Informatik 6ai Schuljahr 2024/2025 als PDF exportieren

# Informatik 6. Klasse - Schuljahr 2024/25

## Lehrplan

- Lehrplaninhalte

## Themengebiete

- 7) Web Design Grundlagen (HTML, CSS)
- 8) Netzwerksicherheit

## Leistungsbeurteilung

### 1/3 - Test (SA)

- 2x Tests pro Semester
  - 1. Test – Do, 17.10.2024 - Themengebiet 7 - 8 (bis Symmetrische Verschlüsselung inkl. Skytale, Cäsar, Vigenere)
  - 2. Test – Do, 12.12.2024 - Themengebiet ??
  - 3. Test – Do, ??.03.2025 - Themengebiet ??
  - 4. Test – Do, ??.05.2025 - Themengebiet ??

### 1/3 - Mitarbeit (MA)

- Aktive Mitarbeit im Unterricht (aMA)
- Mündliche Stundenwiederholungen (mMA)
- Schriftliche Stundenwiederholungen (sMA)

### 1/3 - Praktische Arbeiten (PA)

- 1x praktischer Arbeitsauftrag pro Woche via Google Classroom

## Leistungsstand

Den aktuellen Leistungsstand könnt ihr jederzeit einsehen!

# Lehrplaninhalte

Die nachstehenden Lehrplaninhalte werden in der angegebenen Reihenfolge (=Priorität) durchgenommen werden. Im Unterrichtsfach IT-Labor werden die Themen projektbasiert behandelt.

## 5 + 6. Klasse IT-Labor/WPF

1. Videobearbeitung mit DaVinci Resolve
2. Hardware – PC Systeme (Laptop, Spielekonsole, Handy,..) kennenlernen, Komponenten identifizieren
3. 3D – Modellierung & Animation & Druck
4. Mediendesign (Canva – Flyer, Plakate) & Inkscape Vektorgrafiken (Logo) & Bild (Pixlr) & Podcast bzw. Audiobearbeitung / Videoblog
5. Kamerasysteme (Handy + Gimbal, Drohne, GoPro,..)
6. Netzwerktechnik – Praxis
7. Robotik (Wetterstation,..)
8. Handy-App Programmierung
9. VR – Programmierung
10. Unity – 3D Spielprogrammierung

## 5. Klasse

1. Zahlensysteme
2. Informationseinheiten
3. Zeichencodierung
4. Schaltalgebra
5. Algorithmik und Programmierung Basics (C++ Grundstrukturen bis Funktionen)
6. Tabellenkalkulation & Datenanalyse (Datenimport, filtern, exportieren, SVERWEIS, WENN-DANN, Pivot-Tabellen)

## 6. Klasse

1. Webentwicklung Basics (HTML & CSS)
2. Datenschutz- & Sicherheit & Lizensierung & Kryptographie & Verschlüsselung
3. Netzwerktechnik Theorie & Simulation
4. Algorithmik und Programmierung Datenstrukturen (C++ Rekursionen, Arrays + Sortieralgorithmen)

## 7. Klasse

1. Algorithmik und Programmierung Objektorientierung (Klassen, Vererbung)
2. Betriebssysteme (Windows / Linux – Scripts, Serverdienste – Webserver, DB-Server) & Virtualisierung

3. Datenbanksysteme (ER-Modelle, Relationenmodell, SQL)
4. Textverarbeitung (mehrseitige Dokumente, VWA Vorlagen,..)

# 8. Klasse

1. Webentwicklung Advanced (PHP Formulare & DB-Connection, Javascript, Frameworks Bootstrap, CMS - Systeme)
2. Grundlagen der KI

# 8) Netzwerksicherheit

Zweifel zu haben ist ein unangenehmer Zustand, sich in Sicherheit zu wiegen ist ein absurder Zustand (Voltaire)

Not, Person und Zeit, machen die Gesetze eng und weit.

Es gibt keine Sicherheit, nur mehr oder weniger Unsicherheit (Josef Maler)

Sei vorsichtig, öffne keinem Fremden die Haustür.



## Definition

Netzwerksicherheit steht als Begriff stellvertretend für sämtliche Schutzmaßnahmen, um IT-Infrastrukturen gegen unbefugte Zugriffe, Schäden und Verluste abzusichern. Diese Maßnahmen können technischer oder organisatorischer Natur sein und sorgen dafür, dass ein Netzwerk vertraulich, integer und verfügbar bleibt. So zählen Verschlüsselungstechnologien und Firewalls ebenso zur professionellen Netzwerksicherheit wie Sicherheits- und Passwortrichtlinien und Security-Schulungen.

Um ein IT-Netzwerk umfassend zu sichern, braucht es mehrere Schutzschichten, die jeden Bereich im Netzwerk bedenken. Außerdem ist es sinnvoll, Netzwerk und Sicherheit individuell aufeinander abzustimmen. Wir zeigen Ihnen, welche Security-Schichten notwendig sind und wie umfangreich Ihre IT-Security je nach Unternehmensart und Angriffsrisiken ausfallen sollte.

## Schutz des eigenen IT-Netzwerkes

Ein IT-Netzwerk besteht aus einer Vielzahl an Einzelkomponenten: Wichtige Daten lagern auf internen oder externen Servern, Mitarbeitende sind mit Desktop-PCs, Laptops und Smartphones ausgestattet, ein Gateway oder ein Router sorgt für den Internetzugang und je nach Unternehmensgröße vernetzen ein oder mehrere Switches intern weitere Geräte (LAN) wie beispielsweise Access Points für eine professionelle WLAN-Abdeckung oder auch Arbeitsgeräte oder Drucker.

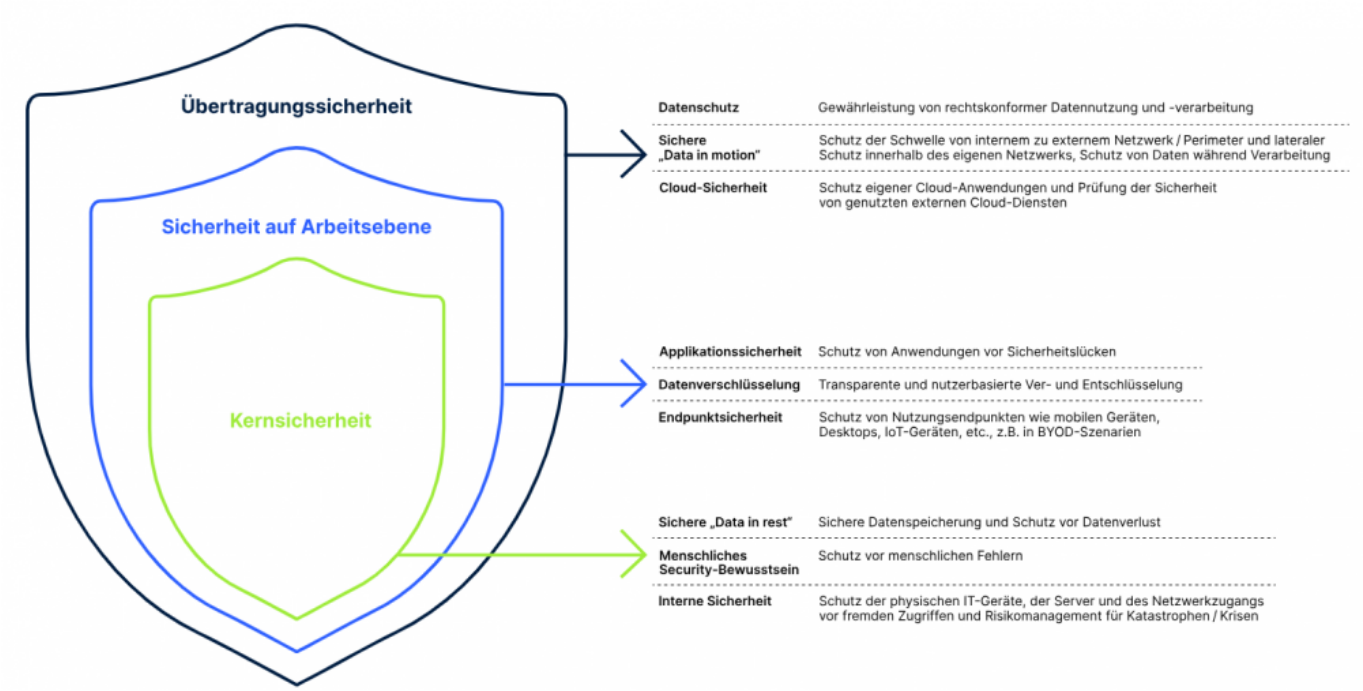
Bei IT-Netzwerken, die lokal weit verteilt und stark verzweigt sind, bedarf es zum Schutz einer mehrschichtigen und skalierbaren IT-Security, die im besten Fall möglichst automatisiert für mehr Sicherheit im Netzwerk sorgt.

## Schichten und Bestandteile professioneller IT-Security

Zum Schutz des eigenen IT-Netzwerkes zählt zum einen die Absicherung sämtlicher sich im Netzwerk befindlichen Daten, Geräte, Server und Applikationen (Datensicherheit, interne Sicherheit und

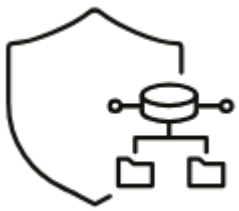
Applikationssicherheit), zum anderen aber auch Perimeter Security, also die Absicherung der Schwelle zwischen internem und externem Netz – demnach auch sichere Übertragungswege und Netzwerkzugänge (Endpunktsicherheit und Datenverschlüsselung).

Zu guter Letzt darf aber auch die Sicherheit der bewegten Daten und externen Cloud-Anwendungen nicht außer Acht gelassen werden (Sicherheit von „data in motion“, Datenschutz und Cloud-Sicherheit), ebenso wenig wie die menschliche Komponente (menschliches Security-Bewusstsein).




## Kernsicherheit

### Sichere „Data in rest“



Sichere Datenspeicherung und Schutz vor Datenverlust durch

- Strenge Passwort-Richtlinien (Komplexität, Ablaufdatum, Zwei-Faktor-Authentifizierung)
- Sicherheitseinstellungen für Dateien
- Getrennte Datenlagerung
- Klar definierte Zugriffs- und Bearbeitungsrechte
- Archive
- Datenbackups

### Interne Sicherheit

Schutz der physischen IT-Geräte, Server und des Netzwerkzugangs vor fremden Zugriffen sowie Risikomanagement für Katastrophen / Krisen durch

- Zugangskontrollen auf Firmengelände
- ggf. Live-Monitoring mit Kameraüberwachung und Aktivitäten-Logs
- Professionelle Next-Generation Web Application UTM-Firewall
- Regelmäßige Security-Patches, Funktionsfähigkeitschecks und Software Updates
- Netzwerksegmentierung

### Menschliches Security-Bewusstsein

Schutz vor menschlichen Fehlern durch

- Umfangreiche IT-Security- und Compliance-Schulungen, insb. zu E-Mail-, Passwort- und Social Media-Sicherheit, Vertraulichkeitsregelungen und Verhalten im Ernstfall
- Tests wie z.B. Phishing-Simulationen
- Vertraulichkeitsklassifikationen für Dateien und Informationen
- Gut zugängliche Sicherheitsrichtlinien
- Regelmäßige Erinnerungen und Auffrischungen

## IT-Sicherheit auf Arbeitsebene

### Endpunktsicherheit

Schutz von Nutzungsendpunkten wie mobilen Geräten, Desktops, IoT-Geräten und -Sensoren, etc., insb. in dezentralen und hybriden Arbeitsumgebungen (BYOD, Remote Work, externe Dienstleister, etc.) durch

- Nutzung von VPN-Clients und ZTNA (Zero-Trust-Network-Access)
- Multifaktor-Authentifizierung von Nutzer:innen
- Übersichtliches, zentral gemanagtes Asset-Inventar aller Geräte und virtuellen Ressourcen
- Individuelle, feingranulare Zugriffsrechte pro Nutzer:in
- Regelmäßige Systemupdates und Securitypatches
- Klare Nutzungs- und Sicherheitsrichtlinien
- Abschalten nicht zwingend notwendiger Ports

**Datenverschlüsselung**

Transparente und nutzerbasierte Ver- und Entschlüsselung im Hintergrund durch

- VPN- oder ZTNA-Netzwerke
- Verschlüsselungsparameter und - algorithmen nach aktuellem BSI-Standard
- Regelmäßige Überprüfung der aktuellen Standards
- Konzept zur Schlüsselverwaltung (PKI - Public Key Infrastructure) inklusive regelmäßiger Audits

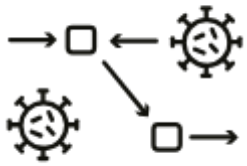
**Applikationssicherheit**

Schutz von Anwendungen vor Sicherheitslücken durch

- Zugriff auf ausschließlich vertrauenswürdige und geprüfte, freigegebene Apps
- Gezielter Einsatz von VPNs
- Zugangskontrolle und Application Monitoring / Steering
- Automatische Sessionterminierung bei Nicht-Nutzung
- Regelmäßige (Security-)Updates
- Aufbewahrungsrichtlinien

