11. Foliensatz Computernetze

Prof. Dr. Christian Baun

Frankfurt University of Applied Sciences
(1971–2014: Fachhochschule Frankfurt am Main)
Fachbereich Informatik und Ingenieurwissenschaften
christianbaun@fb2.fra-uas.de

Lernziele dieses Foliensatzes

- Einführung in die Kryptologie – Teil 1
  - Grundbegriffe der Kryptologie
  - Kryptoanalytische Angriffe
  - Symmetrische Verfahren
    - Skytala
    - Freimaurerchiffre
    - Caesar-Verschlüsselung
    - Häufigkeitsanalyse
    - Chiffrierscheibe
    - Polybius
    - Playfair
    - Vigenère
    - One-Time-Pad
    - ENIGMA
    - DES
    - Triple-DES
    - AES

Motivation

- Netzwerke dienen der Kommunikation zwischen Benutzern untereinander und IT-Komponenten
- Ein Problem ist von je her der Schutz der Nachrichten
  - In dieser Vorlesung werden die Grundlagen der Kryptologie und klassische sowie aktuelle kryptographische Verfahren behandelt
- Es existieren verschiedene kryptographische Verfahren
  - Diese haben Ihre Stärken und Schwächen
- Ein kryptographisches Verfahren verfolgt in der Regel mindestens eines der folgenden Ziele:
  1. **Geheimhaltung**: Das Lesen einer Nachricht für Unbefugte unmöglich bzw. schwer zu machen
  2. **Authentifizierung**: Identitätsnachweis des Senders gegenüber dem Empfänger
  3. **Integrität**: Die Nachricht darf nicht durch Unbefugte verändert werden
  4. **Verbindlichkeit**: Der Sender kann später nicht leugnen, eine Nachricht abgeschickt zu haben

Literatur

- Albrecht Beutelspacher. **Geheimsprachen**. C.H.Beck. 2005
- Albrecht Beutelspacher. **Kryptologie**. Vieweg+Teubner. 2009
- Albrecht Beutelspacher, Jörg Schwenk, Klaus-Dieter Wolfenstetter. **Moderne Verfahren der Kryptographie: Von RSA zu Zero-Knowledge**. Vieweg+Teubner. 2010
- Albrecht Beutelspacher, Heike Neumann, Thomas Schwarzpaul. **Kryptografie in Theorie und Praxis: Mathematische Grundlagen für Internetsicherheit, Mobilfunk und elektronisches Geld**. Vieweg+Teubner. 2009
- Simon Singh. **Geheime Botschaften**. Deutscher Taschenbuch Verlag. 2001

Grundbegriffe der Kryptologie (1/2)

- **Kryptographie** wird verstanden als die Lehre von der Absicherung von Nachrichten durch Verschlüsseln
- **Kryptoanalyse** ist die Kunst Chiffretexte aufzubrechen, also den Klartext ohne Kenntnis des Schlüssels zu reproduzieren
- **Kryptologie** vereinigt Kryptographie und Kryptoanalyse
- Bei der **Steganographie** wird die Existenz der Nachricht durch Verstecken in einem *harmlosen* Text oder Bild verheimlicht
- Ein **Alphabet** A ist eine endliche Menge von Zeichen
- Der lesbare Text M einer **Nachricht** (Message) wird **Klartext** (Plaintext) genannt
  - Er wird als Zeichenkette über dem Alphabet A gebildet
  - Also gilt: $M \in A^n,\ n \in \mathbb{N}$
    - A^n ist die Menge der Worte im Alphabet A mit der Länge n

Grundbegriffe der Kryptologie (2/2)

- **Verschlüsselung** oder **Chiffrierung** bezeichnet das Verfahren, eine Nachricht unverständlich zu machen
  - Die Chiffre E (Encryption) ist eine invertierbare Abbildung, welche aus dem Klartext M und einem Schlüssel K den Geheimtext C (Ciphertext), der auch auch eine Zeichenkette über vielleicht einem anderen Alphabet $B(C) \in B^m,\ m \in \mathbb{N}$ darstellt, erzeugt
- Die Länge des chiffrierten Textes C kann sich von der Länge des Klartextes unterscheiden
  - Sind die Längen gleich, so heißt E längentreu
- Die Umkehrung von E wird **Entschlüsselung** genannt und mit D (Decryption) bezeichnet
- Im Gegensatz zur Chiffrierung werden bei der **Codierung** nicht Buchstaben, sondern Wörter ersetzt
  - Beispiel: Navajo-Codewörter für Funksprüche im 2.Weltkrieg

Einfache Aufgabe zu Alphabeten und Chiffren

- Seien $A = \{a, b, c\}$ und $B = \{u, v, w, x, y, z\}$ Alphabete und C eine Chiffre, die A auf B abbildet und die folgende Vorschrift hat:

$$\begin{array}{lcll} a & \mapsto & u & \text{oder } x \\ b & \mapsto & v & \text{oder } y \\ c & \mapsto & w & \text{oder } z \end{array}$$

- Wobei $a \mapsto u$ oder x bedeutet, dass a zufällig entweder auf u oder x abgebildet wird
- **Frage**: Wie viele verschiedene Chiffretexte gibt es für aabba und wie viele Klartexte sowie Chiffretexte der Länge k gibt es insgesamt?

