

# Starthilfe Kryptologie mit CrypTool

Version 1.4.10

**Einführung in Kryptographie und Kryptoanalyse**  
**Umfang, Technik und Zukunft von CrypTool**



[www.cryptool.de](http://www.cryptool.de)  
[www.cryptool.com](http://www.cryptool.com)  
[www.securecenter.ch](http://www.securecenter.ch)

# Inhalt / Übersicht

## I. CrypTool und Kryptographie – Überblick

1. [Das CrypTool-Projekt](#)
2. [Bedeutung der Kryptographie und Beispiele klassischer Verfahren](#)
3. [Erkenntnisse aus der Entwicklung der Kryptographie](#)

## II. Was bietet CrypTool ?

1. [Überblick](#)
2. [Beispiele zur Interaktion](#)
3. [Herausforderungen für Entwickler](#)

## III. Ausgewählte Beispiele

1. [RSA-Verschlüsselung](#)
2. [Elektronische Signatur visualisiert](#)
3. [Angriff auf RSA-Verschlüsselung](#)
4. [Psion-Analyse](#)
5. [Schwache DES-Schlüssel](#)
6. [Auffinden von NSA-Keys](#)
7. [Authentifizierungsarten](#)
8. [Authentifizierungsarten](#)
9. [Seitenkanalangriffs-Demo](#)
10. [Angriffe auf RSA per Gitterreduktion](#)
11. [Zufallsanalyse mit 3-D Visualisierung](#)
12. [Secret Sharing mittels CRT](#)
13. [Anwendung des CRT in der Astronomie](#)
14. [Visualisierung von Verfahren mit ANIMAL](#)
15. [Erzeugung eines MAC](#)
16. [Hash-Demo](#)

## IV. Projekt / Ausblick / Kontakt

## Das CrypTool-Projekt

- Ursprung im Awareness-Programm einer Großbank (betriebliche Ausbildung)  
→ **Sensibilisierung der Mitarbeiter**
- Entwickelt in Kooperation mit Hochschulen (Verbesserung der Lehre)  
→ **Mediendidaktischer Anspruch**

1998 – **Projektstart** – Aufwand bisher mehr als 15 Mannjahre

2000 – CrypTool als **Freeware** verfügbar

2002 – CrypTool auf der **Bürger-CD des BSI** „Ins Internet – mit Sicherheit“

2003 – CrypTool wird **Open-Source** – Hosting durch die Uni Darmstadt (Fr. Prof. Eckert)

2004 – Auszeichnungen:

TeleTrust (TTT Förderpreis 2004)



NRW (IT-Sicherheitspreis NRW)



RSA Europe (Finalist beim European Information Security Award 2004)



### ▪ **Entwickler**

- Entwickelt von Mitarbeitern verschiedener Firmen und Universitäten
- Weitere Projekt-Mitarbeiter oder verwertbare vorhandene Sourcen sind immer herzlich willkommen (bisher arbeiten ca. 30 Leute weltweit mit).

# Bedeutung der Kryptographie

## Typischer Einsatz von Kryptographie im Alltag

### Einsatzbeispiele für Kryptographie

- Telefonkarten, Handys, Fernbedienungen
- Geldautomaten, Geldverkehr zwischen Banken
- Electronic cash, Online-Banking, Sichere E-Mail
- Satellitenfernsehen, PayTV
- Wegfahrsperrung im Auto
- Digital Rights Management (DRM)



- Kryptographie ist schon lange nicht mehr nur auf Agenten, Diplomaten und Militärs begrenzt. Kryptographie ist eine moderne, mathematisch geprägte Wissenschaft.
- Der Durchbruch für den breiten Einsatz kam mit dem Internet.
- Für Firmen und Staaten ist es wichtig, dass sowohl die Anwendungen sicher sind, als auch, dass ...

*... die Nutzer (Kunden, Mitarbeiter) ein Mindestverständnis und Bewusstsein (Awareness) für IT-Sicherheit besitzen !*

# Kryptographie – Was will man damit erreichen ?

## Sicherheitsziele im Kontext von Kryptographie

- **Vertraulichkeit** (*Confidentiality*)
  - Lesen des eigentlichen Inhalts für Unbefugte „praktisch“ unmöglich machen
- **Authentifizierung** (*Authentication*)
  - Identitätsbeweis des Senders einer Nachricht gegenüber dem Empfänger
- **Integrität** (*Integrity*)
  - Eigenschaft, dass die Nachricht nicht verändert wurde
- **Verbindlichkeit, Verfügbarkeit** (*Non-Repudiation*)
  - Der Empfänger kann den Nachweis erbringen, dass der Sender die Nachricht abgeschickt hat (Leugnen zwecklos) und der Dienst funktioniert

# Kryptographie – Entscheidende Erkenntnisse (I)

- **Kerckhoffs-Prinzip** (formuliert 1883)

- Trennung von Algorithmus (Verfahren) und Schlüssel

**Algorithmus:** “Verschiebe Alphabet um eine best. Anzahl Positionen zyklisch nach links”

**Schlüssel:** die “bestimmte Anzahl Positionen” (bei Caesar: 3)

- Kerckhoffs-Prinzip: Geheimnis liegt im Schlüssel und nicht im Algorithmus bzw. „No security through obscurity“

- **One Time Pad – Shannon / Vernam**

- Nachweislich theoretisch sicher, jedoch praktisch kaum anwendbar (nur Rotes Telefon).

- **Shannons Konzepte: Konfusion und Diffusion**

- Zusammenhang zwischen M, C und K möglichst komplex
- Jedes Chiffrezeichen sollte von möglichst vielen Klartextzeichen und vom gesamten Schlüssel abhängen
- „Avalanche effect“ (kleine Änderung, große Wirkung)

- **Trapdoor Function** (Falltür, Einweg-Funktion, ...)

- in einer Richtung schnell, in die andere (ohne Wissen) langsam
- nur mit dem Geheimnis hat man Zugang zur Falltür



# Schlüsselverteilungsproblem

## Schlüsselverteilung bei symmetrischer Verschlüsselung

Wenn **2 Personen** miteinander mit einer symmetrischen Verschlüsselung kommunizieren, brauchen sie **einen gemeinsamen und geheimen Schlüssel**.

Wenn bei  $n$  Personen jeder mit jedem geheim kommunizieren möchte, dann braucht man  $S_n = n(n-1) / 2$  Schlüssel.