## Übertragungssicherheit

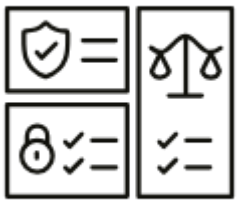
### Sichere „Data in motion“



Schutz der Schwelle von internem zu externem Netzwerk (Perimeter) und lateraler Schutz innerhalb des eigenen Netzwerks durch

- Mehrschichtige Verschlüsselung von Daten während Übertragung
- Ausschließliche Nutzung von VPN- oder ZTNA-Verbindungen
- Netzwerksegmentierung durch VLANs
- Caching Routines zur Vermeidung von öffentlichem Zugang
- Regelmäßige Penetrationstests
- Nutzer- und systemspezifische Zugriffsschlüssel nur auf benötigte Daten

### Datenschutz



Gewährleistung von rechtkonformer Datennutzung und -verarbeitung durch

- DSGVO-konforme Anwendungen und Systeme
- Backdoor-freie Netzwerkkomponenten
- In der EU gehostete Clouddienste

## Cloud-Sicherheit

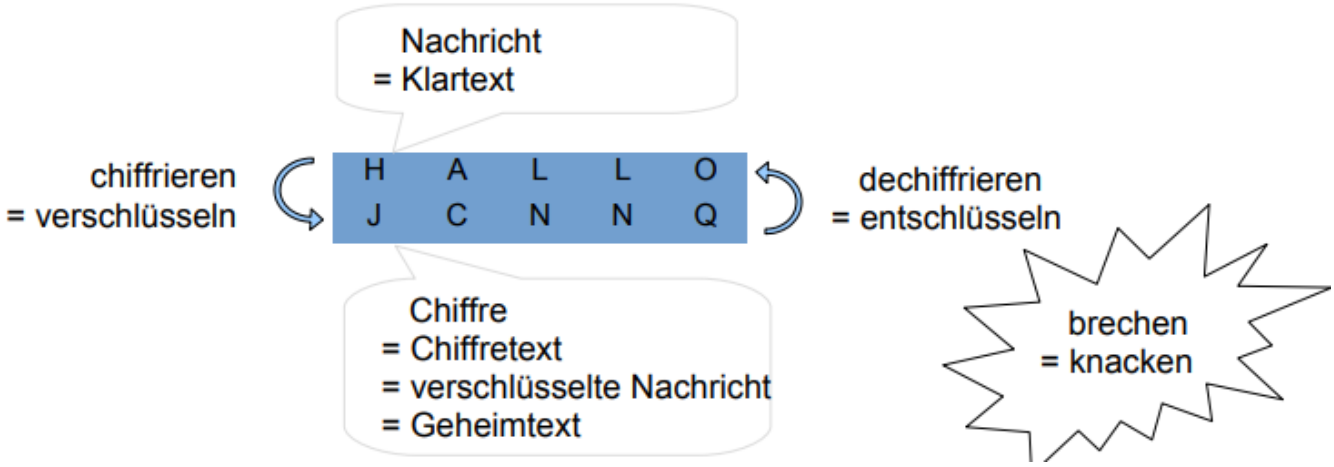
Schutz eigener Cloud-Anwendungen und sorgfältige Prüfung der Sicherheit von genutzten externen Cloud-Diensten durch

- Klare Klärung von Security-Angelegenheiten zwischen Cloud-Anbieter und -Nutzer
- Physische Host-Zugangskontrollen
- Sichere, DSGVO-konforme Infrastruktur
- Georedundanz des Hosts
- Security Patches
- Zugangskontrollen zu Unternehmensdaten in der Cloud
- Backdoor-Freiheit

# Kryptologie

Umgangssprachlich werden kryptologische Begriffe oft nicht eindeutig verwendet. Daher ist inbesonderem Maße auf eine korrekte Verwendung der Begriffe zu achten. Hier eine kurze Zusammenfassung:

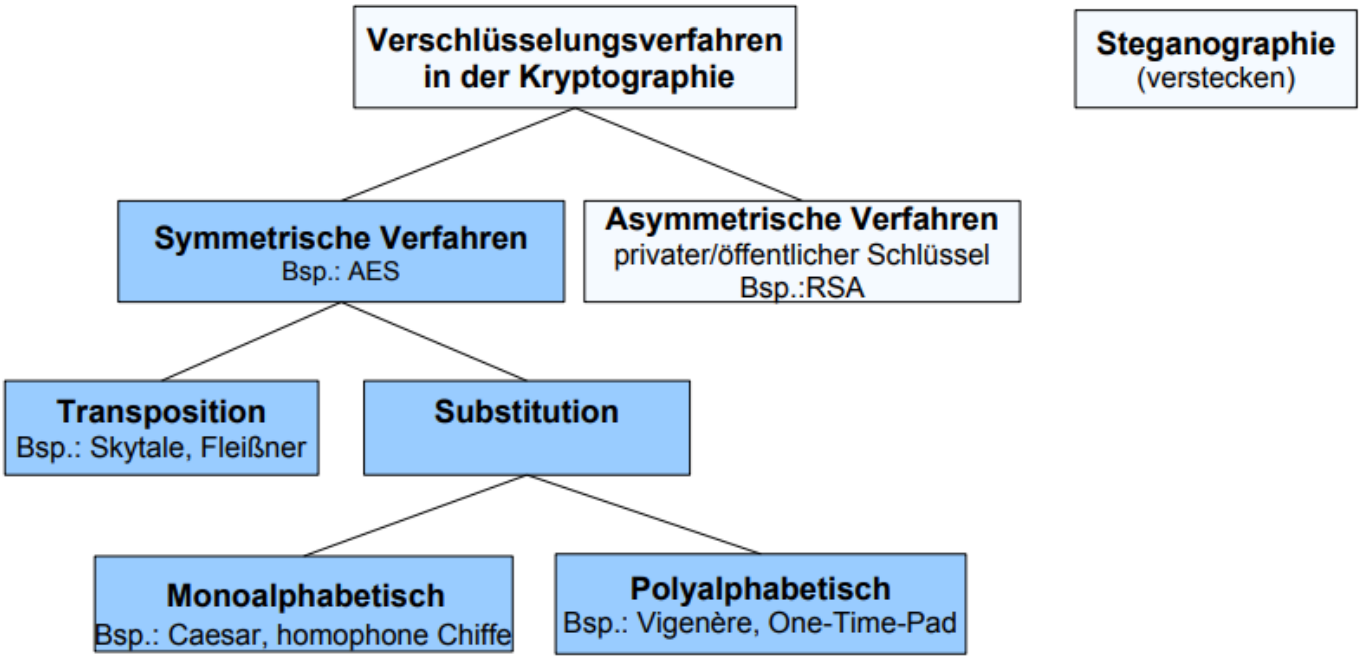
- Beim **Datenschutz** wird die Person mit ihren Rechten geschützt (Persönlichkeitsrecht,Urheberrecht,...).
- Bei der **Datensicherheit** werden die Daten vor unberechtigten Zugriffen geschützt.




Die Kryptologie ist die Wissenschaft der Verschlüsselung und Entschlüsselung von Informationen sowie Analyse kryptografischer Verfahren. Sie umfasst die

- **Kryptografie**: Lehre von den Methoden zur Ver- und Entschlüsselung von Nachrichten zum Zweck der Geheimhaltung von Informationen gegenüber Dritten.
- **Kryptoanalyse**: Analyse und Bewertung der Sicherheit von kryptografischen Verfahren gegen unbefugte Angriffe.

Bei einer **Verschlüsselung** ist das Verfahren (meist) bekannt, der Schlüssel ist geheim. Es geht um den Austausch von Informationen, die nicht für alle bestimmt sind.. Bei einer **Codierung** ist das Verfahren bekannt, und die Anleitung zum Codieren und Decodieren öffentlich. Einen Schlüssel gibt es nicht, und die ausgetauschten Informationen sind nicht geheim. (Blindenschrift, Morsecode, ...).




## Kryptografie

**Kurz:** Kryptografie ist die Lehre der Verschlüsselung von Daten.

**Lang:** Kryptografie ist eine Wissenschaft, die sich mit Methoden beschäftigt, die durch Verschlüsselung und verwandte Verfahren Daten von unbefugten Manipulation schützen sollen.

**Ursprung:** Das Wort Kryptografie kommt aus dem Griechischen, wo kryptein „verstecken“ und gráphein „schreiben“ bedeutet.

Die beiden **wichtigsten Hilfsmittel der Kryptografie** sind:

- Die **Mathematik**, denn nur mit Hilfe von mathematischen Kenntnissen ist es möglich, Verfahren zur sicheren Verschlüsselung von Daten zu entwickeln
- Und der **Computer**, weil er die Verschlüsselungsverfahren ausführt und wichtige Dienste bei der Untersuchung von kryptografischen Methoden auf Schwachstellen leistet.

### Motive der Kryptografie

- **Vertraulichkeit**

Geheimhaltung ist die offensichtlichste und bekannteste Anwendung kryptografischer Verfahren

- **Intigrität**

Für den Empfänger nachprüfbar sein, das er die Nachricht unversehrt erhalten hat

- **Authentizität**

Identität des Absenders einer Nachricht soll für den Empfänger nachprüfbar sein

- **Gültigkeit**

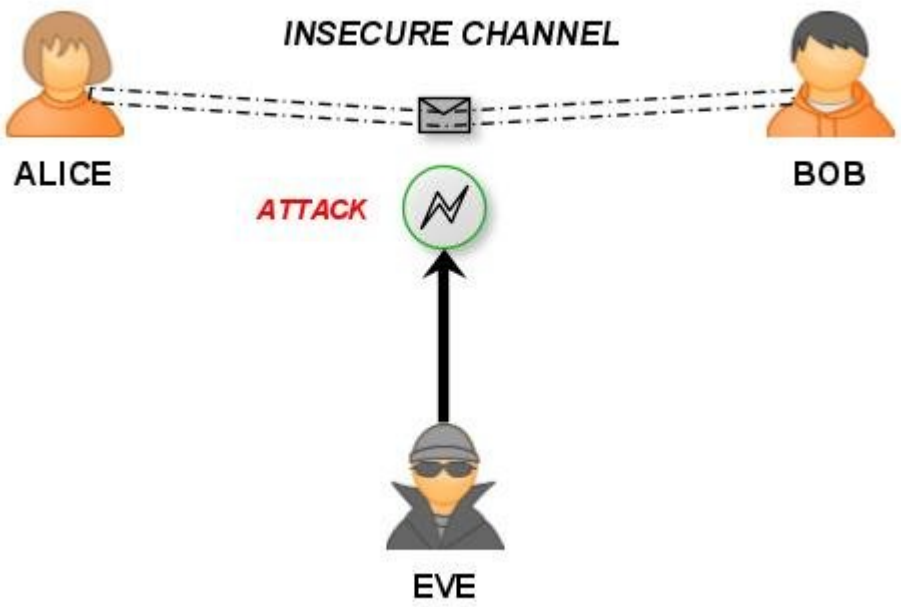
Nachricht kann durch zwischenzeitliche Ereignisse ihre Bedeutung verlieren

- **Nichtabstreitbarkeit**

Dass ist Absender einer Nachricht seine Urheberschaft später nicht verleugnen kann.

### Allgemeines Modell

2 Personen (Alice und Bob) tauschen Daten über einen abhörbaren Kanal aus, heute meist das Internet, es kann dies aber auch eine Telefonleitung, eine Funkverbindung oder der Transport einer Diskette sein. Eine „böse“ gesinnte Person (Mallory/Eve) kann den Übertragungskanal beliebig beeinflussen. Er kann die Daten abfragen, mitlesen, analysieren, manipulieren und weiterleiten.



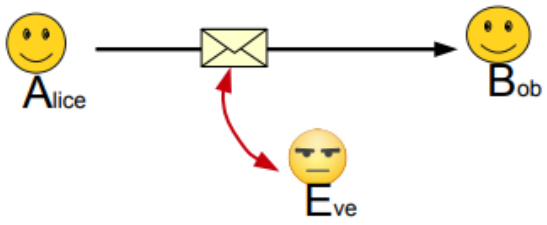

Auf Basis dieses einfachen Modells, kann die Kryptografie durch Verschlüsselung und ähnliche Maßnahmen verhindern dass

- Mallory mit den abgefangenen Daten etwas anfangen kann,
- Mallory übertragene Daten unbemerkt verändert,
- Mallory sich unbemerkt Alice gegenüber als Bob ausgibt (und umgekehrt)
- Alice unerkannt behaupten kann, dass eine von ihr gesendete Nachricht in Wirklichkeit eine Fälschung von Mallory sei.