Einfache Aufgabe zu Alphabeten und Chiffren (Lösung)

- Es gibt $2^5 = 32$ Möglichkeiten

```
uuvvu uuvvx uuvyu uuvyx uuyvu uuyvx uuyyu uuyyx
uxvvu uxvvx uxvyu uxvyx uxyvu uxyvx uxyyu uxyyx
xuvvu xuvvx xuvyu xuvyx xuyvu xuyvx xuyyu xuyyx
xxvvu xxvvx xxvyu xxvyx xxyvu xxyvx xxyyu xxyyx
```

- Für einen Chiffretext der Länge k existieren, da jeder der 3 Buchstaben auf 2 verschiedene Werte abgebildet werden kann, 6^k Möglichkeiten
- Für einen Klartext existieren 3^k Möglichkeiten

Grundbegriffe der Kryptologie (3/3)

- Früher wurden so genannte **eingeschränkte Algorithmen** benutzt
  - Bei diesen hängt die Sicherheit davon ab, ob die Arbeitsweise des Algorithmus geheim bleibt
  - Heute benutzt man meistens Algorithmen mit Schlüssel
    - Der Algorithmus ist allgemein bekannt und nur der zugehörige Schlüssel muss geheim gehalten werden
    - Dieses Vorgehen wurde im 19.Jahrhundert von Auguste Kerckhoff gefordert ($\Longrightarrow$ **Prinzip von Kerckhoff**)
- **Symmetrische Kryptosysteme**: Alle Kommunikationspartner verwenden den gleichen Schlüssel
- **Asymmetrische Kryptosysteme**: Jeder Kommunikationspartner besitzt ein Schlüsselpaar, das aus einem geheimen Teil (privater Schlüssel) und einem nicht geheimen Teil (öffentlicher Schlüssel) besteht
  - Der Schlüssel, mit dem verschlüsselt wird, ist allgemein bekannt
  - Der Schlüssel zum entschlüsseln ist nur dem Empfänger bekannt

Arten von kryptoanalytischen Angriffen

- **Known-Plaintext-Angriff**: Der Kryptoanalytiker kennt zusätzlich zum Chiffretext den gehörenden Klartext oder einen Teil davon
- **Ciphertext-Only-Angriff**: Der Kryptoanalytiker verfügt über eine bestimmte Menge Chiffretext
- **Chosen-Plaintext-Angriff**: Der Kryptoanalytiker kann einen beliebigen Klartext vorlegen und den zugehörigen Chiffretext erhalten
- **Chosen-Ciphertext-Angriff**: Der Kryptoanalytiker kann einen beliebigen Chiffretext vorlegen und den zugehörigen Klartext erhalten
- **Brute-Force-Angriff**: Alle möglichen Schlüssel werden ausprobiert
- Inbesitznahme des Schlüssels mittels **Diebstahl** (Betrug) oder **Gewalt**

Einfache Aufgabe zu Brute-Force

- Berechnen Sie die Zeit für das Knacken des 1024 Bit großen Schlüssels einer 1024 Bit langen Blockchiffre mit einem Brute-Force-Angriff unter der Annahme, dass Sie einen Block von 1024 Bit im Klartext und im Chiffretext vorliegen haben
- Nehmen Sie hierzu an, Sie haben Zugriff auf einen Rechner, der pro Sekunde 1 Megabit verschlüsseln kann

Einfache Aufgabe zu Brute-Force (Lösung)

$$\frac{1\frac{\text{Mbit}}{\text{s}}}{1024\ \text{Bit}} = \frac{2^{20}\ \text{Bit}}{1024\ \text{Bit} * \text{s}} = \frac{1048576\ \text{Bit}}{1024\ \text{Bit} * \text{s}} = \frac{1024}{\text{s}}$$

- Der Rechner kann somit 1024 Versuche pro Sekunde durchführen
- Die Anzahl der möglichen Schlüssel ist: 2^{1024}
- Dauer zum Finden des Schlüssels:

$$\text{bis zu} \quad \frac{2^{1024}}{\frac{1024}{\text{s}}} = \frac{2^{1024}}{\frac{2^{10}}{\text{s}}} = 2^{1014}\text{s} \approx 5,6 * 10^{297}\ \text{Jahre}$$

- Im Schnitt dauert das Finden des Schlüssels nur die Hälfte der Zeit, die notwendig ist den gesamten Schlüsselraum zu durchsuchen
  - In diesem Fall: $\approx 2,8 * 10^{297}$ Jahre

