

# Kryptographie

Laborautomation WS'02/03  
Patrick Gleichmann



## 0. Was ist Sicherheit ?

---

Verstecken != Verschlüsseln

---

Was ist Sicherheit überhaupt ?

Dazu folgendes Beispiel:

Wenn man etwas in einen Safe steckt, um es vor anderen zu verbergen, dann ist dies lediglich ein Verstecken - dies garantiert jedoch nur eine geringe Sicherheit. Nun kann man sich vorstellen, das der Konstruktionsplan des Safes öffentlich ist und es ihn auch mehrfach gibt. Nun kann jeder Safe-Knacker in kurzer Zeit und nahezu ohne Anstrengung an die Informationen heran - die Sicherheit ist dahin.

Bei Verschlüsselung jedoch kann zum „Bauplan“ (= Algorithmus) sogar die Information selbst öffentlich sein - ohne den passenden Schlüssel ist sie jedoch wertlos.

Natürlich gibt es trotzdem noch Einsatzzwecke für das Verstecken von Informationen - Stichwort: Steganographie.

# Inhalt

---

1. Grundlagen
2. Historische Verschlüsselung
3. Einfache Algorithmen
4. Abläufe
5. Schlüsselverwaltung
6. Hinweis zu symmetrischen Algorithmen

- 
- 1.1 Warum Kryptographie
  - 1.2 Ver- und Entschlüsselung
  - 1.3 Ver- und Entschlüsselung mit Schlüssel(n)
  - 1.4 Schlüssel - Definition
  - 1.5 Schlüsselbasierte Algorithmen
  
  - 2.1 Substitution
  - 2.2 Transposition
  - 3.1 XOR
  - 3.2 One-Time-Pads
  
  - 4.1 Symmetrische Verschlüsselung
  - 4.2 Asymmetrische Verschlüsselung
  - 4.3 Hybrid-Verfahren
  
  - 5.1 Erzeugung
  - 5.2 Übermittlung
  - 5.3 Speicherung

# 1. Grundlagen



## 1.1 Warum Kryptographie

---

- Authentifizierung
- Integrität
- Verbindlichkeit

---

Außer der Verschlüsselung und damit der „Versiegelung“ der Information, gibt es noch drei weitere Punkte, warum man kryptographische Methoden - hier speziell digitale Signaturen - verwendet:

Authentifizierung:

Sicherstellen der Herkunft, bestimmen von wem die Information stammt. Natürlich darf auch niemand sich als jemand anderes ausgeben dürfen.

Integrität:

Es soll sichergestellt werden, dass die Information unverändert beim Adressaten ankommt, d.h. Veränderungen bzw. Ersetzen von Teilen soll unterbunden werden.

Verbindlichkeit:

Für den Informationsgeber entstehen natürlich auch Pflichten, so ersetzt sie - heute noch nicht rechtlich 100% realisiert - seine reale Unterschrift. Dies ist wichtig speziell für E-Commerce Geschäfte.

## 1.2 Ver- und Entschlüsselung

---



$$E(M) = C$$

$$D(C) = M$$

$$D(E(M)) = M$$

---

Ein Klartext  $M$ , z.B. ein Text oder Binärdaten, wird durch eine Verschlüsselung  $E$  (Encryption) unkenntlich gemacht, wobei man das Resultat Chiffre  $C$  nennt. Die Umkehrung, die Entschlüsselung  $D$  (Decryption), stellt den ursprünglichen Klartext wieder her, wobei  $D$  gleich  $E$  sein kann.

Dieses Verfahren hat nur noch historische Bedeutung, weil die Sicherheit lediglich vom Algorithmus selbst abhängt („eingeschränkter Algorithmus“). Sobald er bekannt wird, muß er ersetzt werden, da sonst keine Sicherheit mehr gewährleistet ist. Dies verhindert natürlich die Untersuchung durch Experten - gerade heute gelten offene Algorithmen als besonders sicher, weil sie mehrfach begutachtet wurden.