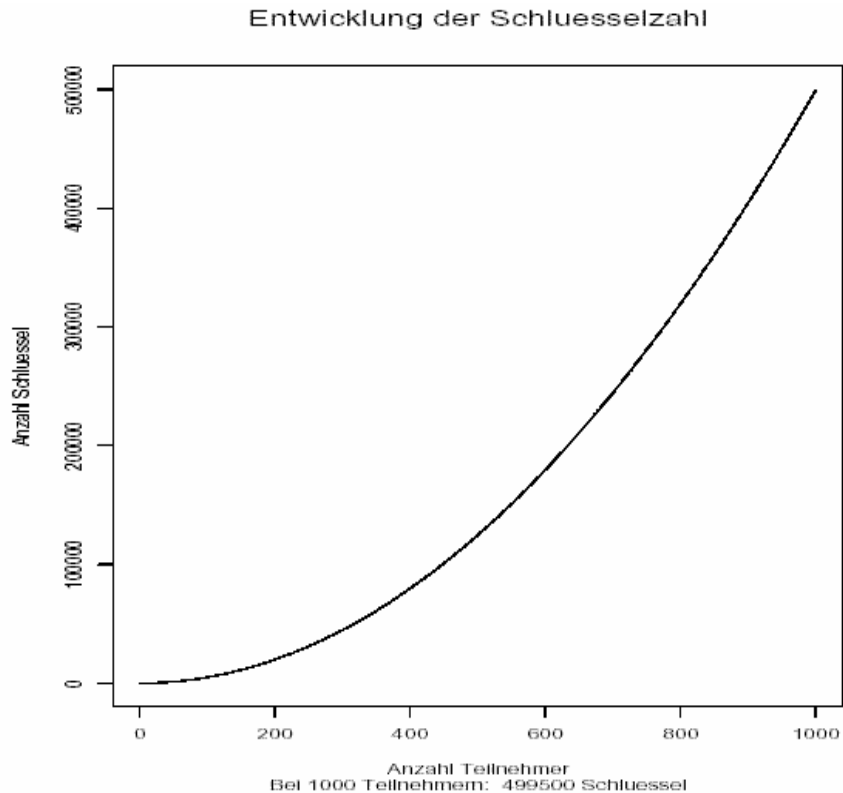
Das sind bei

$n = 100$  Personen bereits

$S_{100} = 4.950$  Schlüssel, bei

$n = 1.000$  Personen sind es

$S_{1000} = 499.500$  Schlüssel.



# Kryptographie – Entscheidende Erkenntnisse (II)

## Lösung des Schlüsselverteilungsproblems durch asymmetrische Kryptographie

### ■ Asymmetrische Kryptographie

- **Jahrhunderte lang glaubte man:** Sender und Empfänger brauchen dasselbe Geheimnis.
- **Neu:** Jeder Teilnehmer hat ein Schlüsselpaar („Lösung“ des Schlüsselverteilungsproblems)

### ■ Asymmetrische Verschlüsselung

- „Jeder kann ein Vorhängeschloss einschnappen lassen oder einen Brief in einen Kasten werfen“, Schatzkistengeschichte als Denksportaufgabe
- MIT, 1977: Leonard Adleman, Ron Rivest, Adi Shamir (bekannt durch RSA)
- GCHQ Cheltenham, 1973: James Ellis, Clifford Cocks (am 18.12.1997 öffentlich zugegeben)

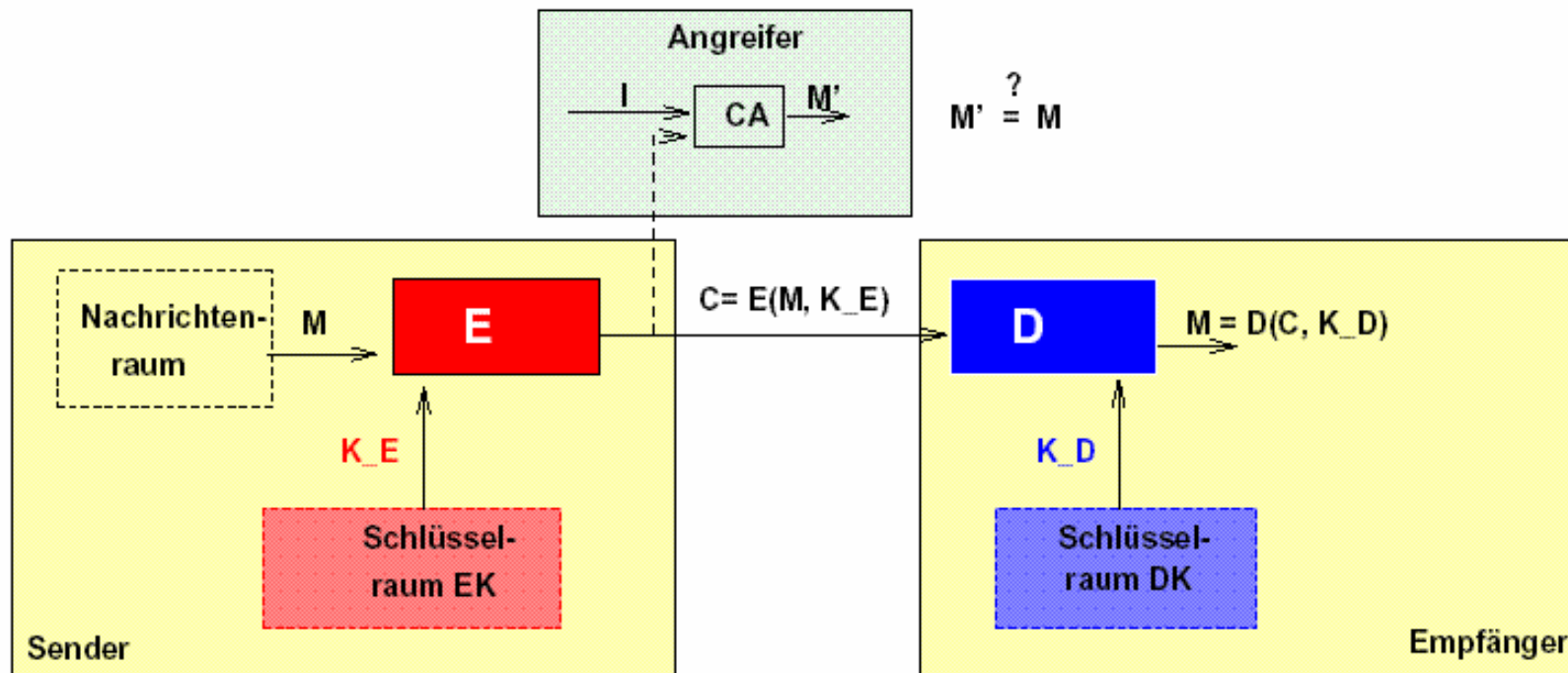
### ■ Schlüsselverteilung

- Stanford, 1976: Whitfield Diffie, Martin Hellman, Ralph Merkle (Diffie-Hellman Key Exchange)
- GCHQ Cheltenham, 1975: Malcolm Williamson

*Sicherheit in offenen Netzen (wie dem Internet) wäre  
ohne asymmetrische Kryptographie extrem teuer und komplex !*

# Durchführung von Ver- und Entschlüsselung

## Symmetrische und asymmetrische Verschlüsselung



(a) Symmetrische :  $K_E = K_D$  geheim! z.B. DES

(b) Asymmetrische :  $K_E \neq K_D$  z.B. RSA

öffentlich      geheim!