Sicherheit von Algorithmen

- Ein Algorithmus gilt als sicher, wenn...
  - der zum Aufbrechen notwendige **Geldaufwand** den Wert der verschlüsselten Daten übersteigt
  - die zum Aufbrechen notwendige **Zeit** größer ist als die Zeit, die die Daten geheim bleiben müssen
  - das mit einem bestimmten Schlüssel chiffrierte **Datenvolumen** kleiner ist als die zum Knacken erforderliche Datenmenge

  Ein Algorithmus ist uneingeschränkt sicher, wenn der Klartext sogar beim bei beliebig viel vorhandenem Chiffretext nicht ermittelt werden kann
- Der **Schlüsselraum**, d.h. die Menge, aus der ein Schlüssel gewählt wird, sollte möglichst groß sein
  - Er sollte so groß sein, dass der Aufwand für einen Angriff unakzeptabel hoch wird

Symmetrische Verfahren

- Die Schlüssel zum Verschlüsseln und zum Entschlüsseln sind identisch
  - plaintext $\longrightarrow$ encryption $\longrightarrow$ ciphertext $\longrightarrow$ decryption $\longrightarrow$ plaintext
- **Transpositionsverfahren**: Der Geheimtext wird durch eine Permutation der Klartextzeichen erzeugt (z.B. Skytala)
- **Substitutionsverfahren**, Jedes Zeichen oder jede Zeichenfolge des Klartextes wird durch ein anderes Zeichen oder eine Zeichenfolge ersetzt
  - **Monoalphabetische Substitution**: Jedem Zeichen oder jeder Zeichenfolge wird über ein Alphabet A eindeutig ein Zeichen oder eine Zeichenfolge über einem Alphabet B zugeordnet (z.B. Caesar, Playfair, Polybius)
  - **Polyalphabetische Substitution**: Jedem Zeichen oder jeder Zeichenfolge wird über ein Alphabet A eindeutig ein Zeichen oder eine Zeichenfolge über den Alphabeten $B_1, B_2, \ldots, B_n$ zugeordnet
  - **Monographische Substitution**: Ersetzt Einzelzeichen
  - **Polygraphische Substitution**: Ersetzt Zeichenfolgen

Skytala von Sparta

- Ältestes bekanntes militärische Verschlüsselungsverfahren
  - Es handelt sich um ein **Transpositionsverfahren**
- Die **Skytala** ist ein (Holz-)Stab mit einem bestimmten Durchmesser
- Wurde von den Spartanern bereits vor mehr als 2500 Jahren zum Verschlüsseln von Texten verwendet
- Ein **Papierstreifen**, **Pergamentband** oder **Lederband** wird um den Stab gewickelt und die Nachricht in Spalten quer darauf geschrieben
- Der Papierstreifen mit dem Geheimtext wird ohne den Stab an den Empfänger übermittelt
- Der geheime **Schlüssel** ist der Umfang der Skytala bzw. die Anzahl der Spalten von Buchstaben, die in einer Umdrehung auf die Skytala passen
- Der Empfänger musste eine Skytala mit genau dem gleichen Radius haben, um den Papierstreifen darum legen zu können
- Ohne die passende Skytala kann die Nachricht nicht gelesen werden, da die Buchstaben scheinbar willkürlich auf dem Band angeordnet sind

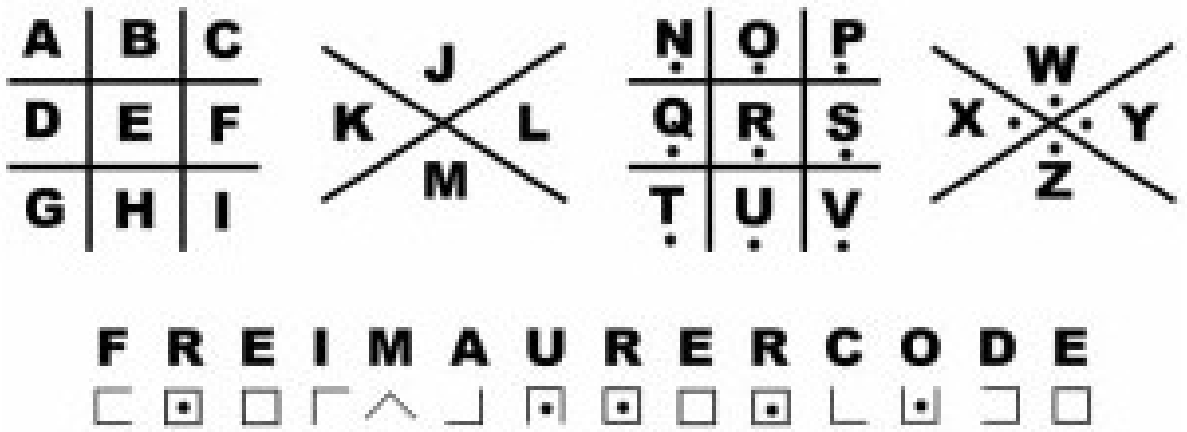
Einfaches Beispiel der Skytala

Bildquelle: `http://www.drymalianaxos.gr`

- Klartext:
  `Diese_Nachricht_ist_geheim.`
- Auf dem Papierstreifen:
  `Diese`
  `_Nach`
  `richt`
  `_ist_`
  `gehei`
  `m.`
- Chiffretext:
  `D_r_gmiNiie.eacshschteeht_i`