## 1.3 Ver- und Entschlüsselung mit Schlüssel(n)

---



$$E_K(M) = C$$

$$D_K(C) = M$$

$$D_K(E_K(M)) = M$$

---

Um die Abhängigkeit der Sicherheit vom Algorithmus zu lösen, wurden Schlüssel  $K$  eingeführt. Um eine Information ent- bzw. verschlüsseln zu können, muß zusätzlich zum Algorithmus auch der Schlüssel bekannt sein. Wichtig ist, das 2 verschiedene Schlüssel oder auch lediglich ein Schlüssel verwendet werden können - dies ist abhängig vom Verfahren.

Im folgenden werden nur noch schlüsselbasierte Algorithmen betrachtet.

## 1.4 Schlüssel - Definition

---

- Datenblock
- Passwort

---

Allgemein kann man Schlüssel in zwei Arten einteilen:

Datenblock:

Eine beliebige große Ansammlung von Bits, wobei sie an ein Speichermedium gebunden sind, zumeist als Datei. Als Medien kommen neben der üblichen Festplatte auch externe Medien wie z.B. Chipkarte oder Dongle in Betracht.

Passwort:

Ein Wort, was man meist nach Aufforderung des zuständigen Programmes, in eine Bildschirmmaske eingibt.

Beide werden begrenzt durch den sog. „Schlüsselraum“, welcher den gültigen Wertebereich als auch die Anzahl der Zeichen festlegt.

## 1.5 Schlüsselbasierte Algorithmen

---

- Symetrisch
  - 1 Schlüssel
- Asymetrisch
  - Private key
  - Public key

---

### Symetrisch:

Es existiert lediglich ein Schlüssel zur Kommunikation, welcher vor Kommunikationsbeginn vereinbart werden muß. Hier liegt auch das Problem von sym. Verschlüsselungsalgorithmen: Sobald der Schlüssel bekannt ist, kann die komplette Kommunikation „mitgehört“ werden.

### Asymetrisch:

Bei diesem Verfahren existieren zwei Schlüssel, ein Public Key zum verschlüsseln und ein Private Key zum Entschlüsseln. Ein populäres Beispiel für dieses Verfahren ist PGP („Pretty Good Privacy“).

## 2. Historische Chiffrierungen



## 2.1 Substitution

---

- Monoalphabetisch („A=B“)
- Homophone („A=1,13“)
- Polygraphisch („ABA=RTQ“)

---

Bei der Substitution werden immer Zeichen durch andere ersetzt, wobei man natürlich durch die Umkehrung die ursprüngliche Information erhält. Bei den folgenden Verfahren, muß natürlich jede Substitution als einzelne Regel schriftlich festgehalten werden.

### Monoalphabetisch:

Es werden lediglich einzelne Zeichen durch andere ersetzt, also aus ‚A‘ wird ‚B‘, etc. .

### Homophone:

Im Gegensatz zu Monoalphabetisch kann ein einzelnes Zeichen durch mehrere andere Zeichen repräsentiert werden.

### Polygraphisch:

Zuerst werden alle Zeichen zu gleichlangen Blöcken zusammengefaßt. Anschließend wendet man das Monoalphabetische Verfahren an.

Ein Beispiel für die mechanische Umsetzung stellt die Enigma dar, ein mehrfach hintereinander geschalteter Rotor. Die Substitution ist keineswegs starr, vielmehr wurde sie nach jeder Verwendung durch „weiterdrehen“ verändert.

## 2.2 Transposition

---

- Prinzip: Vertauschen der Reihenfolge
- Beispiel Spaltentransposition:

Klartext: LABORAUTOMATION FACHHOCHSCHULE WIESBADEN

```
L A B O R A U T
O M A T I O N F
A C H H O C H S
C H U L E W I E
S B A D E N
```

Chiffretext: LOACS AMCHB BAHUA OTHLD RIOEE AOCWN UNHI TFSE

---

Unter Transposition versteht man das schlichte Vertauschen der Reihenfolge der Zeichen innerhalb des Klartextes.

Ein sehr anschauliches Beispiel ist die Spaltentransposition:

Zuerst wird der Klartext in eine Matrize geschrieben, wobei Trennzeichen weggelassen werden. Der Chiffretext ist nun die Matrize spaltenweise abgelesen.

### 3. Einfache Algorithmen



## 3.1 XOR

---

Wahrheitstabelle ( $A = A \oplus B$ ) :

A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

Daraus folgender „Algorithmus“:

$$A \oplus K = B$$

$$B \oplus K = A$$

---

Ein sehr primitiver sym. Algorithmus, welcher aus der Wahrheitstabelle resultiert, wobei A und B jeweils einem Bit des Klartextes und des Schlüssels entsprechen.