Die Kryptografie kann aber nicht verhindern, dass

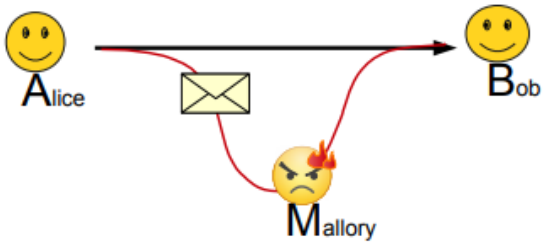
- Mallory Nachrichten verändert (er kann es nur nicht unbemerkt),
- Mallory Daten abfängt (er nur nichts von verschlüsselten Daten),
- Mallory die Leitung zerstört (physikalisch, durch Softwarefehler u.ä.).

| **Szenarien**: | **Gefahren:** | **Ziele** der Kryptologie |
|---|---|---|



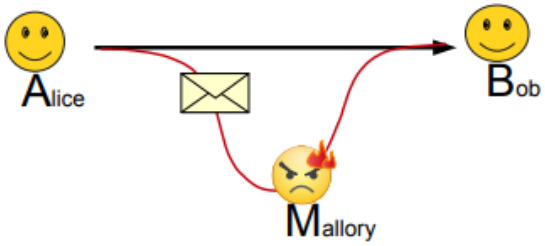

**mitlesen** => **Vertraulichkeit**

Können wirklich nur Alice und Bob die Nachricht lesen?




**ändern** => **Integrität**

Ist die Nachricht unverändert?
Sind die Daten original?



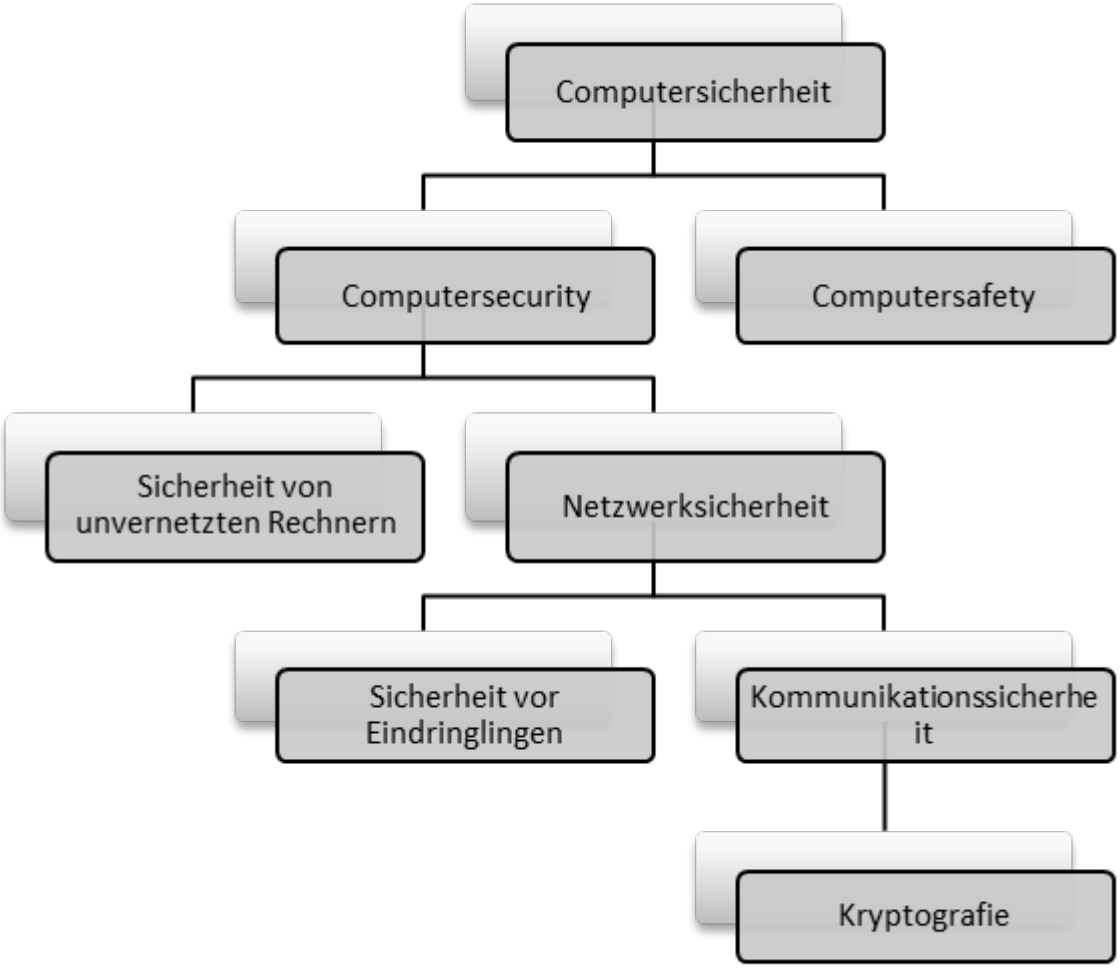

**als A ausgeben**
=> **Authentizität**
Kommt die Nachricht wirklich von Alice?
Landet die Nachricht wirklich bei Bob?

=> **Verbindlichkeit**
Kann Bob beweisen, dass die Nachricht von Alice kommt, selbst wenn sie es abstreitet? ('*Habe ich nie gesagt.*')
Kann Alice beweisen, dass Bob die Nachricht erhalten hat? ('*Habe ich nicht bekommen.*')

=> Unterschiedliche **Ziele** erfordern unterschiedliche **Verfahren**.

Bsp.: Eine Verschlüsselung liefert Vertraulichkeit, aber keine Authentizität.

**Teilgebiet der Computersicherheit**




- In der Computer-Safety geht es um den Schutz vor unbeabsichtigten Schäden. Dazu gehören defekte Geräte, unbeabsichtigtes Löschen, Übertragungsfehler, Festplatten-Crashes, Blitzeinschläge, Überschwemmungen, falsche Bedienung, defekte Speichermedien und Ähnliches.
- Die Computer-Security dagegen bezeichnet die Sicherheit vor absichtlichen Störungen. Dazu gehören die Sabotage von Hardware, Hackereinbrüche, das Schnüffeln in geheimen Dateien und dergleichen.

Der Bereich der Netzwerksicherheit beschäftigt sich vor allem mit zwei Sicherheitsfragen:

- Wie kann ein vernetzter Computer davor beschützt werden, dass ein Unbefugter über das Netzwerk darauf zugreift (man spricht dabei von hacken oder cracken).
- Wie können Nachrichten, die den Computer verlassen vor einem Abhörer oder Manipulierer geschützt werden (Kommunikationssicherheit).

### Gründe für Kryptografie

- **Wirtschaftsspionage**

Staatliche Geheimdienste sind nach dem Ende des kalten Krieges vermehrt auf neue

Beschäftigungsbereiche verlegt worden. Wirtschaftliche Interessen gelten heute als häufiger Grund Spionage zu betreiben. Weltweit führend – nicht nur im Bereich der Wirtschaftsspionage – ist die amerikanische Geheimorganisation NSA (National Security Agancy). Die NSA ist der weltweit größte Arbeitgeber von Mathematikern und größter Hardwareabnehmer. Firmen nutzen heute auch die Kenntnisse von Hackern um die Konkurrenz auszuspionieren. Der geschätzte Schaden durch Wirtschaftsspionage übersteigt in Ländern wie Deutschland die Milliardengrenze. Auch wenn nur ein Teil davon über das Internet erfolgt, so ist die Gefahr vielfach unterschätzt oder nicht bewusst.

- **Kommerzielle Nutzung des Internets**

Ein guter Grund für den Einsatz von Kryptografie ist die Tatsache, dass sich in einem abhör- und manipulationssicheren Internet mittels Online-Shops, Auktionsbörsen, OnlineBanking u. ä. eine große Menge an Geld verdienen/bewegen lässt.

- **Privatsphäre**

Es gibt auch Gründe für den Einsatz von Kryptografie, die keine kommerziellen Gedanken verfolgen. So ist es das Recht jedes Bürgers, eine Privatsphäre zu behalten durch den Einsatz von Kryptografie möglich. Ein privater Brief wird ja auch in einem Umschlag versandt! Es ist im Internet auf jeden Fall einfacher, abgefangene Daten maschinell auszuwerten, als dies mit herkömmlicher Briefpost der Fall war.

Bei allen Vorteilen der Kryptografie darf aber nicht vergessen werden, dass sie auch Gefahren bringt: Kriminelle können durch den Einsatz geeigneter Verschlüsselungsverfahren nach Belieben Nachrichten austauschen.

### Anwendungen

- **Passwörter**

Kryptografie wird häufig eingesetzt, um die Authentizität von Passwörtern zu überprüfen und gleichzeitig gespeicherte Passwörter zu verschleiern. Auf diese Weise können Dienstanbieter Passwörter authentifizieren, ohne eine Klartextdatenbank mit allen Passwörtern führen zu müssen, die für Hacker anfällig sein könnte.


* **Sicheres Surfen im Internet** Beim Surfen auf sicheren Websites schützt die Kryptografie die Benutzer vor Lauschangriffen und „Man-in-the-Middle“-Angriffen (MitM). Die Protokolle Secure Sockets Layer (SSL) und Transport Layer Security (TLS) basieren auf der Verschlüsselung mit öffentlichen Schlüsseln, um die zwischen Webserver und Client gesendeten Daten zu schützen und sichere Kommunikationskanäle herzustellen.


- **Elektronische Signaturen**

Elektronische Signaturen, oder E-Signaturen, werden zum Unterzeichnen wichtiger Dokumente im Internet verwendet und gelten oftmals als rechtsverbindlich. Mit Kryptografie erstellte elektronische Signaturen können validiert werden, um Betrug und Fälschungen zu verhindern.


- **Kryptowährungen**

Kryptowährungen wie Bitcoin und Ethereum beruhen auf einer komplexen Verschlüsselung von Daten, deren Entschlüsselung erhebliche Mengen an Rechenleistung erfordert. Durch diese Entschlüsselungsprozesse erfolgt das sogenannte „Minting“ neuer Coins, die dann in Umlauf gebracht werden. Kryptowährungen stützen sich zudem auf fortschrittliche Kryptografie, um Krypto-Wallets zu sichern, Transaktionen zu verifizieren und Betrug zu verhindern.

- **Authentifizierung**

In Situationen, in denen eine Identitätsauthentifizierung erforderlich ist, wie z. B. bei der Anmeldung bei einem Online-Bankkonto oder beim Zugriff auf ein sicheres Netzwerk, kann die Kryptografie bei der Verifizierung der Identität von Benutzern und der Authentifizierung ihrer Zugriffsberechtigungen helfen.


- **Sichere Kommunikation**

Ganz gleich, ob es um den Austausch von Staatsgeheimnissen oder um eine private Unterhaltung geht – die End-to-End-Verschlüsselung wird zur Authentifizierung von Nachrichten und zum Schutz von Zwei-Wege-Kommunikation wie Videokonferenzen, Sofortnachrichten und E-Mails verwendet. Die End-to-End-Verschlüsselung bietet ein hohes Maß an Sicherheit und Privatsphäre für die Nutzer und wird häufig in Kommunikations-Apps wie WhatsApp und Signal verwendet.

# Steganografie

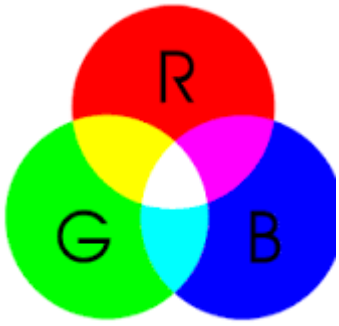
Ein steganografisches Verfahren verheimlicht, dass überhaupt geheime Daten existieren. Der Gedanke dahinter: Wo niemand geheimen Daten vermutet, wird sie auch niemand suchen. Steganografie-Software versteckt die geheimen Daten in einer anderen Datei. Also so genannte Trägerdateien dienen in der Regel meist Bilder, Sound-, Text- und Video-Dateien. Dieses Verfahren kann man nicht nur zum Schutz von Daten benutzen, sondern es wird auch zur Kenntlichmachung von Urheberrechten verwendet. Wer von der Verschlüsselung nichts weiß, nutzt die betreffende Trägerdatei ohne Einschränkungen mit der passenden Anwendung. Nur wer über die Verschlüsselung informiert ist und zudem Zugriff auf den verwendeten Kodierungs-Schlüssel hat, kann die in der Trägerdatei enthaltenen Informationen entschlüsseln und für sich nutzbar machen.

„Vertraue keinem Verschlüsselungsverfahren, das du nicht selbst geknackt hast.“

Beispiel:

Nehmen wir folgendes Bild als Grundlage:


Nun nehmen wir pro Pixel den Hexadezimalen Farbcode und rechnen diesen in das Binäre Zahlensystem um. Dann ergeben sich folgende Werte für die ersten acht Pixel im Bild.