Freimaurerchiffre

- Einfaches Beispiel für **monographische Substitution**
- Klar- und Geheimtextalphabet haben die gleiche Kardinalität
  - Kardinalität = Anzahl der Elemente in dieser Menge
- Entspricht einer normalen Verschlüsselung mit einem Geheimtextalphabet (**monoalphabetische Substitution**)
  - Der einzige Unterschied ist, dass der Geheimtext aus Zeichen und nicht aus Buchstaben besteht



Quelle: Albrecht Beutelspacher. **Geheimsprachen**. C.H.Beck. 2005

Caesar-Verschlüsselung (auch Verschiebechiffre genannt)

- Gaius Julius Caesar verwendete für seine geheimen Kommunikation eine Verschiebung des Alphabets um drei Buchstaben
- Einfaches Verschlüsselungsverfahren, das auf der **monographischen** und **monoalphabetischen Substitution** basiert
- Eines der einfachsten und unsichersten Verschlüsselungsverfahren
- Jeder Buchstabe des Klartexts wird auf einen Geheimtextbuchstaben abgebildet
- Die Abbildung ergibt sich, indem man die Zeichen eines geordneten Alphabets um eine bestimmte Anzahl zyklisch nach rechts **verschiebt** (rotiert)
- Der **Schlüssel** ist die Anzahl der verschobenen Zeichen

```
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
Chiffretext: 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
```

- Zur Entschlüsselung wird das Alphabet um dieselbe Anzahl Zeichen nach links rotiert

Wie kann man die Caesar-Verschlüsselung brechen?

- Eine Möglichkeit auf die Lösung zu kommen ist **Ausprobieren**
  - Es existieren 26 Verschiebechiffren über dem natürlichen Alphabet (a,b,c,...,z), da es möglich ist um $0, 1, 2, 3, \ldots, 25$ Zeichen zu verschieben
    - Die Schlüsselanzahl 25 (entspricht einer Schlüssellänge von ca. 5 Bit)
  - Wenn man alle 26 Möglichen Verschiebungen testet, erhält man den Klartext
- Eine weitere Möglichkeit ist die Erstellung einer **Häufigkeitstabelle** zur **Häufigkeitsanalyse**

Vorgehensweise bei einer Häufigkeitsanalyse

1. Häufigkeit jedes Buchstabens feststellen
   - Der häufigste Buchstabe in der deutschen Sprache ist das `E`
2. Achten Sie auf Doppelbuchstaben. Die häufigsten sind:
   - Im Deutschen: `ss`, `nn`, `ll`, `ee`, `rr`
   - Im Englischen: `ss`, `ee`, `tt`, `ff`, `ll`, `mm`, `oo`
3. Enthält der Geheimtext Leerzeichen zwischen den Wörtern, versuchen Sie, die Wörter mit nur einem (Englisch), 2 oder 3 Buchstaben herauszufinden
   - Deutsch: Häufigste Wörter mit 2 Buchstaben: `am`, `in`, `zu`, `es`
   - Deutsch: Häufigste Wörter mit 3 Buchstaben: `die`, `der`, `und`, `den`, `daß`
   - Englisch: Die einzigen Wörter mit einem Buchstaben: `a`, `I`
   - Englisch: Häufigste Wörter mit 2 Buchstaben: `of`, `to`, `in`, `it`, `is`, `be`, `as`, `at`, `so`
   - Englisch: Häufigste Wörter mit 3 Buchstaben: `the`, `and`

Hypothetische Zeichenwahrscheinlichkeit in Prozent

Zeichen	deutsch	englisch	Zeichen	deutsch	englisch
a	6,47	8,04	n	9,84	7,09
b	1,93	1,54	o	2,98	7,60
c	2,68	3,06	p	0,96	2,00
d	4,83	3,99	q	0,02	0,11
e	17,48	12,51	r	7,54	6,12
f	1,65	2,30	s	6,83	6,54
g	3,06	1,96	t	6,13	9,25
h	4,23	5,49	u	4,17	2,71
i	7,73	7,26	v	0,94	0,99
j	0,27	0,16	w	1,48	1,92
k	1,46	0,67	x	0,04	0,19
l	3,49	4,14	y	0,08	1,73
m	2,58	2,53	z	1,14	0,09

Beispiel für eine erfolgreiche Häufigkeitsanalyse (1/2)