Da der Schlüssel meist kürzer als der Klartext ist, muß er mehrfach benutzt werden. Dies erleichtert allerdings das Ermitteln der Schlüssellänge und damit das Knacken des Chiffers. Sobald die Schlüssellänge bekannt ist, genügt ein Shift des Chiffre und XOR mit dem unverschobenen Chiffre um den Klartext zu erhalten.

Trotz dieser Probleme wird er noch immer eingesetzt und als „vollkommen“ bezeichnet.

## 3.2 One-Time-Pads

---

- Variablen:
  - Klartext (M) mit N Zeichen
  - Schlüsselblock (S) mit N zufälligen Zeichen
- Algorithmus:
$$C_i = (M_i + S_i) \text{ modulo } 26 \quad \text{für } i = 1..N$$
- Fazit:  
„Perfekte“ Verschlüsselung

---

Das simple One-Time-Pad (de.: Einmalblock) Verfahren ist sowohl simpel als auch perfekt. Zu jedem Klartext-Zeichen existiert ein entsprechendes Schlüsselblock-Zeichen, das Chiffre-Zeichen ergibt sich einfach aus der Verknüpfung beider Zeichen.

Das Problem besteht in der kompletten Abhängigkeit des Chiffre zum Schlüsselblock. So muß parallel zum Chiffre auch der Schlüsselblock übertragen werden. Fehlt in der Übertragung auch nur 1 Bit, so werden die kompletten Daten wertlos.

Nicht verschweigen darf man auch doppelte Bandbreitenauslastung.

Anwendung besonders durch den russ. Geheimdienst während des kalten Kriegs.

## 4. Abläufe



## 4.1 Symmetrische Verschlüsselung

---

1. Alice und Bob einigen sich auf ein Kryptosystem.
2. Sie vereinbaren einen Schlüssel
3. Alice verschlüsselt die Nachricht mit dem vereinbarten Algorithmus und Schlüssel.
4. Alice sendet die verschl. Nachricht an Bob.
5. Bob entschlüsselt das ganze mit dem vereinbarten Algorithmus und Schlüssel.

---

### Lauschen:

Eve, eine eifersüchtige Rivalin von Alice, möchte wissen, was Alice und Bob sich erzählen. Dazu versucht sie bei Punkt 2 auch in Besitz des Schlüssels zu kommen. Wenn ihr dies gelingt, so kann sie die restliche Kommunikation mitlesen.

### Modifikation:

Nicht nur Eve ist eifersüchtig, sondern auch die böswillige Mallory. Diese möchte nur mitlesen, sondern die für Bob bestimmten Nachrichten verändern. Dies erreicht sie, indem sie „Man-in-the-Middle“ wird, d.h. den Schlüssel und die Nachrichten von Alice empfängt und dann die veränderten Nachrichten an Bob weiterschickt. Dabei kommuniziert sie mit Alice über den Alice/Bob-Schlüssel und mit Bob über einen mit diesem vereinbarten - eventl. unterschiedlichen - Schlüssel.

Fazit: Der 2. Punkt muß noch sicher gemacht werden.

## 4.2 Asymetrische Verschlüsselung

---

1. Alice und Bob einigen sich auf ein Kryptosystem
2. Alice bekommt Bob's Public Key
  - a) Von Bob direkt
  - b) Schlüsseldatenbank
3. Alice verschlüsselt die Nachricht mit Bob's Public Key.
4. Alice sendet die verschlüsselte Nachricht an Bob.
5. Bob entschlüsselt sie mit mit seinem private Key.

---

Lauschen und Verändern ist bei den asymmetrischen (auch „Public Key“) Verfahren unmöglich - Folge: Eve und Mallory müssen ohne Informationen auskommen.

Das Lauschen wird durch die Verschlüsselung unterbunden, während das Verändern durch eine dig. Signature von Alice verhindert wird.

## 4.3 Hybrid-Verfahren

---

1. Bob sendet Alice seinen Public Key.
2. Alice generiert einen Sitzungsschlüssel  $K$ , verschlüsselt ihn mit Bobs Schlüssel und schickt in an Bob.
3. Bob entschlüsselt den Schlüssel mit seinem Private Key.
4. Die Kommunikation „läuft“ über  $K$ .