# Kryptographie – Entscheidende Erkenntnisse (III)

## Steigende Bedeutung der Mathematik und der Informationstechnologie

- **Moderne Kryptographie** basiert stärker auf **Mathematik**
  - Trotzdem gibt es weiter symmetrische Verfahren wie den AES (bessere Performance und kürzere Schlüssellängen als die auf rein mathematischen Problemstellungen beruhenden asymmetrischen Verfahren).
- Die Sicherheit praktisch eingesetzter Verfahren hängt entscheidend vom Stand der **Mathematik** und der **Informationstechnologie** (IT) ab.
  - Berechnungskomplexität (d.h. Rechenaufwand in Abhängigkeit von der Schlüssellänge, Speicherplatzbedarf, Datenkomplexität)
    - > siehe aktuell RSA: Bernstein, TWIRL-Device, RSA-200 (CrypTool-Skript, Kap. 4.11.3)
  - Sehr hohe Intensität in der aktuellen Forschung:  
Faktorisierung, nicht-parallelisierbare Algorithmen (wegen Quantencomputing), besseres Verständnis von Protokoll-Schwächen und Zufallszahlengeneratoren, ...).
- Entscheidender Irrtum: „*Echte Mathematik*“ hat keine Auswirkungen auf den Krieg. (G.H. Hardy, 1940)
- Hersteller entdecken **Sicherheit** als ein zentrales **Kaufkriterium**

# Was bietet CrypTool?

E-Learning

## 1. Was ist CrypTool?

- Freeware-Programm mit graphischer Oberfläche
- Kryptographische Verfahren anwenden und analysieren
- Sehr umfangreiche Online-Hilfe, ohne tieferes Kryptographiewissen verständlich
- Enthält fast alle State-of-the-art-Kryptographiefunktionen
- „Spielerischer“ Einstieg in moderne und klassische Kryptographie
- Kein „Hackertool“

## 2. Warum CrypTool?

- Ursprung im End-User Awareness-Programm einer Großbank
- Entwickelt in Kooperation mit Hochschulen → mediendidaktischer Anspruch
- Verbesserung der Lehre an Hochschulen und der betrieblichen Ausbildung

## 3. Zielgruppe

- Kernzielgruppe: Studierende der Informatik, Wirtschaftsinformatik, Mathematik
- Aber auch: Computernutzer und Anwendungsentwickler, Mitarbeiter, Schüler
- Voraussetzung: PC-Kenntnisse
- Wünschenswert: Interesse an Mathematik und Programmierung

# Inhalt des Programmpakets

komplett zweisprachig  
Deutsch  
Englisch

## CrypTool-Programm

- alle Funktionen integriert in *einem* Programm mit einheitlicher graphischer Oberfläche
- läuft unter Win32
- Kryptographie aus den Bibliotheken von Secude und OpenSSL
- Langzahlarithmetik per Miracl und GMP, Gitterbasenreduktion per NTL (V. Shoup)

## AES-Tool

- Standalone-Programm zur AES-Verschlüsselung (selbst extrahierend)

## Lernbeispiel

- „Der Zahlenhai“ fördert das Verständnis für Teiler und Primzahlen.

## Umfangreiche Online-Hilfe (HTML-Help)

- kontextsensitive Hilfe mit F1 für *alle* Programmfunktionen (auch auf Menüs)
- ausführliche Benutzungs-Szenarien (Tutorials) für viele Programmfunktionen

## Skript (.pdf-Datei) mit Hintergrundinformationen

- Verschlüsselungsverfahren • Primzahlen/Faktorisierung • Digitale Signatur
- Elliptische Kurven • Public Key-Zertifizierung • Elementare Zahlentheorie

## Zwei Kurzgeschichten mit Bezug zur Kryptographie von Dr. C. Elsner

- „Der Dialog der Schwestern“ (eine RSA-Variante als Schlüsselement)
- „Das chinesische Labyrinth“ (zahlentheoretische Aufgaben für Marco Polo)

# Funktionsumfang (I)

## Kryptographie

### Verschlüsselungsklassiker

- Caesar
- Vigenère
- Hill
- Homophone Substitution
- Playfair
- ADFGVX
- Addition
- XOR
- Vernam
- Permutation
- Solitaire

### Zum besseren Nachvollziehen von Literaturbeispielen ist

- Alphabet wählbar
- Behandlung von Leerzeichen etc. einstellbar

## Kryptoanalyse

### Angriffe auf klassische Verfahren

- Ciphertext Only
  - Caesar
  - Vigenère
  - Addition
  - XOR
  - Substitution
  - Playfair
- Known Plaintext
  - Hill
- Manuell (unterstützt)
  - Monoalphabetische Substitution
  - Playfair
  - ADFGVX
  - Solitaire

### Unterstützende Analyseverfahren

- Entropie, gleitende Häufigkeit
- Histogramm, n-Gramm-Analyse
- Autokorrelation
- Perioden
- Zufallszahlanalyse
- Base64 / UU-Encode

# Funktionsumfang (II)

## Kryptographie

### Moderne symmetrische Verschlüsselung

- IDEA, RC2, RC4, RC6, DES, 3DES
- Serpent, Twofish
- AES-Kandidaten der letzten Auswahlrunden
- AES (=Rijndael)

### Asymmetrische Verschlüsselung

- RSA mit X.509-Zertifikaten
- RSA-Demonstration
  - zum Nachvollziehen von Literaturbeispielen
  - Alphabet und Blocklänge einstellbar

### Hybridverschlüsselung (RSA + AES)

- visualisiert als interaktives Datenflussdiagramm

## Kryptoanalyse

### Brute-force-Angriff auf symmetrische Algorithmen

- für alle Algorithmen
- Annahme: Entropie des Klartextes klein

### Angriff auf RSA-Verschlüsselung

- Faktorisierung des RSA-Moduls
- Gitterreduktions-basierte Angriffe

### Angriff auf Hybridverschlüsselung

- Angriff auf RSA oder
- Angriff auf AES (Seitenkanalangriff)

# Funktionsumfang (III)

## Kryptographie

### Digitale Signatur

- RSA mit X.509-Zertifikaten
  - Signatur zusätzlich visualisiert als interaktives Datenflussdiagramm
- DSA mit X.509-Zertifikaten
- Elliptic Curve DSA, Nyberg-Rueppel