- Der Goldkäfer (Originaltitel: *The Gold-Bug*)
- Kurzgeschichte von Edgar Allan Poe
- In der Geschichte muss die Hauptfigur folgende Geheimschrift lösen:

```
53##+305))6*;4826)4#.)4#);806*;48+8$
60))85;1#(;:#*8+83(88)5*+;46(;88*96*
?;8)*#(;485);5*+2:*#(;4956*2(5*-4)8$
8*;4069285);)6+8)4##;1(#9;48081;8:8#
1;48+85;4)485+528806*81(#9;48;(88;4(
#?34;48)4#;161;:188;#?;
```

Beispiel für eine erfolgreiche Häufigkeitsanalyse (2/2)

- Lösungsweg:
  - Als Sprache des Klartextes die englische Sprache angenommen
  - Als verwendetes Verschlüsselungsverfahren wurde eine **monoalphabetische** Verschlüsselung angenommen
  - Es wurden die Häufigkeiten der häufigsten Zeichen ermittelt

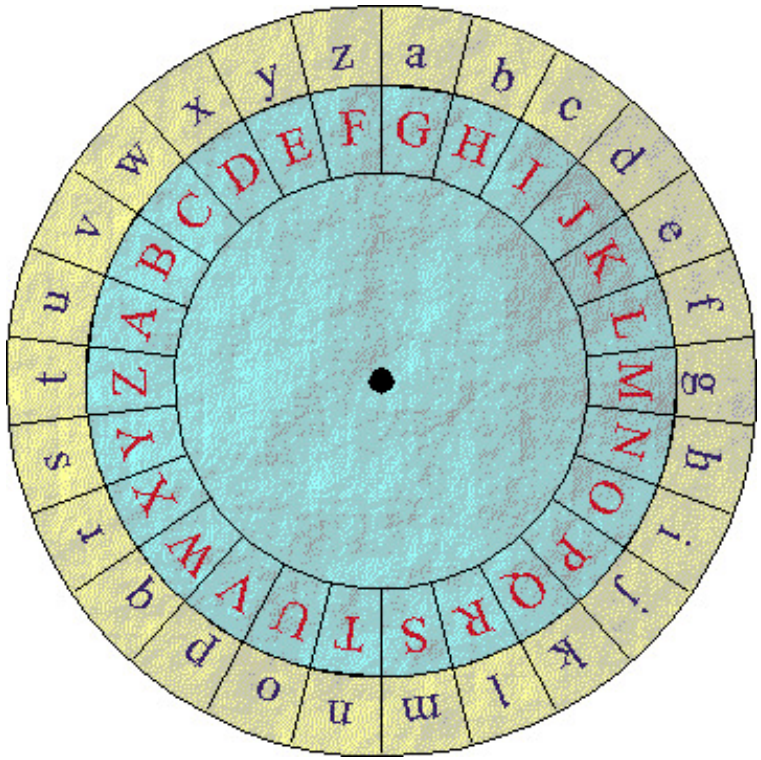
8 = 33 mal	; = 26 mal	4 = 19 mal	# = 16 mal	) = 16 mal
* = 13 mal	5 = 12 mal	(= 11 mal	6 = 11 mal	+ = 8 mal
1 = 8 mal	0 = 6 mal	9 = 5 mal	2 = 5 mal	: = 3 mal
? = 3 mal	3 = 3 mal	- = 1 mal	. = 1 mal	

- Es wurde mit Hilfe bekannter Zeichen versucht Wörter zu entschlüsseln und damit weitere Buchstaben in Erfahrung zu bringen
  - E ist der Buchstabe, der im Englischen am häufigsten auftritt
- Das Ergebnis ist folgender Klartext: *A good glass in the bishop's hostel in the devil's seat forty-one degrees and thirteen minutes northeast and by north main branch seventh limb east side shoot from the left eye of the death's head a bee line from the tree through the shot fifty feed out.*

Chiffrierscheibe (1/2)

Bildquelle: http://www.mathe.tu-freiberg.de

- Im 15.Jahrhundert entwickelte Leon Battista Alberti die Chiffrierscheibe
- Die Chiffrierscheibe ist eine Vereinfachung der Caesar-Verschlüsselung
- Die innere Scheibe wird um die Anzahl der verschobenen Buchstaben zur äußeren Scheibe gedreht
- So lassen sich die ersetzten Buchstaben einfach und schnell ablesen