Ein hexadezimaler Farbcode hat 6 Ziffern (z.B.: #ed1c24). Immer zwei Ziffern bilden eine Farbe ab (Rot, Grün, Blau). Eine Hex-Ziffer kann 16 mögliche Zeichen annehmen. Sprich Ein Farbcode (z.B.: Rot) hat nun 16*16 = 256 Möglichkeiten (0-255). Für 256 Möglichkeiten benötigen wir insgesamt 8 Bits. Nachdem wir bei RGB drei Farben darstellen und beliebig kombinieren können, benötigen wir 3×8 = 24Bits. Somit kann man 3^24 = 16777216 = ca. 16,8 Mio. verschiedene Farben mit dem RGB-Modell darstellen.

```
1.Pixel: 111011010001110000100100
2.Pixel: 111011010001110000100100
3.Pixel: 111011010001110000100100
4.Pixel: 111011010001110000100100
5.Pixel: 111011010001110000100100
6.Pixel: 111011010001110000100100
7.Pixel: 111011010001110000100100
8.Pixel: 111011010001110000100100
```

Jetzt nehmen wir beispielsweise einen Text, welcher lautet „Das hier ist ein geheiner Text.“ und rechnen auch hier erstmal das erste Zeichen des Textes in das Binär-System um. Es ergeben sich hierbei folgende Werte für, in dem Fall, den Buchstaben „D“.

ASCII Table

| Dec | Hex | Oct | Char | Dec | Hex | Oct | Char | Dec | Hex | Oct | Char | Dec | Hex | Oct | Char |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | | 32 | 20 | 40 | [space] | 64 | 40 | 100 | @ | 96 | 60 | 140 | ` |
| 1 | 1 | 1 | | 33 | 21 | 41 | ! | 65 | 41 | 101 | A | 97 | 61 | 141 | a |
| 2 | 2 | 2 | | 34 | 22 | 42 | " | 66 | 42 | 102 | B | 98 | 62 | 142 | b |
| 3 | 3 | 3 | | 35 | 23 | 43 | # | 67 | 43 | 103 | C | 99 | 63 | 143 | c |
| 4 | 4 | 4 | | 36 | 24 | 44 | $ | 68 | 44 | 104 | D | 100 | 64 | 144 | d |
| 5 | 5 | 5 | | 37 | 25 | 45 | % | 69 | 45 | 105 | E | 101 | 65 | 145 | e |
| 6 | 6 | 6 | | 38 | 26 | 46 | & | 70 | 46 | 106 | F | 102 | 66 | 146 | f |
| 7 | 7 | 7 | | 39 | 27 | 47 | ' | 71 | 47 | 107 | G | 103 | 67 | 147 | g |
| 8 | 8 | 10 | | 40 | 28 | 50 | ( | 72 | 48 | 110 | H | 104 | 68 | 150 | h |
| 9 | 9 | 11 | | 41 | 29 | 51 | ) | 73 | 49 | 111 | I | 105 | 69 | 151 | i |
| 10 | A | 12 | | 42 | 2A | 52 | * | 74 | 4A | 112 | J | 106 | 6A | 152 | j |
| 11 | B | 13 | | 43 | 2B | 53 | + | 75 | 4B | 113 | K | 107 | 6B | 153 | k |
| 12 | C | 14 | | 44 | 2C | 54 | , | 76 | 4C | 114 | L | 108 | 6C | 154 | l |
| 13 | D | 15 | | 45 | 2D | 55 | - | 77 | 4D | 115 | M | 109 | 6D | 155 | m |
| 14 | E | 16 | | 46 | 2E | 56 | . | 78 | 4E | 116 | N | 110 | 6E | 156 | n |
| 15 | F | 17 | | 47 | 2F | 57 | / | 79 | 4F | 117 | O | 111 | 6F | 157 | o |
| 16 | 10 | 20 | | 48 | 30 | 60 | 0 | 80 | 50 | 120 | P | 112 | 70 | 160 | p |
| 17 | 11 | 21 | | 49 | 31 | 61 | 1 | 81 | 51 | 121 | Q | 113 | 71 | 161 | q |
| 18 | 12 | 22 | | 50 | 32 | 62 | 2 | 82 | 52 | 122 | R | 114 | 72 | 162 | r |
| 19 | 13 | 23 | | 51 | 33 | 63 | 3 | 83 | 53 | 123 | S | 115 | 73 | 163 | s |
| 20 | 14 | 24 | | 52 | 34 | 64 | 4 | 84 | 54 | 124 | T | 116 | 74 | 164 | t |
| 21 | 15 | 25 | | 53 | 35 | 65 | 5 | 85 | 55 | 125 | U | 117 | 75 | 165 | u |
| 22 | 16 | 26 | | 54 | 36 | 66 | 6 | 86 | 56 | 126 | V | 118 | 76 | 166 | v |
| 23 | 17 | 27 | | 55 | 37 | 67 | 7 | 87 | 57 | 127 | W | 119 | 77 | 167 | w |
| 24 | 18 | 30 | | 56 | 38 | 70 | 8 | 88 | 58 | 130 | X | 120 | 78 | 170 | x |
| 25 | 19 | 31 | | 57 | 39 | 71 | 9 | 89 | 59 | 131 | Y | 121 | 79 | 171 | y |
| 26 | 1A | 32 | | 58 | 3A | 72 | : | 90 | 5A | 132 | Z | 122 | 7A | 172 | z |
| 27 | 1B | 33 | | 59 | 3B | 73 | ; | 91 | 5B | 133 | [ | 123 | 7B | 173 | { |
| 28 | 1C | 34 | | 60 | 3C | 74 | < | 92 | 5C | 134 | \ | 124 | 7C | 174 | \| |
| 29 | 1D | 35 | | 61 | 3D | 75 | = | 93 | 5D | 135 | ] | 125 | 7D | 175 | } |
| 30 | 1E | 36 | | 62 | 3E | 76 | > | 94 | 5E | 136 | ^ | 126 | 7E | 176 | ~ |
| 31 | 1F | 37 | | 63 | 3F | 77 | ? | 95 | 5F | 137 | _ | 127 | 7F | 177 | |

Laut der ASCII-Tabelle hat der Buchstabe D den dezimalen Wert 68. Binär ergibt sich somit folgende Ziffernfolge:

```
01000100
```

Jetzt wirds interessant. Wir nehmen jetzt quasi pro Pixel im Bild die schwächste Bit (die letzte) und ändern es so um, wie wir es brauchen. Das heißt, bei dem ersten Pixel, nehmen wir, in dem Fall die Null und schauen, ob das mit dem ersten Bit des Buchstaben „D“ übereinstimmt.

Danach kommt der nächste Pixel. Auch hier nehmen wir das die schwächste Bit (auch wieder eine Null) und schauen ob dieses mit dem zweiten Bit des Buchstaben „D“ übereinstimmt. Da die zweite Bit des Buchstaben „D“ allerdings eine 1 ist ändern wir die letzte Bit des zweiten Pixels auf eine 1 ab. Somit verändert sich die Farbe fast garnicht. Hier ein vergleich der Farbe (vorher/nachher):

**Vorher**:

```
111011010001110000100100
```

**Nachher**:

```
111011010001110000100101
```

Dieses vorgehen wenden wir nun an jedem Pixel des Bildes vor, bis der ganze Satz im Bild versteckt ist. In folgendem Beispiel ist im Bild der folgende Text versteckt:

```
Steganographie ist im Grunde genommen eine Technik um Daten jeglicher Art zu
verstecken. Das Wort Steganographie stammt vom griechichen Wort „steganos“
ab, was so viel wie Verbergen heißt. Es gibt verschiedene Arten der
Steganographie. Die am verbreitetste Art der Steganographie ist mittlerweile
die technische Steganographie. Hier werden meistens bestimmte Daten
innerhalb eines Bildes versteckt.
Eine spezifische Art dieser Kunst/Technik wurde auch beim weltbekannten
Internet-Rätsel „Cicada 3301“ verwendet.
Im Folgenden werde ich versuchen eine bestimmte Art der Steganographie zu
erklären.
```

Originalbild:


Bild mit Steganografie:

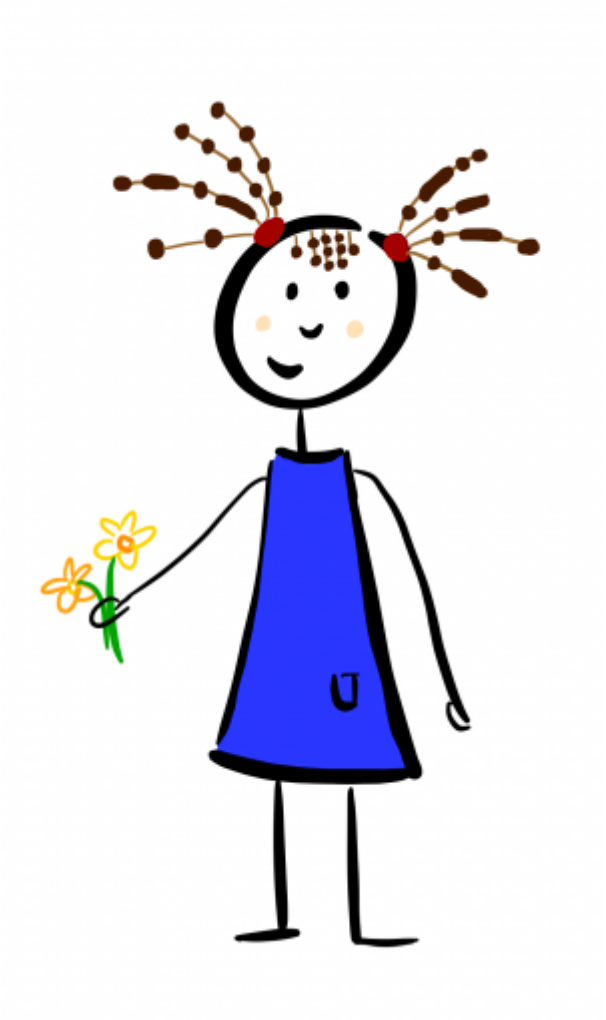

Wie man Sieht kann so eine Nachricht oder eine Datei vollkommen unerkannt übertragen werden. Diese Technik kann so weit geführt werden, dass ganze Bilder in anderen Bildern versteckt werden.

Je mehr Informationen in dem Grundbild sind desto Mehr Daten können darin versteckt werden.

### Weitere Beispiele

In dem Bild des Mädchens wurde ihr Name verborgen. Die Nachricht ist mit einem bekannten Verfahren codiert.

*Tipp: Um die Nachricht zu lesen, benötigst du den „Morse-Code“, bei dem jeder Buchstabe durch eine Kombination aus Punkten und Strichen ersetzt wird!*



Aufgabe 1) Finde die verbotene Nachricht!

Aufgabe 2) Erstelle selbst ein Bild mit einer geheimen Nachricht!

Mehr zu Steganographie

# Kryptografie

Die Kryptografie basiert auf mathematischen Verfahren. Die Sicherheit eines Kryptosystems lässt sich also mathematisch beweisen und berechnen. Die mathematische Beweisführung einer gewissen Sicherheit beruht jedoch oft nur auf Annahmen. Zum Beispiel: „Solange diese Bedingung erfüllt ist, ist dieses Verschlüsselungsverfahren sicher.“ Das hat Konsequenzen. Denn ein ungeschickt implementiertes Kryptosystem kann ein eigentlich sicheres Verschlüsselungsverfahren unsicher machen.

Wie sicher ein kryptografisches Verfahren ist, ist zu allen Zeiten immer zu optimistisch gewesen. Prinzipiell neigen wir zur Selbstüberschätzung, was die Sicherheit einer Technik angeht. Dabei zeigt die Erfahrung, dass kein Aufwand zu groß ist, um ein Verfahren zu brechen. Die Fragestellung ist nur, ob sich der Aufwand, in Erwartung des Inhalts verschlüsselter Daten, lohnt.

## Kerckhoffs Prinzip

Die Sicherheit des Verschlüsselungsverfahrens beruht nur auf der Geheimhaltung des Schlüssels und nicht auf der Geheimhaltung des Verschlüsselungsverfahrens! Die Sicherheit eines Systems sollte nie allein von der Geheimhaltung der Funktionsweise abhängig sein (sonst: Security by Obscurity).

# Symmetrische Kryptografie/Verschlüsselung

Die Verschlüsselungsverfahren, die mit einem geheimen Schlüssel arbeiten, der zum Ver- und Entschlüsseln dient, nennt man symmetrische Verfahren oder Secret-Key-Verfahren. Üblich sind auch die Bezeichnungen Secret-Key-Kryptografie und Secret-Key-Verschlüsselung. Fast alle symmetrischen Verfahren sind auf ressourcenschonende Umgebungen optimiert. Sie zeichnen sich durch geringe Hardwareanforderungen, geringen Energieverbrauch und einfache Implementierung in Hardware aus.

## Prinzip


Die Verschlüsselungsverfahren der symmetrischen Kryptografie arbeiten **mit einem einzigen Schlüssel**, der bei der Ver- und Entschlüsselung vorhanden sein muss. Diese **Verfahren sind schnell** und bei entsprechend **langen Schlüsseln bieten sie auch eine hohe Sicherheit**.

Der **Knackpunkt liegt in der Schlüsselübergabe** zwischen den Kommunikationspartnern. Vor der sicheren Datenübertragung mit Verschlüsselung müssen sich die Kommunikationspartner auf den Schlüssel einigen und austauschen. Wenn der Schlüssel den selben Kommunikationspfad nimmt, wie die anschließend verschlüsselten Daten, dann besteht die Gefahr, dass ein Angreifer in Besitz des Schlüssels gelangt, wenn er die Kommunikation abhört. Wenn der Angreifer den Schlüssel hat, dann kann er nicht nur die Daten entschlüsseln, sondern auch selber Daten verschlüsseln, ohne das es die Kommunikationspartner bemerken. Knackpunkt ist der unsichere Schlüsselaustausch und die Authentifizierung der Kommunikationspartner.