### Hashfunktionen

- MD2, MD4, MD5
- SHA, SHA-1, RIPEMD-160

### Zufallsgeneratoren

- Secude
- $x^2 \bmod n$
- Linearer Kongruenzgenerator (LCG)
- Inverser Kongruenzgenerator (ICG)

## Kryptoanalyse

### Angriff auf RSA-Signatur

- Faktorisierung des RSA-Moduls
- praktikabel bis ca. 250 bits bzw. 75 Dezimalstellen (auf Einzelplatz-PC)

### Angriff auf Hashfunktion / digitale Signatur

- Generieren von Hash-Kollisionen für ASCII-Texte (Geburtstagsparadox) (bis 40 bits in etwa 5 min)

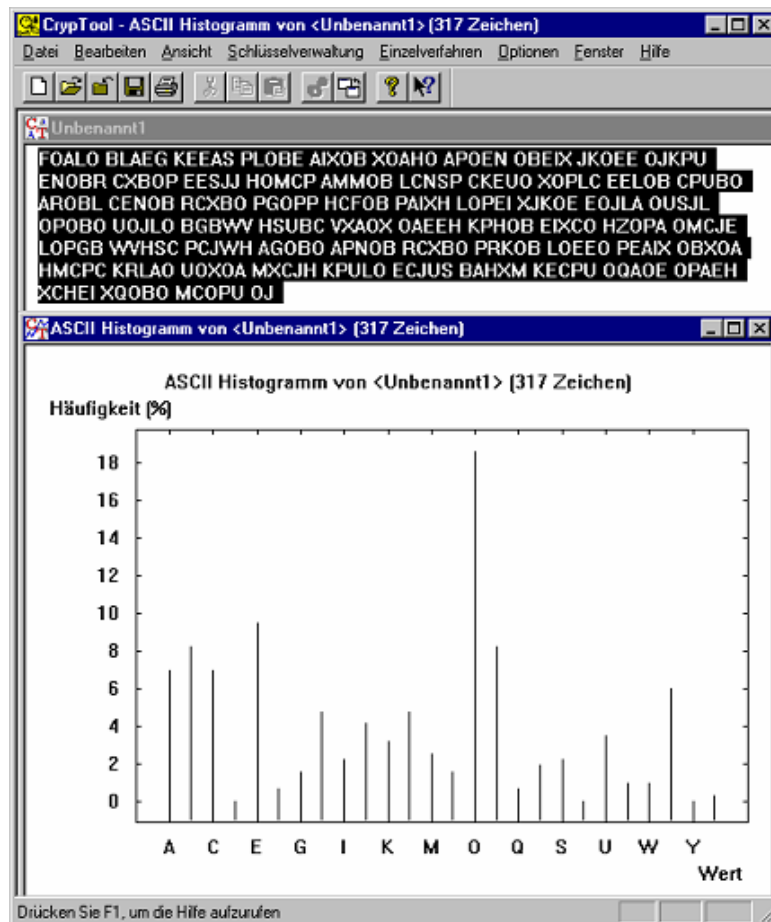
### Analyse von Zufallsdaten

- FIPS-PUB-140-1 Test-Batterie
- Periode, Vitany, Entropie
- Gleitende Häufigkeit, Histogramm
- n-Gramm-Analyse, Autokorrelation
- ZIP-Kompressionstest

# Codestruktur analysieren

## Chiffratgüte, Anzahl Einzelzeichen, n-Gramme, Entropie

- z.B. mit CrypTool per Analyse / Allgemein / ...



The dialog box titled 'Entropie <Unbenannt1>' displays the entropy analysis results. It states that the analyzed document contains 23 of the 26 characters of the installed alphabet. The entropy is 3.99, which is close to the maximum possible entropy of 4.70.

Das analysierte Dokument enthält 23 der 26 Zeichen des eingestellten Alphabets.

**Seine Entropie beträgt 3.99  
(maximal mögliche Entropie 4.70).**

OK

The dialog box titled 'N-Gramm Liste' shows a list of N-grams. The 'Auswahl' section has radio buttons for 'Histogramm', 'Digramm', 'Trigramm', and '4 -Gramm'. The 'Anzeige der' field is set to '26' and is labeled 'häufigsten N-Gramme'. The 'Liste bestimmen' button is highlighted. The 'Liste speichern' and 'Schließen' buttons are also visible. The table on the right lists the N-grams and their frequencies.

Nr.	Zeichen...	Häufigkeit in %
1	O	18.612
2	E	9.464
3	B	8.202
4	P	8.202
5	A	6.940
6	C	6.940
7	X	5.994
8	H	4.732
9	L	4.732
10	J	4.101
11	U	3.470
12	K	3.155
13	M	2.524
14	I	2.208
15	S	2.208
16	R	1.893
17	G	1.577
18	N	1.577
19	V	0.946
20	W	0.946
21	F	0.631
22	Q	0.631
23	Z	0.315

## Herausforderungen für den Programmierer (Beispiele)

### 1. Viele Funktionen parallel laufen lassen

- Bei der Faktorisierung laufen die verschiedenen Algorithmen in Threads

### 2. Hohe Performance

- Bei der Anwendung des Geburtstagsparadoxons zum Finden von Hashkollisionen oder bei der Brute-Force-Analyse

### 3. Speicherbeschränkung beachten

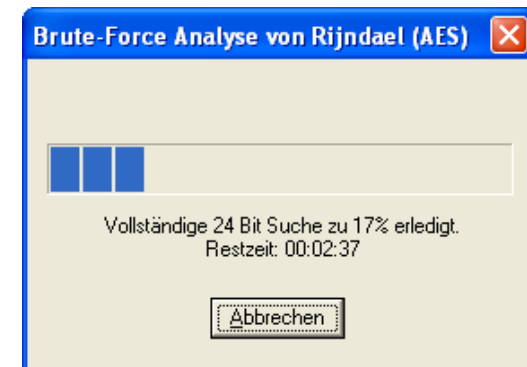
- Beim Floyd-Algorithmus (Mappings für das Finden von Hashkollisionen) oder beim Quadratic Sieve.