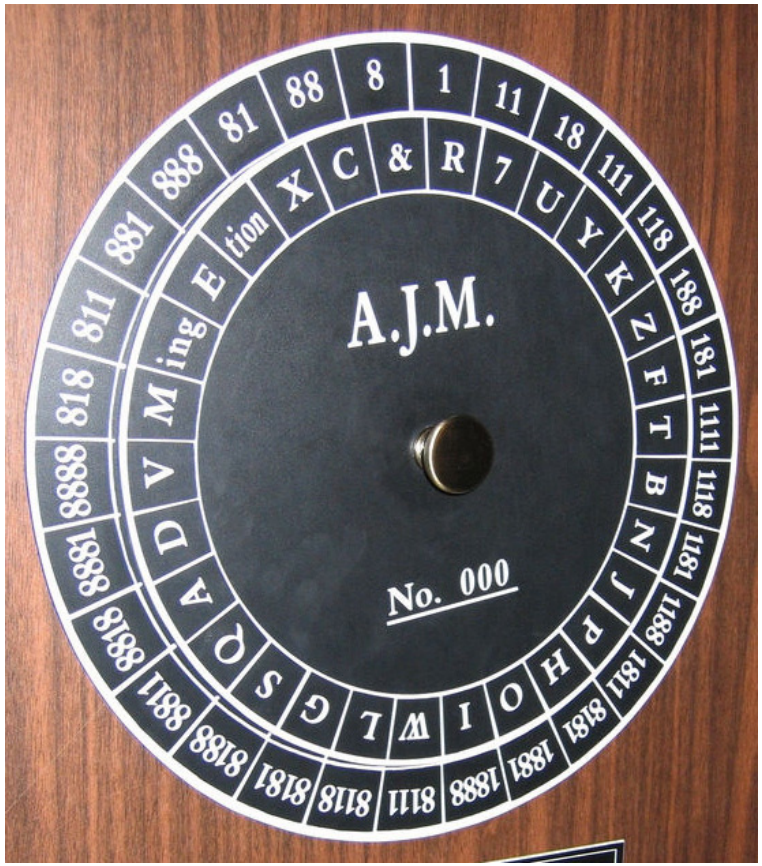


- Die Chiffrierscheibe bietet **monoalphabetische Substitution**
  - Verändert man während der Verschlüsselung verändert die Stellung der Scheiben zueinander hat man eine **polyalphabetische Substitution**

Chiffrierscheibe (2/2)

Bildquelle: Wikipedia

- Auf der äußeren Scheibe kann auch ein anderes Alphabet als auf der inneren Scheibe oder alternativ Symbole angegeben sein
- Die Abbildung zeigt eine Chiffrierscheibe aus dem amerikanischen Bürgerkrieg
  - Die ausschließliche Verwendung der Zeichen 1 und 8 als Symbole auf der äußeren Scheibe und die unterschiedlichen Symbollängen verbessern das Verfahren



Tafel des Polybius (auch Polybius-Chiffre genannt)

- Der griechische Schriftsteller Polybius (200 v.Chr.) erfand die nach ihm benannte **Tafel des Polybius**
- **Monographische** und **monoalphabetische** Verschlüsselung
- Er ordnete die Buchstaben in einem Quadrat an und nummerierte die Zeilen und Spalten
- Nimmt man das deutsche Alphabet und fasst `I` und `J` zusammen, bekommt man eine 5x5 Matrix
- Jeder Buchstabe wird durch eine zweistellige Zahl beschrieben
  - Koordinaten (Zeile, Spalte) sind z. B. B=12, R=42, U=45, Y=54
- Polybius schlug vor, die Zahlen mit Hilfe von Fackeln zu übermitteln (siehe Vorlesung 1)

	1	**2**	**3**	**4**	**5**
1	A	B	C	D	E
2	F	G	H	I/J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Playfair-Verschlüsselung

- **Monoalphabetische** und **bigraphische Substitution**
  - Klartext-Bigramme (Buchstabenpaare) werden in Geheimtext-Bigramme verschlüsselt
- Erfunden 1854 von Sir Charles Wheatstone und von Lord Lyon Playfair zur Benutzung beim britischen Militär empfohlen
  - War bis zum Ersten Weltkrieg in Gebrauch

Playfair-Verschlüsselung: Beispiel

- Beispiel
  - Nachricht: `VORLESUNG IST AM MITTWOCH`
  - Schlüssel: `CODIERUNG`
- Playfair-Quadrat (5x5) erstellen
  - Nur Großbuchstaben und „J“ fällt weg
  - Zuerst wird der Schlüssel eingetragen (jeder Buchstaben nur ein mal)
  - Danach wird mit den restlichen Buchstaben in alphabetischer Reihenfolge aufgefüllt

C	O	D	I	E
R	U	N	G	A
B	F	H	K	L
M	P	Q	S	T
V	W	X	Y	Z

- Klartext: `VORLESUNG IST AM MITTWOCH`
- Klartext in Bigrammen schreiben:
  `VO RL ES UN GI ST AM MI TX TW OC HX`
  - Umlaute werden aufgelöst und Leerzeichen sowie Satzzeichen weggelassen
  - Es dürfen keine Bigramme aus zwei identischen Buchstaben entstehen
    - Um dies zu vermeiden, wird gegebenenfalls ein „X“ eingefügt
  - Tritt am Ende des Textes ein einziger Buchstabe allein auf, wird er durch Anhängen eines weiteren „X“ zu einem Bigramm erweitert

Playfair-Verschlüsselung: Verschlüsselung (1/2)