Sicher ist die Schlüsselübergabe nur dann, wenn sich zwei Personen persönlich treffen und den Schlüssel austauschen oder der Schlüssel einen anderen Weg nimmt (Seitenkanal), wie es die Daten tun. Ein Möglichkeit wäre der postalische Weg (Brief, Einschreiben mit Rückschein). Allerdings nicht per E-Mail (Postkarten-Effekt). Zur Unsicherheit trägt außerdem bei, wenn einer der Kommunikationspartner den Schlüssel nur ungenügend sicher aufbewahrt.

Der sichere Schlüsselaustausch ist eines der vielen Probleme der Kryptografie. Mit der asymmetrischen Kryptografie versucht man dieses Problem zu lösen. Weil die asymmetrische Kryptografie weit komplexere Verfahren umfasst, kombinieren die übliche kryptografischen Protokolle sowohl symmetrische als auch asymmetrische Verfahren.

## Vorteile

- Gleicher Schlüssel zum Verschlüsseln und Entschlüsseln
- Je zwei Teilnehmer benötigen einen Schlüssel
- Beide müssen den Schlüssel stets geheim halten
- Anzahl der Schlüssel wächst quadratisch mit der Teilnehmerzahl

# Nachteile

- Sichere Verteilung des Schlüssels (Telefon, schriftlich,...)
- Nicht geeignet für Digitale Signatur

# Symmetrische Verschlüsselungsverfahren

Jede symmetrische Verschlüsselung basiert auf einem bestimmten Algorithmus. Bei einem Verschlüsselungsalgorithmus bzw. Chiffre wird in den Klartext eine Geheiminformation, den Schlüssel, eingebracht und so der Geheimtext gebildet. Der Schlüssel kann ein Passwort, eine geheime Nummer oder auch nur eine zufällige Bitfolge sein.

## Monoalphabetische Substitutionschiffren

Die einfachste Art der Verschlüsselung erreicht man, in dem man jeden Buchstaben ein festes Symbol zuordnen. Diese Verfahren sind monoalphabetisch. Sie sind bei genügend Verschlüsselungsmaterial leicht durch eine Häufigkeitsanalyse zu brechen. In jeder Schriftsprache kommen bestimmte Buchstaben häufiger vor. Man kann also mit einfachen statistischen Mitteln eine Kryptoanalyse machen. Mit Computer-Unterstützung geht es automatisch und noch schneller.

- 8.1.2.1.1) Cäsar-Chiffre

## Polyalphabetische Substitutionschiffren

Wesentlich schwieriger sind polyalphabetische Geheimtexte. Hier kann ein Buchstabe mehreren Symbole entsprechen. Statistische Verfahren funktionieren hier nicht mehr so einfach.

- 8.1.2.1.2) Vigenere-Chiffre
- 8.1.2.1.3) One Time Pad (Vernam-Chiffre)

## Permutationschiffren

Eine Umordnung, eine Permutation einer gegebenen Zeichenfolge, nennt man Permutations- oder Transpositionschiffre. Dies trifft in diesem Fall auf die Skytale zu. Permutationschiffren werden auch als Transposition bezeichnet. Die Skytale ist ein Spezialfall der Transposition. Denkbar wäre nämlich eine Permutationschiffre, die zur Erstellung des Geheimtextes erst den ersten, dann den 47-ten, danach den 32-ten Buchstaben nimmt, usw. Bei der Skytale wird jedoch, wie oben als Matrix betrachtet, die Nachricht zeilenweise aufgetragen und chiffriert liegt diese spaltenweise vor. Die Skytale ist also letztendlich eine einfache Matrixtransposition.

Bei einer Permutations-Chiffre werden somit die Buchstaben den Klartext nicht ersetzt sondern durcheinander gewürfelt. Fast man zB immer 5 Buchstaben des Klartextes zusammen und lässt sich durch die Permutations-Chiffre mit dem Schlüssel (4,1,2,5,3), dann erhält man zB folgenden Chiffretext:

```
IE SBGE TZIW EARNT LVU ENNUTS: EHOLEC, ZDI UE EENB DGRIENS; NWS ANI
EFAEANNG.

ES GIBT ZWEI ARTEN VON LEUTEN: SOLCEH DIE ZU ENDE BRINGEN, WAS SIE ANFANGEN.
```

Die Art der Verschlüsselung lässt sich durch Probieren – abhängig von der Schlüssellänge – mehr oder weniger schnell knacken. Auch die Häufigkeitsanalyse liefert wieder Rückschlüsse über die verwendete Sprache etc.

- 8.1.2.1.4) Skytale

## Operationen

Alle gängigen symmetrische Verfahren arbeiten ausschließlich mit Bit-weisen Operationen. Hier werden Schlüssel, Klartext und Geheimtext in Form von Bitfolgen verarbeitet. In dem die Funktionen nahezu beliebig miteinander kombiniert werden, lassen sich neu symmetrische Verfahren in nahezu beliebiger Zahl entwickeln und mit bekannten Angriffen auf Schwächen testen. In der Regel kombinieren symmetrische Verschlüsselungsalgorithmen Substitutionschiffren und Permutationschiffren miteinander und wiederholen den Vorgang mehrmals (Runden), wobei eine härtere Verschlüsselung entsteht. Typische Bestandteile von symmetrischen Verschlüsselungsalgorithmen sind:

- Exklusiv-oder-Verknüpfung
- Permutation: Reihenfolge einer Bit-Folge wird verändert.
- Substitution: Eine Bit-Folge wird durch eine andere ersetzt.

Erfahrungsgemäß sind für eine wirkungsvolle Verschlüsselung keine aufwendigen Funktionen notwendig. Insbesondere beim Hardware-nahen Programmieren oder der Implementierung in Hardware ist das von Vorteil, weil sich so eine hohe Geschwindigkeit erreichen lässt. Beim praktischen Einsatz von Verschlüsselungsalgorithmen stellt sich auch immer die Frage, wie groß die Rechenleistung für die Verschlüsselung ist. Generell gilt, je schneller ein Verschlüsselungsverfahren arbeitet, desto niedriger sind die Hardwarekosten.

## Moderne(re) Verschlüsselungsverfahren

Bei den symmetrischen Verschlüsselungsverfahren gilt der AES als Maß der Dinge. Es gibt aber auch weitere...

- 8.1.2.1.5) DES
- 8.1.2.1.6) AES
- 3DES - Triple DES
- IDEA - International Data Encryption Algorithm
- RC4 (Rivest-Cipher 4)

- Blowfish (von Bruce Schneier)
- RC5, RC5a, RC6 (Rivest-Cipher 5 bzw. 5a bzw. 6)
- A5 (GSM)
- Serpent
- Twofish (von Bruce Schneier)
- MARS
- SAFER/SAFER+
- CAST (Carlisle Adams und Stafford Tavares)
- MAGENTA
- MISTY1
- Camellia
- Ascon

## Quellen

- Elektronik Kompendium
- Kryptografie.de
- Cryptool
- Wikipedia

# Cäsar Chiffre

Die Cäsar-Chiffre ist eines der einfachsten, aber auch unsichersten Verfahren, um Texte zu verschlüsseln. Das Verfahren wurde nach dem römischen Kaiser Julius Cäsar benannt, der auf diese Weise bereits vor über 2000 Jahren Nachrichten verschlüsselt haben soll.

Die Cäsar-Chiffre ist eine monoalphabetische Substitution, das heißt, jeder Buchstabe des Textes wird durch genau einen anderen Buchstaben des Alphabets ersetzt. Dieser Austausch geschieht jedoch nicht zufällig, sondern basiert auf zyklischer Rotation des Alphabets um *k* Zeichen, wobei *k* der verwendete Schlüssel ist.

Als eines der einfachsten und unsichersten Verfahren dient es heute hauptsächlich dazu, Grundprinzipien der Kryptologie anschaulich darzustellen. Der Einfachheit halber werden oftmals nur die 26 Buchstaben des lateinischen Alphabets ohne Unterscheidung von Groß- und Kleinbuchstaben als Alphabet für Klartext und Geheimtext verwendet und Sonderzeichen, Satzzeichen usw. nicht beachtet.

## Verschlüsselung

Die Verschlüsselung einer Nachricht erfolgt buchstabenweise mit einem Schlüssel *k* aus der Menge $Z_{26}$ = *{0,1,2,3...25}*, wobei der Wert *k = 0* nicht sinnvoll ist, da der Originaltext in diesem Fall keine Änderung erfährt.

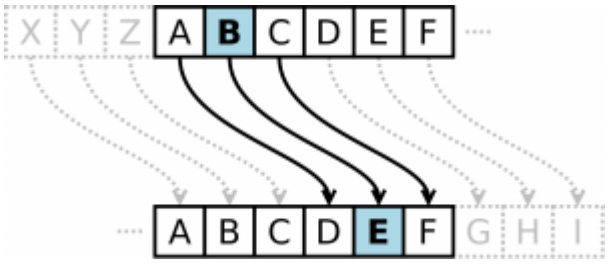
Für einen gegebenen Buchstaben wird zunächst anhand der folgenden Tabelle seine Position *m* im Alphabet bestimmt.

| A | B | C | D | E | F | G | H | I | J | K | L | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| | | | | | | | | | | | | |

Anschließend erhält man den Wert *c* des verschlüsselten Buchstaben durch folgende kurze Berechnungsformel:

```
c = (m+k) mod 26
```

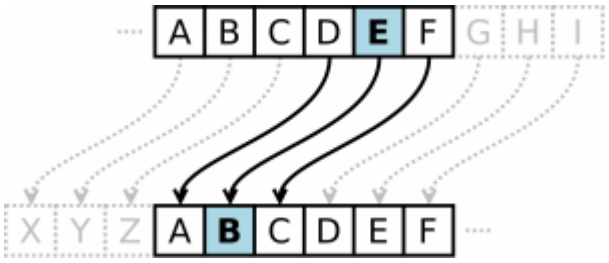
Mit Hilfe obiger Tabelle kann dieser Wert *c* wieder in einen Buchstaben transformiert werden.




## Entschlüsselung

Die Entschlüsselung einer Nachricht erfolgt ähnlich wie die Verschlüsselung mit Schlüssel , wir verwenden jedoch die Formel:

```
m = (26+c-k) mod 26
```




## Beispiel

Der Satz „OTTO KOMMT“ wird mit dem Schlüssel *k=3* verschlüsselt.

| unverschlüsselt | O | T | T | O | | K | O | M | M | T |
|---|---|---|---|---|---|---|---|---|---|---|
| $m$ | 14 | 19 | 19 | 14 | | 10 | 14 | 12 | 12 | 19 |
| $c \equiv (m + k) \bmod 26$ | 17 | 22 | 22 | 17 | | 13 | 17 | 15 | 15 | 22 |
| verschlüsselt | R | W | W | R | | N | R | P | P | W |
| | | | | | | | | | | |

Statt nur über dem Alphabet $Z_{26}$ kann man analog allgemeine Cäsar-Chiffre über beliebigen endlichen Alphabeten $Z_{26}$ = {0,1,..a-1} definieren.

Klartext: H A L L O

Geheimtext: K D O O R

Video

# Vignere Chiffre

Die Vigenère-Chiffre (auch: Vigenère-Verschlüsselung) ist eine aus dem 16. Jahrhundert stammende Handschlüsselmethode zur Verschlüsselung von geheim zu haltenden Textnachrichten.

Es handelt sich um ein monographisches polyalphabetisches Substitutionsverfahren. Der Klartext wird in Monogramme (Einzelzeichen) zerlegt und diese durch Geheimtextzeichen substituiert (ersetzt), die mithilfe eines Kennworts aus mehreren (poly) unterschiedlichen Alphabeten des „Vigenère-Quadrats“ ausgewählt werden. Dabei handelt es sich um eine quadratische Anordnung von untereinander stehenden verschobenen Alphabeten (siehe Bild).

Die Vigenère-Chiffre steht im Gegensatz zu den einfacheren monoalphabetischen Substitutionsmethoden, bei denen nur ein einziges (mono) Alphabet verwendet wird. Aufgrund ihrer für die damalige Zeit als besonders hoch eingeschätzten kryptographischen Sicherheit wurde sie auch als le chiffre indéchiffrable (frz. für „die unentzifferbare Chiffre“) bezeichnet, eine aus damaliger Sicht vielleicht zutreffende, aber aus heutiger Sicht falsche Beurteilung.

## Methode

Ausgehend vom Standardalphabet mit seinen 26 Großbuchstaben werden alle möglichen Caesar-verschobenen Alphabete daruntergeschrieben. Man erhält eine quadratische Anordnung von 26 × 26 Buchstaben, ursprünglich als Tabula recta, später auch als carré de Vigenère (frz. für „Vigenère-Quadrat“) bezeichnet. In der folgenden Darstellung sind der Deutlichkeit halber oberhalb des eigentlichen Quadrats eine Zeile mit den Klartextbuchstaben und links eine Spalte mit den Schlüsselbuchstaben ergänzt worden, die prinzipiell nicht benötigt werden.