### 4. Zeitmessung und -abschätzung

- Ausgabe der Elapsed Time bei Brute-Force

### 5. Wiederverwendung / Integration

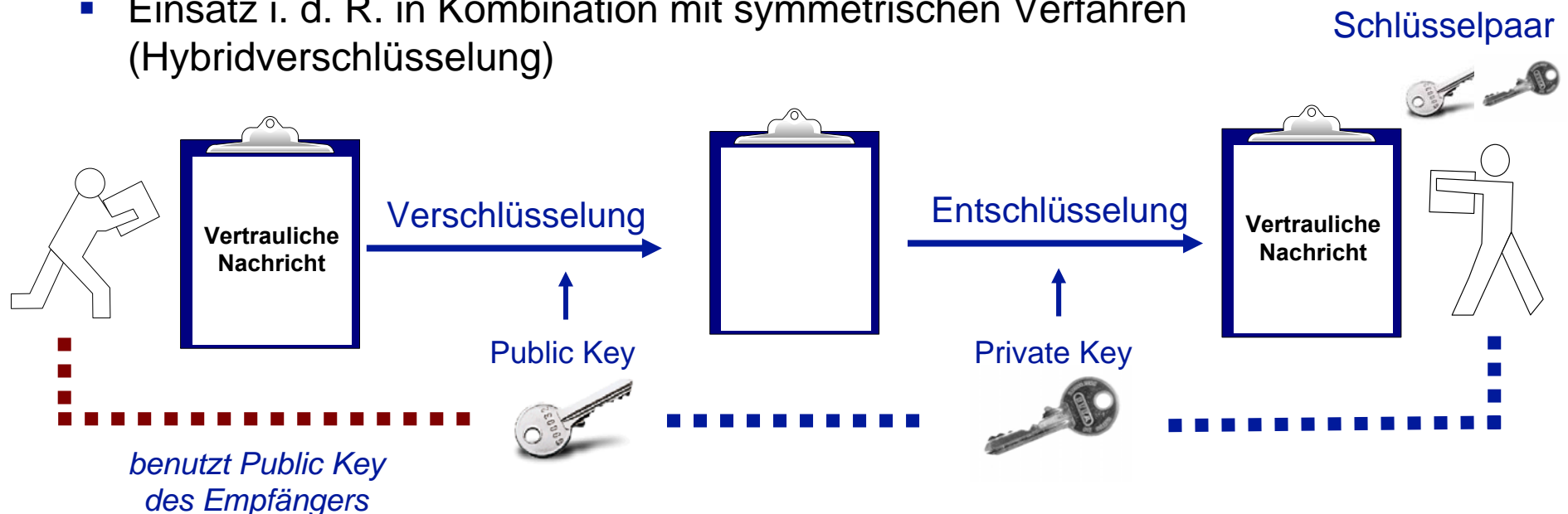
- Masken zur Primzahlgenerierung
- RSA-Kryptosystem (schaltet nach erfolgreicher Attacke von der Ansicht des Public-Key-Anwenders zur Ansicht des Private-Key-Besitzers)



# Anwendungsbeispiele (I)

## Verschlüsselung mit RSA

- **Grundlage** für z.B. SSL-Protokoll (Zugriff auf gesicherte Web Seiten)
- **Asymmetrische Verschlüsselung mit RSA**
  - Jeder Benutzer hat ein Schlüsselpaar – einen öffentlichen und einen privaten.
  - Sender verschlüsselt mit dem öffentlichen Schlüssel (*public key*) des Empfängers.
  - Empfänger entschlüsselt mit seinem privaten Schlüssel (*private key*).
- Einsatz i. d. R. in Kombination mit symmetrischen Verfahren (Hybridverschlüsselung)



# Anwendungsbeispiele (I)

## Verschlüsselung mit RSA – Mathematischer Hintergrund / Verfahren

- Öffentlicher Schlüssel (public key):  $(n, e)$
- Privater Schlüssel (private key):  $(d)$

### wobei:

$p, q$  große zufällig gewählte Primzahlen mit  $n = p \cdot q$ ;

$d$  wird unter den NB  $\text{ggT}[\varphi(n), e] = 1$ ;  $e \cdot d \equiv 1 \pmod{\varphi(n)}$ ; bestimmt.

Ver- und Entschlüsselungs-Operation:  $(m^e)^d \equiv m \pmod{n}$

- $n$  ist der Modulus, dessen Schlüssellänge beim RSA-Verfahren angegeben wird.
- $\text{ggT}$  = größter gemeinsamer Teiler.
- $\varphi(n)$  ist die Eulersche Phi-Funktion.

### Vorgehen:

- Transformation von Nachrichten in binäre Repräsentation
- Nachricht  $m = m_1, \dots, m_k$  blockweise verschlüsseln, wobei für alle  $m_j$  gilt:  
 $0 \leq m_j < n$ ; also maximale Blockgröße  $r$  so, dass gilt:  $2^r \leq n$

# Anwendungsbeispiele (I)

## Primzahltests – Für RSA werden große Primzahlen benötigt.

- Schnelle probabilistische Tests
- Deterministische Tests

Die bekannten Primzahltest-Verfahren können für große Zahlen viel schneller testen, ob die Zahl prim ist, als die bekannten Faktorisierungsverfahren eine Zahl ähnlicher Größenordnung in ihre Primfaktoren zerlegen können.

Für die AKS-Methode wurde die GMP-Bibliothek (**GNU Multiple Precision Arithmetic Library**) in CrypTool integriert.

The screenshot shows a window titled "Primzahltest" with a close button in the top right corner. The main text area contains the following text: "Um zu testen, ob eine Zahl eine Primzahl ist, gibt es verschiedene Verfahren (Mathematiker sagen auch, man testet, ob die Zahl prim ist). Am häufigsten werden die probabilistischen Verfahren verwendet: Sie sind sehr schnell, geben aber nur mit einer bestimmten (einstellbar kleinen) Wahrscheinlichkeit an, ob die Zahl prim ist. Daneben gibt es noch deterministische Verfahren: Wenn diese ein Ergebnis liefern, ist es mit 100% Wahrscheinlichkeit korrekt (aus mathematischer Sicht)." Below this text is a section titled "Algorithmen zum Primzahltest" with four radio button options: "Miller-Rabin-Test" (selected), "Fermat-Test", "Solovay-Strassen-Test", and "AKS-Test (deterministisches Verfahren)". Below the algorithm list is a section titled "Primzahltest" containing a "Zahl aus Datei laden" button, a text input field with "2^255-1", and an "Ergebnis" field with a red "X" icon and the text "Die Zahl ist keine Primzahl: 578960446186580977117854925043439539266349". At the bottom of the window are two buttons: "Zahl testen" and "Abbrechen".