- Grundlage der Verschlüsselung ist mit dem Kennwort erzeugte Playfair-Quadrat und der in Bigramme zerlegte Klartext
- Es werden immer Klartext-Bigramme in Geheimtext-Bigramme umgewandelt
  - Stehen beide Buchstaben in der gleichen Spalte, werden jeweils die unteren Nachbarbuchstaben als Geheimbuchstaben genommen
  - Stehen beide Buchstaben in der gleichen Zeile, werden jeweils die rechten Nachbarbuchstaben als Geheimbuchstaben genommen
  - Stehen die Buchstaben am Rand des Playfair-Quadrats stehen, wird einfach am anderen Rand fortgesetzt
    - Das Quadrat ist also links und rechts sowie oben und unten als verbunden anzunehmen
  - Stehen beide Buchstaben in unterschiedlichen Zeilen und Spalten, ersetzt man den ersten Klarbuchstaben durch den in derselben Zeile aber in der Spalte des zweiten liegenden
    - Der zweite Klarbuchstabe wird durch den in derselben Zeile aber in der Spalte des ersten Klarbuchstabens ersetzt

Playfair-Verschlüsselung: Verschlüsselung (2/2)

- Klartext: VO RL ES UN GI ST AM MI TX TW OC HX

C	**O**	D	I	E
R	U	N	G	A
B	F	H	K	L
M	P	Q	S	T
V	**W**	X	Y	Z

VO $\Longrightarrow$ WC

C	O	D	I	E
R	**U**	**N**	**G**	A
B	F	H	K	L
M	P	Q	S	T
V	W	X	Y	Z

UN $\Longrightarrow$ NG

C	O	D	I	E
R	U	N	G	**A**
B	F	H	K	L
M	P	Q	S	**T**
V	W	X	Y	Z

AM $\Longrightarrow$ RT

C	O	D	I	E
R	U	N	G	A
B	F	H	K	L
M	**P**	Q	S	**T**
V	**W**	X	Y	**Z**

TW $\Longrightarrow$ PZ

C	O	D	I	E
R	U	N	G	**A**
B	F	H	K	**L**
M	P	Q	S	T
V	W	X	Y	Z

RL $\Longrightarrow$ AB

C	O	D	**I**	E
R	U	N	**G**	A
B	F	H	**K**	L
M	P	Q	S	T
V	W	X	Y	Z

GI $\Longrightarrow$ KG

C	O	D	**I**	E
R	U	N	G	A
B	F	H	K	L
M	P	Q	**S**	T
V	W	X	Y	Z

MI $\Longrightarrow$ SC

C	**O**	**D**	I	E
R	U	N	G	A
B	F	H	K	L
M	P	Q	S	T
V	W	X	Y	Z

OC $\Longrightarrow$ DO

C	O	D	**I**	**E**
R	U	N	G	A
B	F	H	K	L
M	P	Q	**S**	**T**
V	W	X	Y	Z

ES $\Longrightarrow$ IT

C	O	D	I	E
R	U	N	G	A
B	F	H	K	L
M	P	Q	**S**	**T**
V	W	X	Y	Z

ST $\Longrightarrow$ TM

C	O	D	I	E
R	U	N	G	A
B	F	H	K	L
M	P	**Q**	S	**T**
V	W	**X**	Y	**Z**

TX $\Longrightarrow$ QZ

C	O	**D**	I	E
R	U	N	G	A
B	F	**H**	K	L
M	P	**Q**	S	T
V	W	**X**	Y	Z

HX $\Longrightarrow$ QD

- Geheimtext: WC AB IT NG KG TM RT SC QZ PZ DO QD

Playfair-Verschlüsselung: Entschlüsselung

- Die Entschlüsselung ist die Umkehrung der Verschlüsselung
- Wie der Sender erzeugt der Empfänger mit dem ihm bekannten Kennwort das identische Playfair-Quadrat
  - Stehen beide Buchstaben in der gleichen Spalte, werden jeweils die **obere** Nachbarbuchstaben als Klartextbuchstaben genommen
  - Stehen beide Buchstaben in der gleichen Zeile, werden jeweils die **linken** Nachbarbuchstaben als Klartextbuchstaben genommen
  - Das übrige Vorgehen ist zur Verschlüsselung identisch

Playfair-Verschlüsselung: Eigenschaften

- Die Entzifferung ist mit einer Häufigkeitsverteilung der Buchstabenpaare (Bigramme) möglich
- Da kein Buchstabe mit sich selbst gepaart wird, gibt es nur 600 $(25 * 24)$ mögliche Buchstabenkombinationen, die substituiert werden
- Da die Schlüsselwörter meistens kürzer sind als 25 Buchstaben, ist das Ende des Playfair-Quadrat bekannt