**Vigenère-Quadrat**

| Schlüssel | Klartext: A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Geheimtext |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | |
| B | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | |
| C | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | |
| D | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | |
| E | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | |
| F | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | |
| G | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | |
| H | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | |
| I | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | |
| J | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | |
| K | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | |
| L | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | |
| M | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | |
| N | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | |
| O | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | |
| P | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | |
| Q | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | |
| R | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | |
| S | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | |
| T | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | |
| U | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | |
| V | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | |
| W | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | |
| X | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | |
| Y | Y | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | |
| Z | Z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | |

Zur Verschlüsselung eines Klartextes wie beispielsweise des Satzes „Werde Mitglied bei Wikipedia“ benötigt der Verschlüssler zunächst einen Schlüssel. Idealerweise sollte dieser möglichst lang sein und aus einer möglichst „zufälligen“ Buchstabenfolge bestehen. Erreicht die Länge des Schlüssels die des Klartextes und wird der Schlüssel nicht mehrfach verwendet, dann erhält man ein tatsächlich „unknackbares“ Verfahren, wie es aber erst Jahrhunderte später, im Jahr 1882, vom amerikanischen Kryptologen Frank Miller (1842–1925) vorgeschlagen wurde, und das heute als One-Time-Pad (Abkürzung: OTP, deutsch: „Einmalschlüssel-Verfahren“) bezeichnet wird. Zur Zeit von Vigenère und noch bis ins 20. Jahrhundert hinein wurden allerdings regelmäßig relativ kurze und häufig auch leicht

zu erratende Schlüssel benutzt, die zudem mehrfach verwendet wurden. Ein Beispiel wäre die Verwendung von WILLKOMMEN als Schlüsselwort.

Als praktisches Hilfsmittel kann der Verschlüssler den zu verschlüsselnden Text in eine Zeile schreiben und darüber das Kennwort so oft wiederholen, wie es nötig ist:

```
WILLKOMMEN WILLKOMMEN WILLK
WerdeMitgl iedbeiWiki pedia
```

Die entsprechenden Geheimtextbuchstaben kann er nun leicht mithilfe des Vigenère-Quadrats ermitteln. Dazu sucht er den Kreuzungspunkt der durch den jeweiligen Schlüsselbuchstaben gekennzeichneten Zeile und der Spalte des Quadrats, die oben durch den Klartextbuchstaben gekennzeichnet ist. Beispielsweise zur Vigenère-Verschlüsselung des ersten Buchstabens W des Textes sucht er den Kreuzungspunkt der Zeile W mit der Spalte W und findet als Geheimtextbuchstaben das S. Der auf diese Weise vollständig verschlüsselte Geheimtext lautet:

```
SMCOOAUFKY EMOMOWIUOV LMOTK
```

Üblicherweise wird er in Gruppen fester Länge, beispielsweise in Fünfergruppen übertragen. Diese Maßnahme dient auch dazu, die Länge des Kennworts (hier zehn) nicht zu verraten. Der zu übermittelnde Geheimtext lautet hier:

```
SMCOO AUFKY EMOMO WIUOV LMOTK
```

Der befugte Empfänger ist, wie der Absender, im Besitz des geheimen Kennworts (hier: WILLKOMMEN) und kann durch Umkehrung der oben beschriebenen Verschlüsselungsschritte aus dem Geheimtext durch Entschlüsselung mithilfe des Kennworts den ursprünglichen Klartext wieder zurückgewinnen:

```
SMCOOAUFKYEMOMOWIUOVLMOTK
WILLKOMMENWILLKOMMENWILLK
WERDEMITGLIEDBEIWIKIPEDIA
```

Vigenere Chiffre Erklärung

# Kryptoanalyse

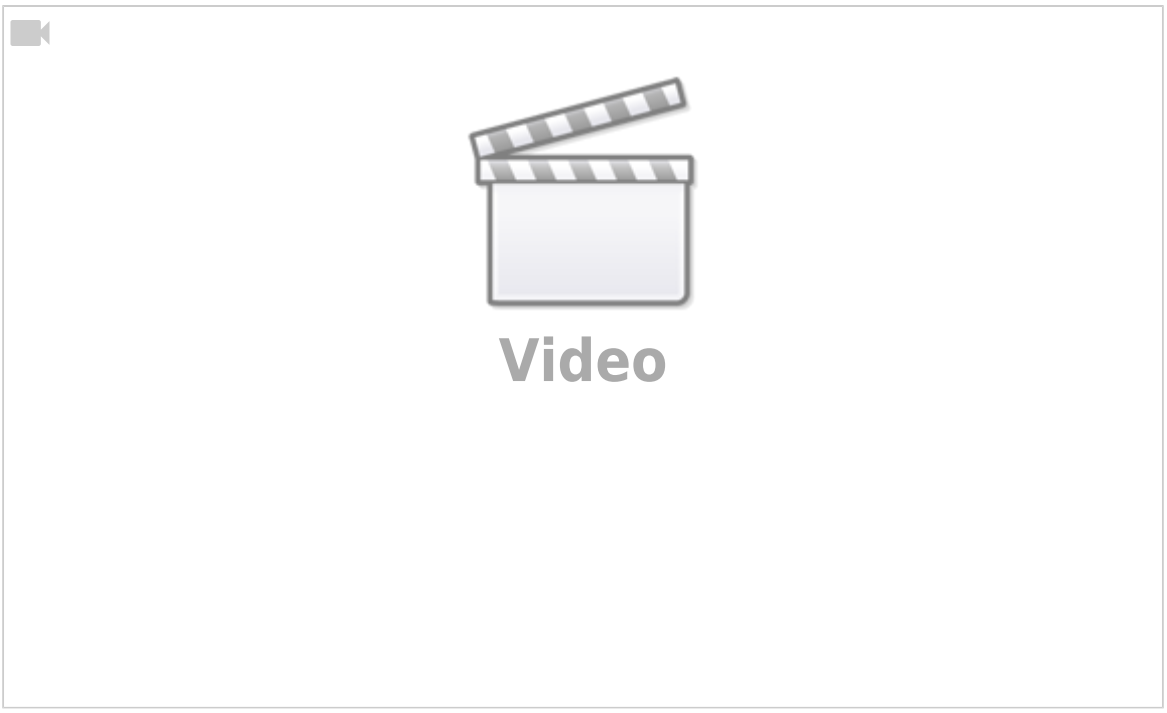
Vorteile einer polyalphabetischen Methode wie der Vigenère-Chiffre gegenüber den in den damaligen Jahrhunderten üblichen einfachen monoalphabetischen Methoden – dazu gehören auch die damals sehr beliebten Nomenklatoren – ist das durch die Verwendung von vielen unterschiedlichen Alphabeten bewirkte Abschleifen des bei den monoalphabetischen Verfahren so verräterischen Häufigkeitsgebirges. Der systematische Wechsel der Alphabete stärkt das Verfahren gegenüber statistischen Angriffsmethoden. Auch der erst im 20. Jahrhundert entwickelte Koinzidenzindex, ein universell einsetzbares kryptanalytisches Hilfsmittel, wird bei polyalphabetischen Verfahren wesentlich abgeschwächt. Lange wurde – abgesehen von Ausnahmen, in denen der Codeknacker das Schlüsselwort oder Teile des Klartextes erraten konnte – keine systematische Angriffsmethode gegen die Vigenère-Verschlüsselung gefunden, die sich über die Jahrhunderte den Ruf einer „unknackbaren Chiffre“ erwarb. Dennoch wurde sie nur selten verwendet und stattdessen lieber auf die althergebrachten Verfahren, wie Nomenklatoren, zurückgegriffen, wohl auch, weil viele Anwender die

Chiffre als zu kompliziert in der Anwendung empfanden.

Im Jahr 1854 fand der englische Wissenschaftler Charles Babbage (1791–1871) eine Lösung der Chiffre, die er jedoch nie publizierte. Der Erste, der eine allgemeingültige Angriffsmethode auf die Vigenère-Chiffre beschrieb, war der preußische Infanteriemajor und Kryptologe Friedrich Wilhelm Kasiski (1805–1881). Er veröffentlichte 1863 in Berlin sein Buch „Die Geheimschriften und die Dechiffrir-Kunst“ und erläuterte darin seine Idee zur Entzifferung von Vigenère-verschlüsselten Texten. Seine Entzifferungsmethode ist noch heute unter seinem Namen als Kasiski-Test bekannt. Als Erstes ist die Länge des verwendeten Schlüsselworts zu ermitteln. Dazu durchsuchte Kasiski den Geheimtext nach Buchstabenfolgen der Länge zwei (Bigramme) oder länger (Trigramme, Tetragramme etc.), die mehrmals vorkommen, genannt: „Doppler“. Anschließend bestimmte er den Abstand zwischen den Dopplern. Er erzeugte so eine möglichst vollständige Liste mit im Geheimtext auftretenden Dopplern und deren Abständen. In dieser suchte er mithilfe der Faktorisierung (Primfaktorzerlegung) nach gemeinsamen Längen, um so auf die vermutliche Schlüsselwortlänge zu schließen. Im Cryptologia-Artikel Breaking Short Vigenère Ciphers (siehe Literatur) ist die wichtige Seite 41 aus Kasiskis Buch abgebildet.[9] Nach der Untersuchung seines Vigenère-verschlüsselten Beispieltextes mit 180 Buchstaben zieht er das Fazit: „Hier kommt der Faktor 5 am häufigsten vor, der Schlüssel muß demnach 5 Buchstaben enthalten.“

Hat man die Schlüssellänge gefunden, so kann man im zweiten Schritt der Entzifferung den Geheimtext in seine Bestandteile zerlegen, die mit jeweils demselben Alphabet verschlüsselt wurden. In Kasiskis Beispielfall würde man den ersten, sechsten, elften Buchstaben und so fort als erste Gruppe betrachten. Die zweite Gruppe besteht aus dem zweiten, siebten, zwölften Buchstaben und so fort. Die dritte aus dem dritten, achten, dreizehnten und so weiter. Innerhalb jeder Gruppe liegt eine einfache Caesar-Verschlüsselung vor, die mithilfe der Häufigkeitsanalyse leicht zu knacken ist. In vielen Fällen entspricht schlicht der am häufigsten auftretende Geheimtextbuchstabe jeder Gruppe dem Klartext-„e“, also dem in den meisten europäischen Sprachen häufigsten Buchstaben. Hat man das „e“ identifiziert, dann ergeben sich unmittelbar alle anderen Buchstaben, denn die Vigenère-Chiffre benutzt ja nur verschobene Alphabete und keine verwürfelten, wie es der Namensgeber eigentlich vorgeschlagen hatte.

Nach „Rohrbachs Forderung“ sollte der Codeknacker zum Schluss seiner Arbeit noch versuchen, das Schlüsselwort zu erschließen. Erst dann gilt seine Arbeit als erfolgreich beendet. Im Idealfall gelingt ihm dies einfach mit Kenntnis des Klartextes durch anschließendes direktes Ablesen im Quadrat. In der Praxis wurden jedoch nicht immer plump einfache Wörter als Schlüssel benutzt. Dann gilt es, auch noch den Algorithmus zu erschließen, nach dem der Verschlüssler das Schlüsselwort (beispielsweise aus einem Merksatz) bildet und möglichst auch, wie und in welchem Rhythmus er es wechselt.



# One Time Pad (Vernam-Chiffre)

Das One Time Pad (Abk. OTP, dt. Einmalverschlüsselung) Verschlüsselungsverfahren, auch Vernam Verschlüsselung nach seinem Erfinder.

Vernam arbeitete bei der US-amerikanischen Telefongesellschaft AT&T und war dort mit der damals noch neuen Fernschreiber-Technik betraut. Ein Problem war, dass man Fernschreiben leicht abhören konnte - besonders, wenn diese per Funk übertragen wurden. 1917 hatte er eine Idee für die Lösung: er wollte die Bits des Baudot-Codes, den man damals für Fernschreiben nutzte und die aus einer Null oder einer Eins bestanden, verschlüsseln, indem er jedes Bit mit einem zufälligen, anderen Bit kombinierte. Dazu benutzte er einen zweiten Lochstreifen mit Zufallsmuster zum ersten mit der Botschaft. Zuerst nahm er nur einen kurzen Lochstreifen, dessen Ende er an den Anfang klebte und so einen sich wiederholenden Endlosstreifen erhielt. Doch dann merkte er, dass absolute Sicherheit nur ein Schlüsselstreifen bieten konnte, der genau so lang war wie der Lochstreifen mit dem Klartext.

Der amerikanische Major (und später General) Joseph O. Mauborgne setzte die Idee 1918 als Erster für militärische Zwecke um und erweiterte sie um die Prämisse, dass ein Schlüsselcode zufällig und nur einmal benutzt werden darf. Das Verfahren wurde als One-time-system bekannt. Aus Gründen der Praktikabilität verwendete er allerdings einen sich wiederholenden Schlüssel. Alsbald beschäftigten sich auch die Deutschen mit dem Verfahren und setzten es im diplomatischen Dienst der Weimarer Republik ein.