# Anwendungsbeispiele (I)

## Hybridverschlüsselung und Digitale Zertifikate

- **Hybridverschlüsselung** – Kombination aus asymmetrischer und symmetrischer Verschlüsselung
  1. Generierung eines zufälligen symmetrischen Sitzungs-Schlüssels (Session Key)
  2. Der Session Key wird – geschützt mit dem asymmetrischen Schlüssel – übertragen
  3. Die Nachricht wird – geschützt mit dem Session Key – übertragen
- **Problem:** Man-in-the-middle Angriffe: Gehört der öffentliche Schlüssel (Public Key) des Empfängers auch wirklich dem Empfänger?
- **Lösung: Digitale Zertifikate** – Eine zentrale Instanz (z.B. VeriSign, Deutsche Bank PKI), der alle Benutzer trauen, garantiert die Authentizität des Zertifikates und des darin enthaltenen öffentlichen Schlüssels (analog zu einem vom Staat ausgestellten Personalausweis).
- **Hybridverschlüsselung auf Basis von digitalen Zertifikaten** ist die Grundlage für sichere elektronische Kommunikation (z.B. SSL):
  - Internet Shopping und Online Banking
  - Sichere eMail

- **OpenSSL mit Delphi (DataWebSecure)**

- **Hybride Chiffrierung:**

Symmetrisch: Session Key (Master Secret): AES 256

Asymmetrisch: - RSA = [(public)+(private)=(certificate)]

Asymmetrisch schützt Symmetrischen Key !

- **Integrität: MAC SHA1 (hash)**

- **Authentifizierung (Digitale Signatur), Verbindlichkeit**

# Anwendungsbeispiele (I)

## Gesicherte Online-Verbindung mit SSL und Zertifikaten

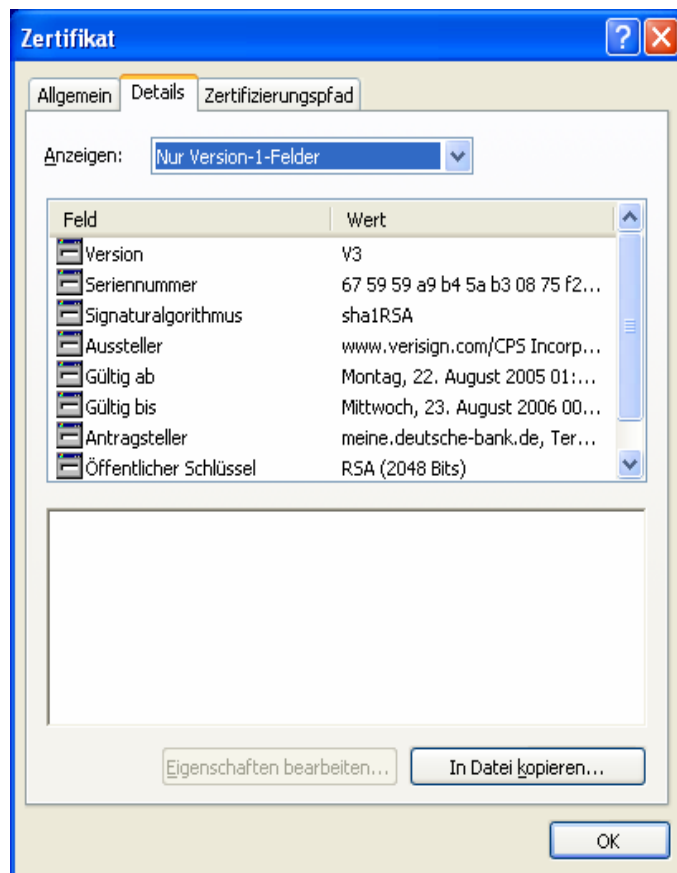
The image shows two overlapping windows. The left window is a Microsoft Internet Explorer browser displaying the Deutsche Bank Online-Banking login page. The address bar shows 'https://meine.deutsche-bank.de/mod/WebObjects/dbpbc.woa'. The page has a 'Herzlich willkommen!' greeting and a login form with fields for 'Filiale', 'Konto', 'Unterkonto', and 'PIN'. A 'Login ausführen' button is visible. Below the form, there is a notice about the 'Start des neuen, erweiterten TAN-Verfahrens 21.02.2006'. The right window is a 'Zertifikat' (Certificate) dialog box. It shows 'Zertifikatsinformationen' for a certificate issued to 'meine.deutsche-bank.de' by 'www.verisign.com/CPS Incorp.by Ref. LIABILITY LTD.(c)97 VeriSign'. The validity period is from 22.08.2005 to 23.08.2006. Buttons for 'Zertifikat installieren...' and 'Ausstellereklärung' are present.

D.h. die Verbindung ist (zumindest einseitig) authentisiert und der übertragene Inhalt wird stark verschlüsselt.



# Anwendungsbeispiele (I)

## Attribute / Felder von Zertifikaten



### Grundlegende Attribute / Felder

- Aussteller (z.B. VeriSign)
- Antragsteller
- Gültigkeitszeitraum
- Seriennummer
- Zertifikatsart / Version (X.509v3)
- Signaturalgorithmus
- Öffentlicher Schlüssel (und Verfahren)

Öffentlicher Schlüssel



# Anwendungsbeispiele (II)

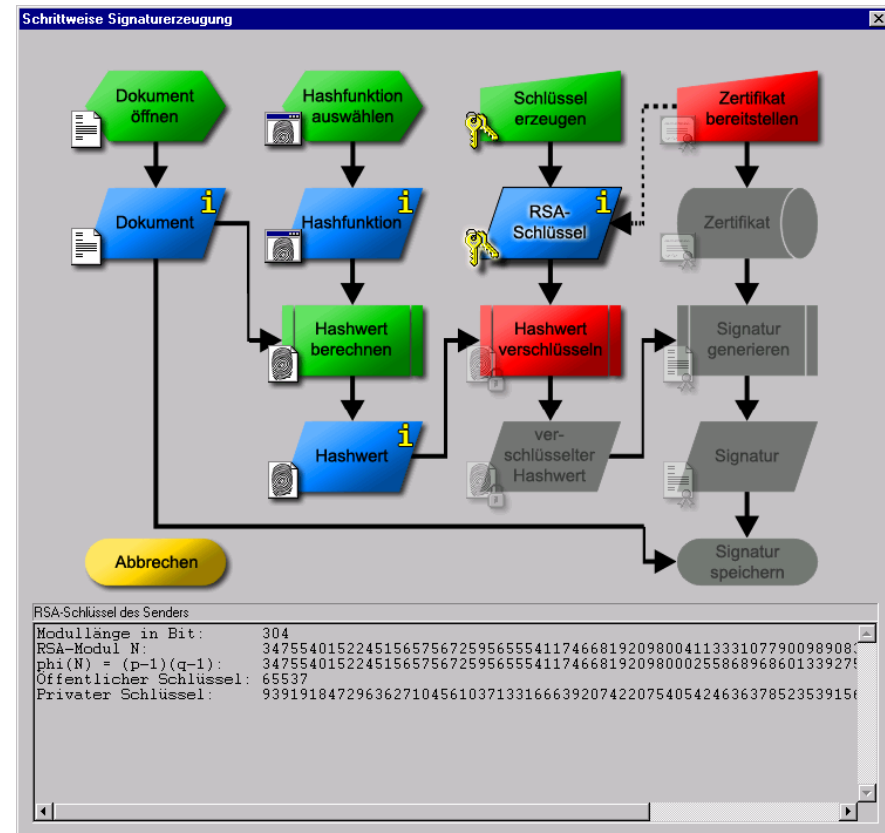
## Elektronische Signatur visualisiert

### Elektronische Signatur

- Wird immer wichtiger durch
  - Gleichstellung mit manueller Unterschrift (Signaturgesetz)
  - Zunehmenden Einsatz in Wirtschaft, durch den Staat und privat
- Wer weiß, wie sie funktioniert?

