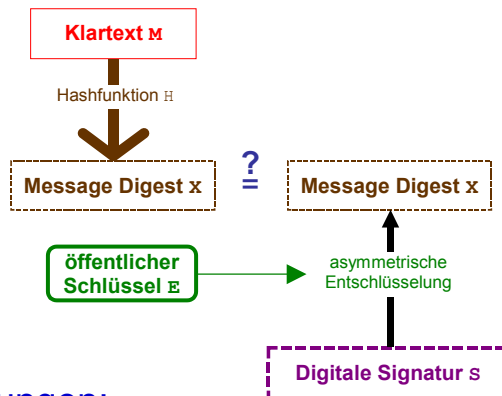
05.11.2003

Kryptographie praktisch erlebt

Folie24

infodas®

# Überprüfung digitaler Signaturen



• Beziehungen:

$$\begin{aligned}
 X &= H(M) \text{ und } X' = E(S) \\
 X &= X' ?
 \end{aligned}$$



05.11.2003

Kryptographie praktisch erlebt

Folie25

infodas®

## Erzeugung gemeinsamer Schlüssel

- Diffie/Hellman-Algorithmus erlaubt die Erzeugung eines Kommunikationsschlüssels ohne die Notwendigkeit zum Austausch geheimer Informationen
  - angreifbar durch Man-in-the-Middle Attacke

## Diffie-Hellman-Algorithmus

- Jeder Kommunikationspartner  $x$  erzeugt ein Schlüsselpaar

- seinen privaten Schlüssel  $D_x$ , den kein anderer Kommunikationspartner kennt

- den öffentlichen Schlüssel  $E_x$ , der allen Kommunikationspartnern bekannt sein darf

- Für jeden Kommunikationspartner  $x$  gilt

$$2^3 = 8 \longrightarrow \boxed{E_x = g^{Dx}} \longleftarrow 2^9 = 512 \equiv 6 \pmod{11}$$

- mit einer globalen Konstanten  $g$  modulo einer Primzahl  $p$

# Diffie-Hellman-Algorithmus

- Jeder Kommunikationspartner **x** sendet seinen öffentlichen Schlüssel  $E_x$  an den anderen Kommunikationspartner

$$6^3 = 216 \equiv 7 \pmod{11}$$

- Für die Kommunikation mit **B** berechnet **A**:

$$K_{AB} = E_B^{DA} = (g^{DB})^{DA} = g^{(DB \cdot DA)}$$

- Bei Empfang der ersten Nachricht von **A** berechnet **B**:

$$8^9 = 134217728 \equiv 7 \pmod{11}$$

$$K_{BA} = E_A^{DB} = (g^{DA})^{DB} = g^{(DA \cdot DB)} = K_{AB}$$

# Diffie-Hellman-Demonstration

Von beiden benutzte öffentliche Parameter

Primzahlmodul p:

Generator g:

**Alice**

Geheimnis:

a:

Berechnen

A:

Berechnen

s:

**Bob**

Geheimnis:

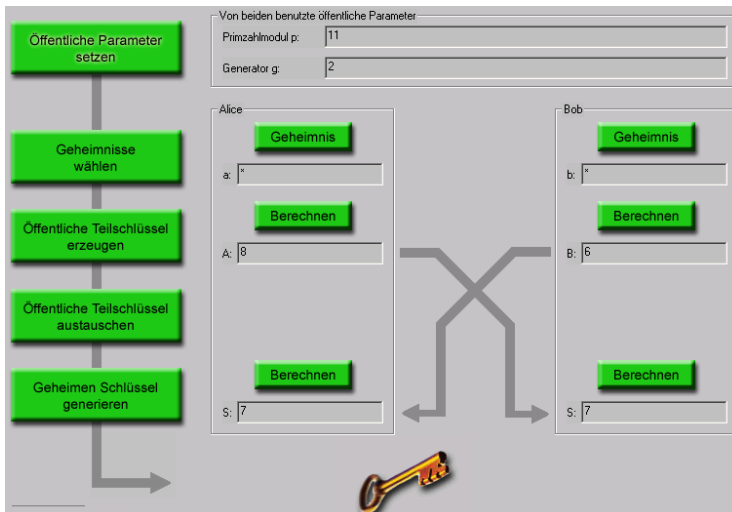
b:

Berechnen

B:

Berechnen

s:



---

## Quellen

- Die Software des Krypto-Toolkits kann von dieser Adresse geladen werden:  
<http://www.cryptool.org>
- **B. Esslinger, H. Koy:** *CrypTool – Kryptographie spielerisch verstehen*;  
in: Bundesamt für Sicherheit in der Informationstechnik (BSI): IT-Sicherheit im verteilten Chaos, Tagungsband 8. Deutscher IT-Sicherheitskongreß des BSI, SecuMedia Verlags-GmbH, Ingelheim, 2003