One Time Pad ist also ein Verfahren, bei dem jedes Zeichen des Klartextes mit einem Zeichen eines Schlüssels kombiniert wird, um zu einem Chiffrat zu gelangen. Dies bedeutet aber auch, dass der Schlüssel genau so lang sein muss wie der zu verschlüsselnde Text.

Damit das Verfahren sicher ist, ist es außerdem wichtig, dass der Schlüssel rein zufällig ist und dass der Schlüssel nur ein einziges mal verwendet wird. Denn würde der Schlüssel zweimal verwendet und wäre dem Gegner bekannt, dass zweimal derselbe Schlüssel für zwei unterschiedliche Klartexte verwendet wurden, so ließe sich ein Datenstrom aus den Differenzen erstellen, der wiederum durch Häufigkeitsanalyse der verwendeten Zeichen angreifbar wäre.

Von Prinzip her könnte man die One Time Pad Verschlüsselung auch als polyalphabetische Substitution bezeichnen, bei dem für jedes Zeichens des Klartextes ein anderer Schlüssel verwendet wird.

Auf der anderen Seite stellt die Länge des Schlüssels doch einige Anforderungen bei längeren Texten, so dass der Schlüssel wohl zumeist der Output eines Pseudo-Zufallsgenerators sein wird, wobei dann der Terminus Stromchiffre wieder passen würde.

Ein Vorteil des One Time Pad Verfahrens ist außer der Sicherheit bei richtiger Anwendung auch, dass es leicht mit Papier und Bleistift bewerkstelligt werden kann. So war es im kalten Krieg unter Geheimdiensten oft eingesetzt. Dabei wurden die Zeichen einer Geheimbotschaft mittels Dekodierschablonen zu Ziffern umgewandelt, die dann mittels langen Ziffernkolonnen in einem Heft oder Block, sogenannte Wurmtabellen kombiniert wurden.

Z. B.: Klartext 6, Schlüsselziffer 7 ergibt 13 → 3 (Addition Modulo 10). Mit der Umkehrrechnung (Subtraktion Absolut), hier also 7 - 13 = -6 → 6 konnte dann wieder auf den Klartext entschlüsselt werden. Ein einmal verwendete Wurmtabelle wurde nach dem Verschlüsseln dann nach einem festen Muster (z. B. alle angefangenen Blätter) vernichtet.

Auch der Heiße Draht (das sogenannte Rote Telefon) zwischen dem amerikanischen Präsidenten und dem sowjetischen Generalsekretär wurde durch ein One Time Pad Verfahren gesichert.

Das OTP lässt sich auch einfach per Computer realisieren. Die Bits der dort vorliegende Binärdaten werden dann aber meistens mittels XOR verknüpft, weil dies weniger Rechenoperationen erfordert. Außerdem ist XOR eine reversible Operation und kann so für Ver- und Entschlüsselung zugleich eingesetzt werden.

Der Hauptnachteil des OTP in der modernen Umgebung liegt in der erforderlichen Schlüssellänge. Wollte man zum Beispiel eine gesamte Festplatte verschlüsseln, so bräuchte man eine zweite, mindestens genau so große, die den Schlüssel enthält. Noch dazu müsste der Schlüssel aus echten Zufallszahlen und nicht aus berechneten Pseudozufallszahlen bestehen, um wirklich sicher zu sein. Dies würde viel Aufwand bedeuten. Außerdem kann der Schlüssel bei dieser Größe nicht mehr gemerkt werden, so dass er an Medien gebunden ist, die dem Feind in die Hände fallen könnten.

Darum hat das OTP Verfahren in der Moderne zunehmend an Bedeutung verloren, insbesondere, wenn größere Datenmengen verschlüsselt werden müssen.

# Beispiel

Eine einfache Handmethode zur Verschlüsselung ist beispielsweise die buchstabenweise Addition von Klartext und Schlüssel. Hierzu ersetzt man zunächst mithilfe einer beliebigen Substitutionstabelle die Buchstaben des Klartextalphabets durch Zahlen. Im einfachsten Fall ordnet man den 26 Großbuchstaben des lateinischen Alphabets Zahlen zu, die ihrer Position im Alphabet entsprechen. Mit anderen Worten, man nummeriert das Alphabet wie folgt durch:

```
A B C D E F G H I J  K  L  M  N  O  P  Q  R  S  T  U  V  W  X  Y  Z
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
```

Jetzt ist eine buchstabenweise Addition leicht möglich. Beispielsweise ergibt die Addition von A und F den Buchstaben G, entsprechend ihren Platznummern 1 + 6 = 7. Falls die Summe den Wert 26 überschreiten sollte, so zieht man einfach 26 ab (Modulo-Operation) und erhält so wieder einen der 26 Alphabetbuchstaben. Beispielsweise X plus U ist numerisch 24 + 21 = 45, nach Abziehen von 26 ergibt sich 19 und damit der Buchstabe S, also X + U = S.

Die Zusammenhänge bei der Addition von Buchstaben lassen sich an der folgenden Tabelle, die Ähnlichkeit mit einer klassischen Tabula recta (Vigenere Quadrat) hat, übersichtlich darstellen.

Zur Verschlüsselung wird man einen zufälligen Schlüssel benutzen, der in diesem Beispielfall passenderweise ebenfalls aus den 26 Großbuchstaben zusammengesetzt ist und dessen Länge (mindestens) der Länge des zu verschlüsselnden Klartextes entspricht. Entscheidend für die Sicherheit der Verschlüsselung ist, dass die einzelnen Buchstaben des Schlüssels wirklich zufällig verteilt sind, unvorhersagbar sind und in keinerlei Zusammenhang untereinander stehen. Als Beispiel für einen zufälligen Schlüssel dient die folgende Buchstabenfolge:

```
S = WZSLXWMFQUDMPJLYQOXXB
```

Der Schlüssel S ist in diesem Beispiel recht kurz, er umfasst nur 21 Buchstaben und ist bei bestimmungsgemäßer Verwendung sehr schnell verbraucht, nämlich bereits nach Verschlüsselung eines Textes aus 21 Buchstaben.

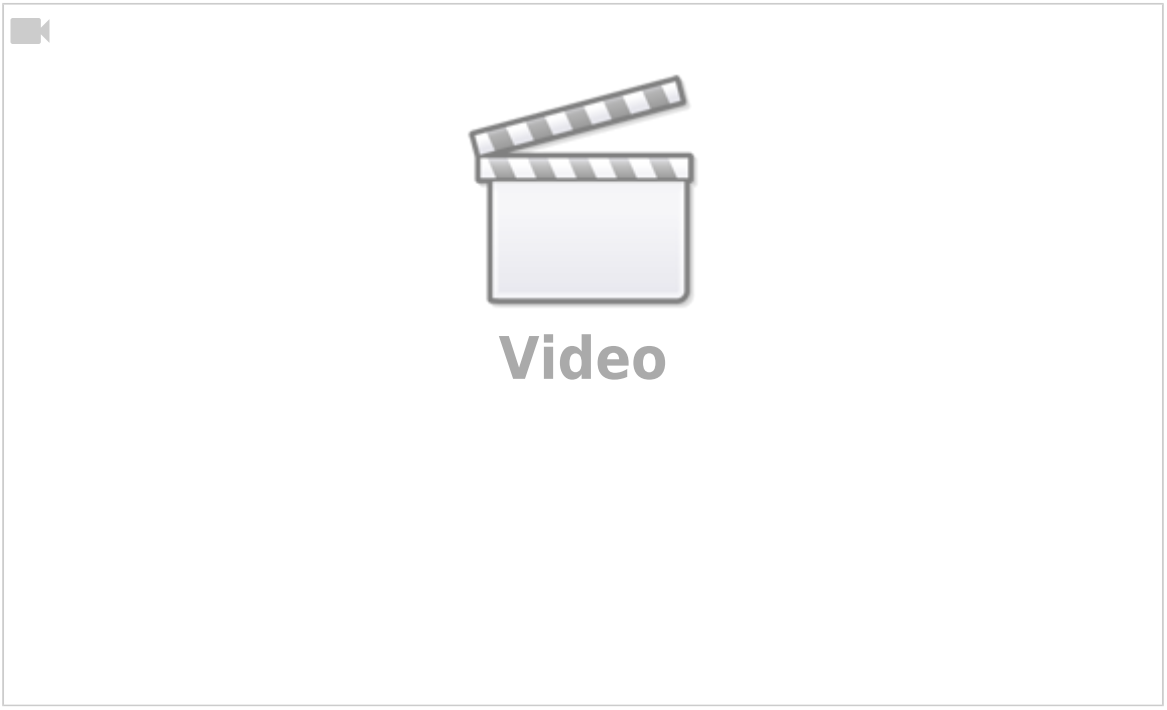
Beispielsweise soll der folgende Klartext K verschlüsselt werden:

```
K = ANGRIFFIMMORGENGRAUEN
```

Zur Verschlüsselung werden Klartext K und Schlüssel S, wie oben erläutert, buchstabenweise addiert. Als Summe (K + S = G) erhält man nach der so durchgeführten Einmalverschlüsselung den Geheimtext G:

```
G = XNZDGCSODHSEWOZFIPSCP
```

Der im Ergebnis erhaltene Geheimtext G ist von einem Zufallstext nicht zu unterscheiden und kann prinzipiell mit keiner noch so gearteten kryptanalytischen Angriffsmethode (weder jetzt noch in Zukunft) entziffert werden. Allein die Kenntnis des Schlüssels S erlaubt es, aus dem Geheimtext G durch Subtraktion des Schlüssels wieder den Klartext K zu gewinnen. Ohne den Schlüssel kann man prinzipiell alle denkbaren und mehr oder weniger sinnvollen Buchstabenkombinationen aus 21 Buchstaben konstruieren. Theoretisch könnte ein Angreifer dies probieren. Das wären aber 26^21 = 518 131 871 275 444 637 960 845 131 776 Möglichkeiten.



# Skytale

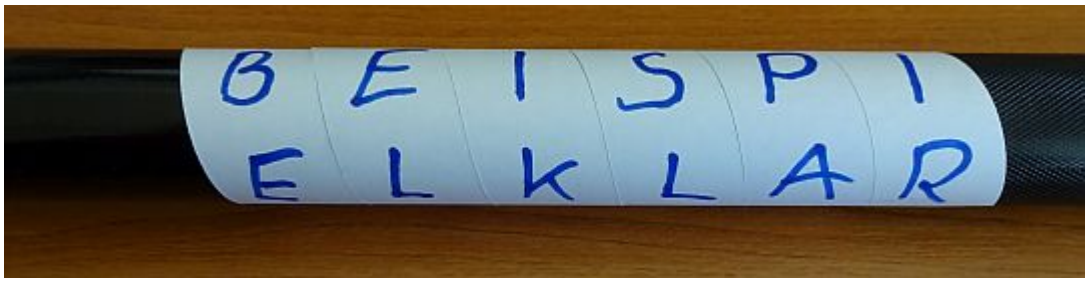
Die Skytale von Sparta (griech.: „scytale“; Stock, Stab) ist das älteste (5. Jh. v. Chr.) bekannte militärische Verschlüsselungsverfahren und basiert auf einem Stock mit einem bestimmten Durchmesser, auf den ein Lederstreifen wendelförmig gewickelt wurde. Dann wurde die Nachricht quer über den Stab auf das Leder geschrieben. Nach dem Abwickeln waren dann alle Buchstaben durcheinander und konnten erst wieder gelesen werden, wenn sie um einen Stab mit dem richtigen Durchmesser gewickelt wurden.

Der Durchmesser entspricht dem Versatz, also dem Schlüssel dieser Transpositions-Chiffre. Bitte beachten Sie, dass auch Leerzeichen mitkodiert werden, da diese ja eine Leerraum darstellen und somit einen Versatz bedeuten. Sollen keine Leerzeichen mitkodiert werden, löschen Sie diese vorher.

Die Chiffre ist nicht sonderlich sicher, denn man einfach Stöcke verschiedener Durchmesser ausprobieren. Oder mathematisch die Versätze durchrechnen. Dann muss man nur noch den Klartext erkennen.

Den Lederstreifen, auf den die Nachricht geschrieben war, konnte man auch umgedreht als Gürtel tragen, so dass er nicht weiter auffiel. Damit war die Skytale auch eine frühe Form der Steganografie.