---

Die asym. Verschlüsselung hat nur einen Nachteil: Sie ist aufgrund ihres Aufbaus (u.a. Schlüssellänge) sehr zeitintensiv, was sie natürlich für Realzeitdaten disqualifiziert.

Also benutzt man für die eigentliche Verschlüsselung der Daten einen symmetrischen Algorithmus, wobei das Problem der Schlüsselübertragung wiederum durch eine asym. Verschlüsselung des sym. Schlüssels gelöst wird.

## 5. Schlüsselverwaltung



## 5.1 Erzeugung

---

- Schlechte Schlüsselwahl
- Reduzierter Schlüsselraum
- Zufällige Schlüssel
- Pass-Phrasen

---

Was beim Erstellen eines Schlüssels zu beachten ist:

Schlechte Schlüsselwahl:

Zu einfache Schlüssel, also z.B. Name der Ehefrau, Posterüberschrift an der Wand, etc., hindern oftmals keinen an die Informationen heranzukommen. Deshalb gibt es mittlerweile Richtlinien, wie ein Schlüssel gewählt werden soll(te).

Reduzierter Schlüsselraum:

Bei manchen Software-Implementierungen ist die Anzahl der möglichen Schlüssel durch schlechtes Design beschränkt. Besonders häufig ist das nur ASCII-Zeichen akzeptiert werden - Folge: 7 anstelle von 8 Bit.

Zufällige Schlüssel:

Der Schlüssel wird ohne Zutun des Benutzers generiert. Leider kann ein Computer keine absolut beliebige Zahlenfolge erzeugen.

Passphrase:

Da einfache Schlüssel verboten sind, man sich jedoch komplizierte nicht merken kann, gibt es das Passphrase-Prinzip:

Der Benutzer merkt sich einen beliebig langen Satz aus welchem dann mittels Key-Ch crunching der eigentliche Schlüssel produziert wird.

## 5.2 Übermittlung

---

- Manuelle Übergabe („Diskette“)
- Key-Encryption Schlüssel
- Schlüsseldatenbank

---

Will man nun das sich mehrere Personen einen Private Key teilen („Gruppen-Schlüssel“), so kann die Übermittlung auf einem Medium geschehen oder aber der Schlüssel wird selbst durch einen anderen Schlüssel (Key-Encryption Key) geschützt. Problem bei letzterem ist wieder die Übertragung des Key-Encryption Key.

Bei dem Public Key ergibt sich das Problem nicht - man kann sich ihn problemlos aus einer Schlüsseldatenbank holen. Die Zertifizierung, ob nun der Schlüssel wirklich vom Betreffenden stammt, der Public Keys übernimmt man selbst, indem man den Fingerprint (ein Hash über den Key) auf nicht digitalem Weg (z.B. telefonisch) mit dem Eigentümer oder einer anderen vertrauenswürdigen Person, die bereits den Schlüssel zertifiziert hat, überprüft. Es entsteht ein „Web of trust“.

Zusätzlich existieren noch Zertifizierungsstellen, z.B. c't Crypto-Kampagne, welche geprüfte Schlüssel erstellen.

## 5.3 Speicherung

---

- „Merken“
- Externes Medium (z.B. „Magnetstreifen“)
- Key Encryption Schlüssel

---

Die Speicherung des priv. Schlüssels ist nahezu analog zur Übermittlung (5.2). Lediglich das Merken des Schlüssels ist eine weitere Möglichkeit.

## 6. Symetrische Algorithmen

---

- Stromalgorithmen
- Blockalgorithmen

---

Symetrische Algorithmen lassen sich in zwei Typen kategorisieren:

Stromalgorithmen:

Der Klartext wird als eine fortlaufende Bit-Kette behandelt, d.h. der Algorithmus ist zeitabhängig.

Blockalgorithmus:

Mehrere Bits (meist 64) werden zu Blöcken zusammengefaßt und dann chiffriert. Die Transformation ist bei allen Blöcken gleich - keine zeitliche Varianz.

## Literaturliste

---

- Schneider, Bruce: „Angewandte Kryptographie“, Addison-Wesley
  - Beutelspacher, Albrecht: „Kryptologie“, Vieweg
-