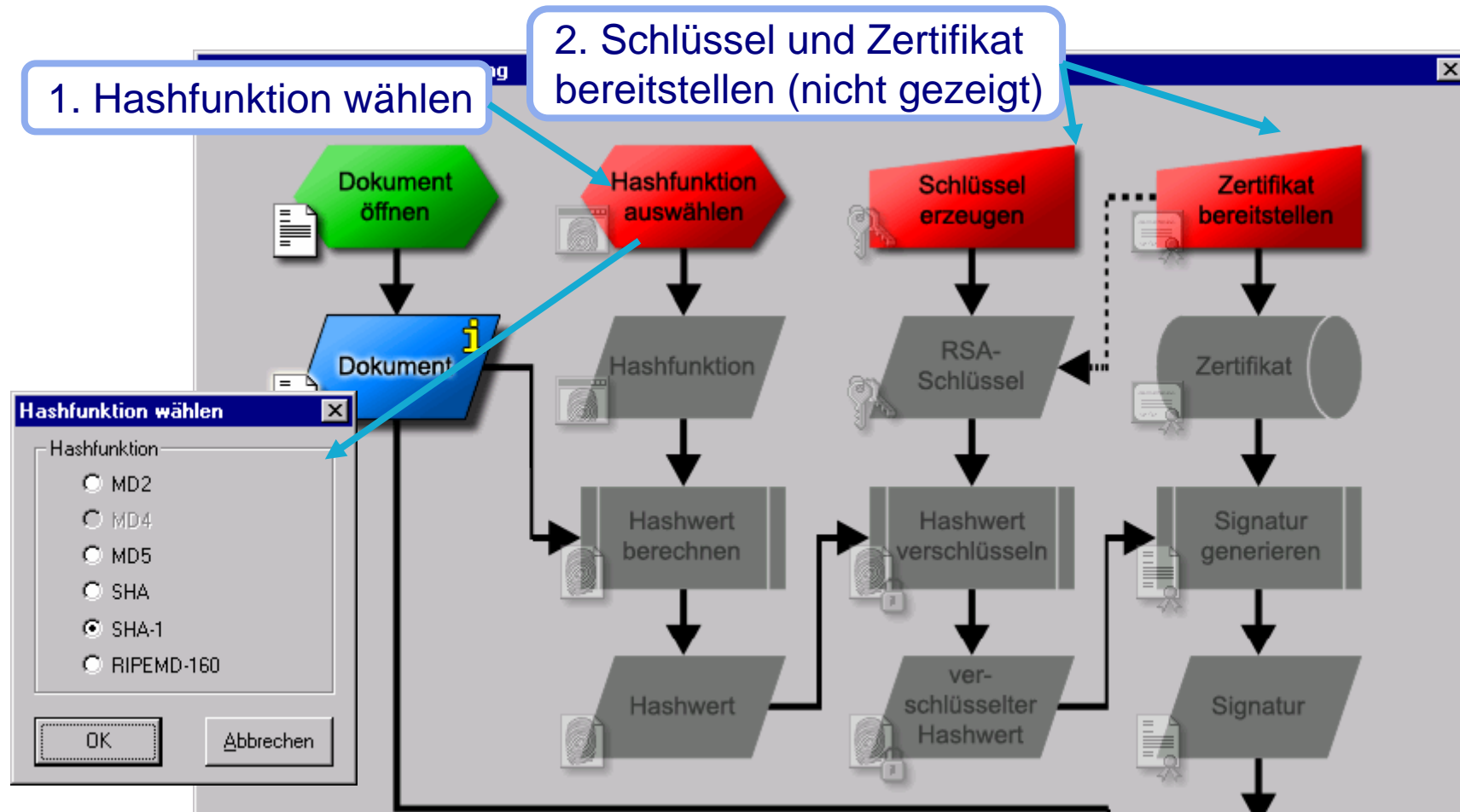
### Visualisierung in CryptTool

- Siehe Menü „Digitale Signaturen/PKI“ \ „Signaturdemo (Signaturerzeugung)“
- Interaktives Datenflussdiagramm
- Ähnlich wie die Visualisierung der Hybridverschlüsselung



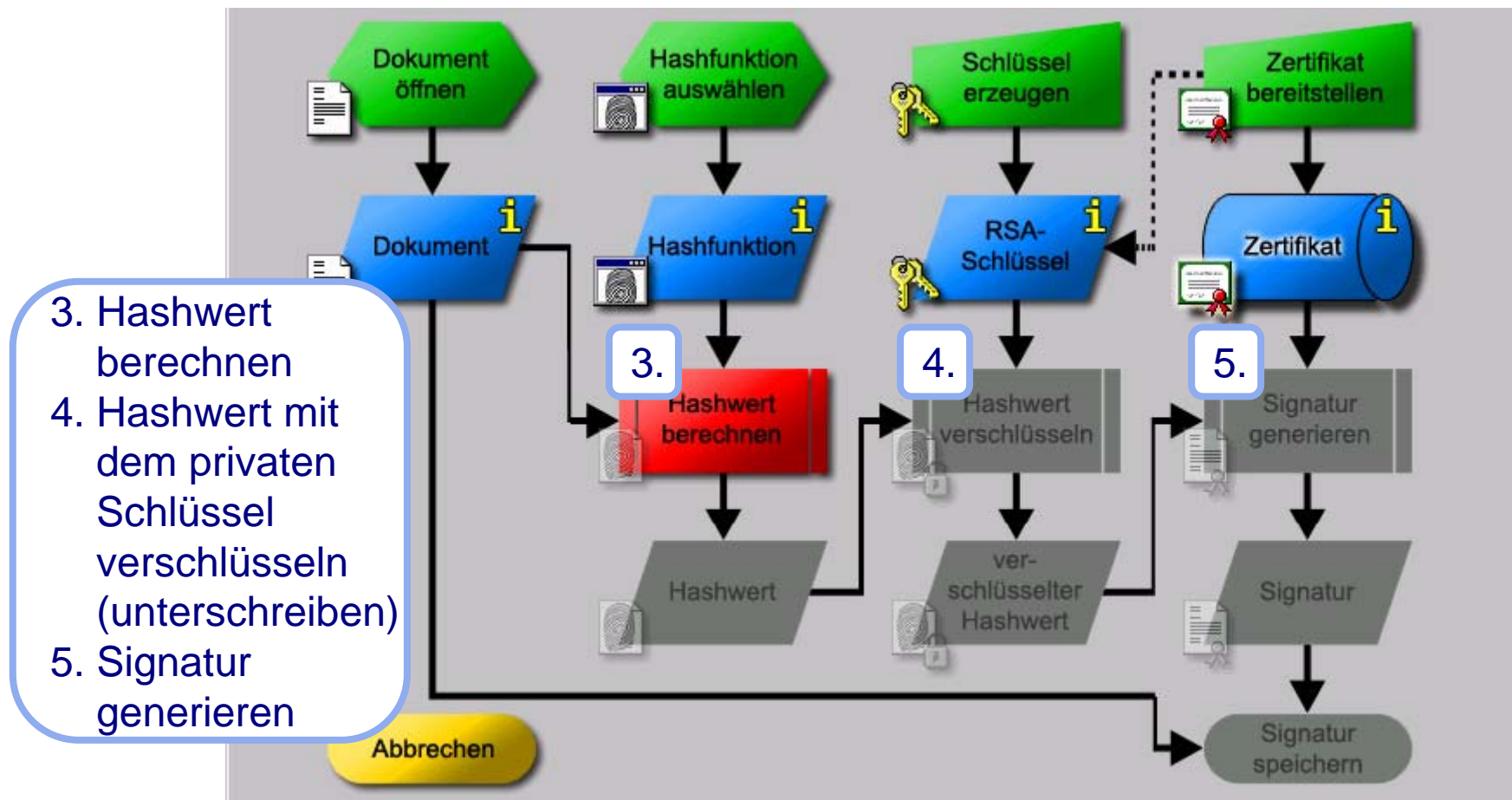
# Anwendungsbeispiele (II)