## Beispiel




Klartext:

```
BEISPIELKLARTEXT
```

Schlüssel:

```
6
```

Kodiert:

```
BETELEIKXSLTPAIR
```

Chiffrierung Versatz 6:

```
1 2 3 4 5 6
B E I S P I
E L K L A R
```

```
T E X T
^ ^ ^ ^ ^ ^ --- spaltenweise auslesen

BET ELE IKX SLT PA IR
```

Dechiffrierung per Bruteforce:

```
01: BETELEIKXSLTPAIR
02: BXESTLETLPEAIIKR
03: BITEKPTXAESILLRE
04: BLXPEESATILIEKTR
05: BLKLAEEXTITISPRE
06: BEISPIELKLARTEXT
07: BEIXLPIELKSTARTE
08: BTLIXLPIEEEKSTAR
09: BTLIXLPIREEEKSTA
10: BTLIXLPAIREEEKST
11: BTLIXLTPAIREEEKS
12: BTLIXSLTPAIREEEK
13: BTLIKXSLTPAIREEE
14: BTLEIKXSLTPAIREE
15: BTELEIKXSLTPAIRE
16: BETELEIKXSLTPAIR
```

# DES (Data Encryption Standard)

- Entwickelt Anfang der 70er Jahre bei IBM („Lucifer“, „Feistelchiffre“)
- 1977 wurde er von der US-Standardisierungsbehörde NIST (National Institute of Standards and Technology) als Standard anerkannt
- NSA an der Entwicklung beteiligt und die Designkriterien unter Verschluss gehalten sowie Schlüssellänge verkürzt
- DES gilt als Muster aller modernen Chiffren
- DES ist eine Blockchiffre (64 Bit Blöcke)
- Schlüssellänge 56 Bit (+8 Bit Prüfsumme)
- Entwickelt für Hardwareverschlüsselung

Der Data Encryption Standard (DES) ist eine Blockchiffre mit 8 Byte Blocklänge und weit verbreiteter symmetrischer Verschlüsselungsalgorithmus und wurde als offizieller Standard für die US-Regierung im Jahr 1977 bestätigt und wird seither international vielfach eingesetzt. Seine Entstehungsgeschichte hat wegen der Beteiligung der NSA am Design des Algorithmus immer wieder Anlass zu Spekulationen über seine Sicherheit gegeben. Heute wird DES aufgrund der verwendeten Schlüssellänge von nur 56 Bits für viele Anwendungen als nicht ausreichend sicher erachtet.

Ursprünglich hieß der bei IBM unter der Leitung von Horst Feistel entwickelte Algorithmus Lucifer und bot eine Schlüssellänge von 16 Byte / 128 Bit. Die NSA soll dafür verantwortlich gewesen sein, dass die Schlüssellänge für DES auf 56 Bit gekürzt wurde, weil dies die Schlüssellänge ist, die man mit den Supercomputern bei der NSA in den 1970er-Jahren gerade noch so per Brute Force hätte knacken können.

Die Schlüssellänge kann durch Mehrfachanwendung des DES jedoch auf einfache Weise vergrößert werden. Als Triple-DES, auch als TDES, 3DES oder DESede bezeichnet, wird der DES weiterhin am häufigsten, zum Beispiel von Banken in Chipkartenanwendungen, eingesetzt, obwohl der TDES als offizieller Standard für die USA durch den Advanced Encryption Standard (AES) abgelöst wurde.

Weil die Schlüssellänge nur 56 Bit beträgt, konnte DES bereits durch Brute-Force-Angriffe gebrochen werden, indem systematisch alle möglichen Schlüssel (256 = ca. 72 Billiarden) getestet wurden. Die EFF baute 1998 eine etwa 250.000 Dollar teure Maschine mit dem Namen Deep Crack. Dieser Superrechner enthielt 1536 spezielle Krypto-Chips und konnte pro Sekunde etwa 88 Milliarden Schlüssel testen. Im Juli 1998 gelang es mit dieser Maschine, einen DES-Code in 56 Stunden zu knacken.

Die einzige andere öffentlich bekannte Maschine zum Brechen von DES ist COPACOBANA. Sie wurde 2006 an den Universitäten Bochum und Kiel gebaut. Im Gegensatz zu Deep Crack besteht eine COPACOBANA aus rekonfigurierbaren Hardware-Bausteinen, sog. FPGAs. COPACOBANA kann 65 Milliarden DES-Schlüssel pro Sekunde testen, woraus sich eine durchschnittliche Suchzeit von 6,4 Tagen für eine DES-Attacke ergibt. Durch den Einsatz rekonfigurierbarer Hardware kann COPACOBANA auch zum Brechen anderer Chiffren wie A5 eingesetzt werden. Die Material- und Herstellungskosten von COPACOBANA belaufen sich auf „nur“ etwa 10.000 Dollar.

## Verfahren

Bei DES handelt es sich um einen symmetrischen Algorithmus, das heißt zur Ver- und Entschlüsselung

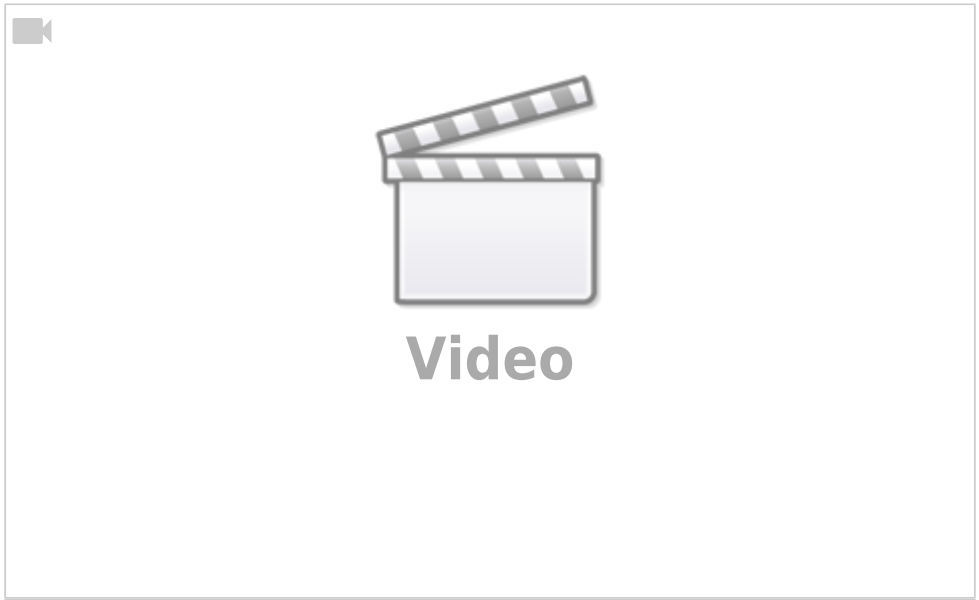
wird derselbe Schlüssel verwendet. DES funktioniert als Blockchiffre, jeder Block wird also unter Verwendung des Schlüssels einzeln chiffriert, wobei die Daten in 16 Runden von Substitutionen und Transpositionen (Permutation) nach dem Schema von Feistel verwürfelt werden.

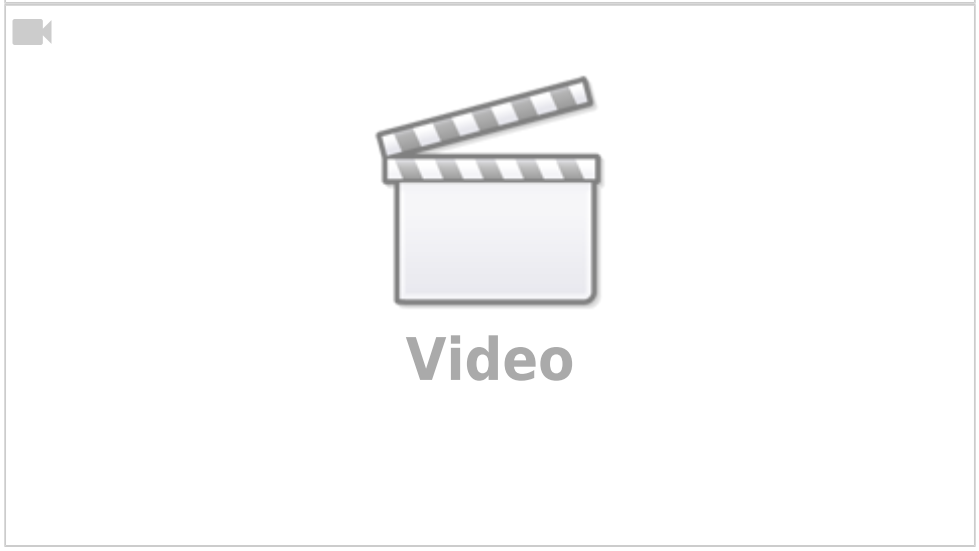
Die Blockgröße beträgt 64 Bits, das heißt ein 64-Bit-Block Klartext wird in einen 64-Bit-Block Chiffretext transformiert. Auch der Schlüssel, der diese Transformation kontrolliert, besitzt 64 Bits. Jedoch stehen dem Benutzer von diesen 64 Bits nur 56 Bits zur Verfügung; die übrigen 8 Bits (jeweils ein Bit aus jedem Byte) werden zum Paritäts-Check benötigt. Die effektive Schlüssellänge beträgt daher nur 56 Bits. Die Entschlüsselung wird mit dem gleichen Algorithmus durchgeführt, wobei die einzelnen Rundenschlüssel in umgekehrter Reihenfolge verwendet werden.

Auf den 64 Bit Block wird eine initiale Permutation angewandt. Danach wird der Block in zwei Teile aufgeteilt und jeder Teil in ein 32 Bit Register gespeichert, auf die das Prinzip eines Feistel-Netzwerkes angewandt wird.

Der DES-Algorithmus beschreibt zunächst nur, wie ein Datenblock mit 64 Bits verarbeitet wird. Zur Verarbeitung einer Nachricht beliebiger Länge lässt sich der DES wie auch jede andere Blockchiffre in verschiedenen Betriebsmodi verwenden. Für bestimmte Betriebsmodi, wie zum Beispiel ECB oder CBC, ist ein Auffüllen des Klartextes auf ein Vielfaches der vollen Blocklänge notwendig (Padding). Dies geschieht indem die Bitfolge 1000… angehängt wird.

Die genaue Spezifikation findet sich als FIPS 46-3 beim NIST.



# AES (Advanced Encryption Standard) / Rijndael-Chiffre

AES steht für 'Advanced Encryption Standard', ist eine Blockchiffre und der Sieger-Algorithmus einer Ausschreibung in 2000 des NIST und gilt als Nachfolger von DES (Data Encryption Standard) von 1977. Der Algorithmus wurde von Joan Daemen und Vincent Rijmen entwickelt und die Chiffre wird deshalb auch Rijndael-Chiffre genannt. AES lässt einem die Wahl bei der Schlüssellänge von 128, 192 und 256 Bit. AES-192 und AES-256 sind in den USA für staatliche Dokumente mit höchster Geheimhaltungsstufe zugelassen.

AES benutzt eine Blocklänge von 16 Bytes, das heißt, dass ein Chiffrat 15 Zeichen länger werden kann als der ursprüngliche Klartext. Es empfiehlt sich, als Schlüssel den Hash eines Klartextpasswortes zzgl. eines (wenn gewünscht gehashten) Salts zu benutzen, z. B. SHA-256 für die 256-bit-Variante von AES. Dies ergibt eine gute Sicherheit..

AES hat sich mittlerweile als Standard durchgesetzt und neuere CPUs enthalten inzwischen spezielle Instruktionen, um die Verschlüsselung damit zu beschleunigen. Er wird u. a. bei der WLAN-Verschlüsselung WPA2, bei SSH, IPSec und in der IP-Telefonie benutzt.

Während Rijndael der Sieger-Algorithmus und zukünftiger Namensträger von AES wurde, waren die folgende vier weiteren Kandidaten in der engere Auswahl für AES gezogen worden, haben es letztendlich aber nicht geschafft: MARS, RC6, Serpent und Twofish. Weitere Kandidaten, die es nicht in die Endrunde schafften sind: CAST-256, CRYPTON, DEAL, DFC, E2, FROG, HPC, LOKI97, MAGENTA und SAFER+.

## Beschreibung

Der in AES implementierte Algorithmus heißt Rijndael und ist ein als Substitutions-Permutations-Netzwerk entworfene Blockchiffre. Jeder Block wird zunächst in eine zweidimensionale Tabelle mit vier Zeilen geschrieben, deren Zellen ein Byte groß sind. Die Anzahl der Spalten variiert je nach Blockgröße von 4 (128 Bits) bis 8 (256 Bits). Jeder Block wird nun nacheinander bestimmten Transformationen unterzogen. Aber anstatt jeden Block einmal mit dem Schlüssel zu verschlüsseln, wendet Rijndael verschiedene Teile des erweiterten Originalschlüssels nacheinander auf den Klartext-Block an. Die Anzahl der Runden variiert und ist von Schlüssellänge und Blockgröße abhängig (bei AES also nur von der Schlüssellänge).

Eine S-Box (Substitutionsbox) mit 256 Bytes dient als Basis für eine monoalphabetische Verschlüsselung. Sie gibt an, wie in jeder Runde jedes Byte eines Blocks durch einen anderen Wert zu ersetzen ist. Typischerweise wird die S-Box in Blockchiffren eingesetzt, um die Beziehung zwischen Klar- und Geheimtext zu verwischen (in der kryptologischen Fachsprache Konfusion genannt). Die S-Box des AES setzt auch teilweise das Shannon’sche Prinzip der Diffusion um. Die Konstruktion der S-Box unterliegt Designkriterien, die die Anfälligkeit für die Methoden der linearen und der differentiellen Kryptoanalyse sowie für algebraische Attacken minimieren sollen.

Die genaue Spezifikation findet sich beim NIST.