Elektronische Signatur visualisiert: a) Vorbereitung



# Anwendungsbeispiele (II)

## Elektronische Signatur visualisiert: b) Kryptographie



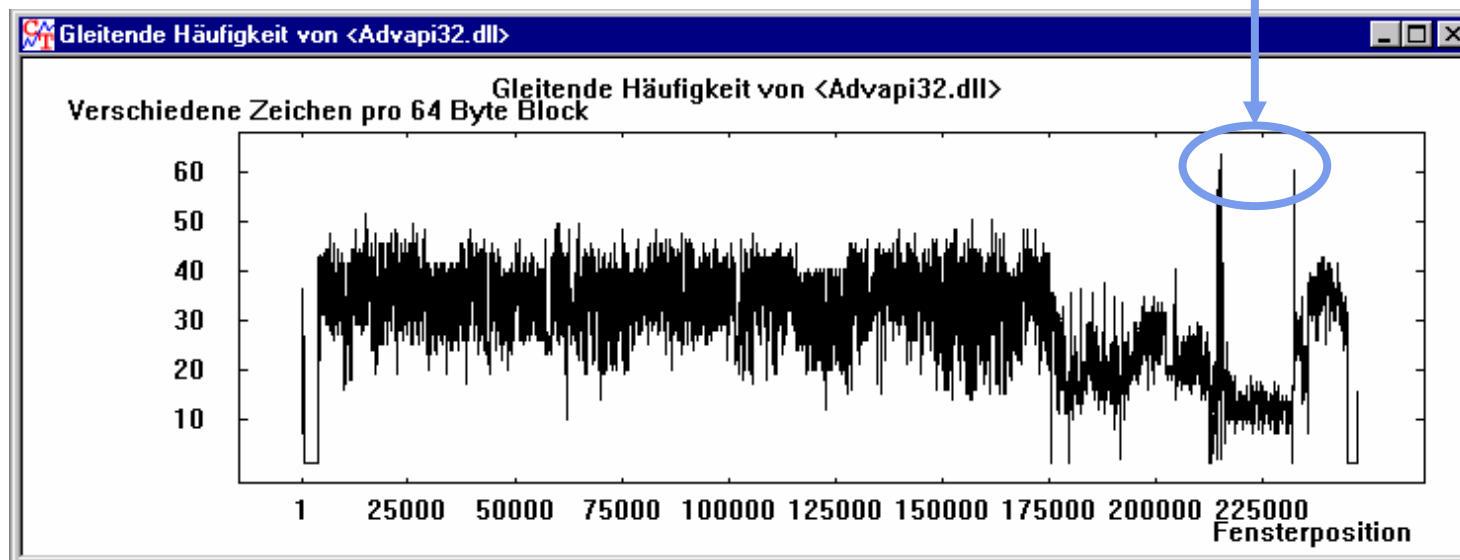
# Anwendungsbeispiele (VI)

## Auffinden von Schlüsselmaterial

Die Funktion „Gleitende Häufigkeit“ eignet sich zum Auffinden von Schlüsselmaterial und verschlüsselten Bereichen in Dateien.

### Hintergrund:

- diese Daten sind „zufälliger“ als Text oder Programmcode
- sie sind als Peak in der „gleitenden Häufigkeit“ zu erkennen
- Beispiel: der „NSAKEY“ in advapi32.dll



## Weitere Lektüre (auch als Einstieg in die Kryptologie)

- Simon Singh, *“Geheime Botschaften”*, 2000, Hanser [in Deutsch]
- ▣ Simon Singh, *“The Codebook”*, 1999, Doubleday [Englisches Original]
- ▣ **OpenSSL mit Delphi : <http://sourceforge.net/projects/delphiwebstart>**
- Max Kleiner, *“Bankraub Online”*, Output Nr.11 Oktober 1997, Output
- A. Beutelspacher / J. Schwenk / K.-D. Wolfenstetter, *“Moderne Verfahren der Kryptographie”*, 5. Auflage, 2004, Vieweg
- [HAC] Menezes, van Oorschot, Vanstone, *“Handbook of Applied Cryptography”*, 1996, CRC Press
- van Oorschot, Wiener, *“Parallel Collision Search with Application to Hash Functions and Discrete Logarithms”*, 1994
- **Vielfältige Krypto-Literatur** – siehe Links auf der CrypTool-Webseite sowie Quellenangaben in der Online-Hilfe (z.B. von Wätjen, Buchmann, Salomaa, Brands, Schneier, Shoup, ...)
- **Bedeutung der Kryptographie in dem breiteren Rahmen von IT-Sicherheit, Risikomanagement und organisatorischen Kontrollen**
  - Siehe z.B. Kenneth C. Laudon / Jane P. Laudon / Detlef Schoder, *“Wirtschaftsinformatik”*, 2005, Pearson, Kapitel 14
  - Siehe Wikipedia (<http://de.wikipedia.org/wiki/Risikomanagement>)