Nachteil monoalphabetischer Verschlüsselung

- Im Gegensatz zur Caesar-Verschlüsselung mit nur 25 Möglichkeiten gibt es sehr viele Möglichkeiten zur Verwürfelung des Standardalphabetes
  $\Longrightarrow$ Große Anzahl möglicher Schlüssel
  - Der erste Buchstabe A kann an eine von 26 mögliche Positionen im Alphabet platziert werden
  - Für den zweiten Buchstaben B gibt es dann noch 25 mögliche Plätze
  - Für den dritten Buchstaben C gibt es dann noch 24 mögliche Plätze
  - ...
- Insgesamt gibt es also $26 * 25 * 24 * 23 \ldots 4 * 3 * 2 * 1 = 26! \approx 4 * 10^{26}$ mögliche Schlüssel
  - Das sind ca. 88 Bit
  - Eine Entzifferung durch Ausprobieren aller Fälle (Brute-Force) ist praktisch unmöglich
- Trotz der großen Anzahl möglicher Schlüssel sind monoalphabetische Verschlüsselungen durch eine Häufigkeitsanalyse leicht zu knacken
  - Der Angreifer entschlüsselt die häufigsten Buchstaben und rät die restlichen

Polyalphabetische Verfahren

- Der Schwachpunkt der monoalphabetischen Verfahren ist, dass die Häufigkeit der Buchstaben erhalten bleibt
  - Besser wäre es so zu verschlüsseln, dass die Häufigkeiten der Buchstaben in der Chiffre möglichst gleich groß sind
- Lösung: Die Geheimtextalphabete wechseln $\Longrightarrow$ Polyalphabetische Verschlüsselung
- Bekannte polyalphabetische Chiffre: **Vigenère-Verschlüsselung**
  - Man verwendet wechselnde Alphabete
  - der Wechsel der Alphabete wird durch ein Schlüsselwort gesteuert

Polyalphabetische Chiffren – Beispiel

- Schlüsselwort: `PROSEMINAR`
- Klartext: `diesistgeheim`
- Mit der Verschlüsselungstabelle (nächste Folie) kann der Chiffretext generiert werden
  - Die Verschlüsselungstabelle wird auch **Vigenère-Quadrat** genannt
  - Das Schlüsselwort wird laufend wiederholt
- Vorgehen:
  - 1.Buchstaben verschlüsseln: In Zeile 15 und Spalte (`d`) nachschauen
  - 2.Buchstaben verschlüsseln: In Zeile 17 und Spalte (`i`) nachschauen
  - 3.Buchstaben verschlüsseln: In Zeile 14 und Spalte (`e`) nachschauen
  - usw.

Schlüsselwort	P 15	R 17	O 14	S 18	E 4	M 12	I 8	N 13	A 0	R 17	P 15	R 17	O 14
Klartext	d	i	e	s	i	s	t	g	e	h	e	i	m
Chiffre	S	Z	S	K	M	E	B	H	E	Y	T	Z	A

Vigenère-Verschlüsselung: Verschlüsselungstabelle

	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
0	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	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
2	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
3	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
4	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
5	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
6	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
7	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
8	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
9	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
10	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
11	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
12	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
13	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
14	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
15	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
16	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
17	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
18	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
19	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
20	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
21	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
22	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
23	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
24	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
25	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

Polyalphabetische Chiffren – Qualität

Schlüsselwort	P 15	R 17	O 14	S 18	E 4	M 12	I 8	N 13	A 0	R 17	P 15	R 17	O 14
Klartext	d	i	e	s	i	s	t	g	e	h	e	i	m
Chiffre	S	Z	S	K	M	E	B	H	E	Y	T	Z	A

- Diese Verschlüsselung hat eine höhere Sicherheit
  - Gleiche Buchstaben des Klartextes werden nun in verschiedene Geheimtextbuchstaben verschlüsselt
  - Gleiche Geheimtextbuchstaben haben ihren Ursprung in unterschiedlichen Klartextbuchstaben
- Die Vigenère-Verschlüsselung kam im 16.Jahrhundert auf und galt über 300 Jahre als sicher

Kryptoanalyse einer Vigenère-Chiffre

- Bei einer Kryptoanalyse einer Vigenère-Chiffre ist das erste und das wichtigste Ziel die Bestimmung der Schlüsselwortlänge k
- Ist k gefunden, funktioniert der Rest der Analyse wie bei einer gewöhnlichen Verschiebechiffre, wobei nun k Geheimtexte separat analysiert werden
- Es werden 2 Tests zur Bestimmung der Schlüsselwortlänge vorgestellt
  - Der **Kasiski-Test** liefert nicht den exakten Wert für die Schlüsselwortlänge, sondern ihn bis auf ein Vielfaches
  - Der **Friedmann-Test** liefert eine Abschätzung der Schlüsselwortlänge
- Eine Kombination der beiden Tests führt dann häufig zum Ziel

Der Kasiski-Test (1863) von Major Friedrich Kasiski

- Man untersucht den Geheimtext nach Wiederholungen von Zeichenfolgen mit mindestens 3 Zeichen und misst deren Abstand
  - Man zählt vom ersten Buchstaben der ersten Folge (einschließlich) bis zum ersten Buchstaben der zweiten Folge (ausschließlich)
  - So verfährt man mit allen Folgen und schreibt die Abstände auf
    - Man erhält eine Liste natürlicher Zahlen
  - Diese Zahlen werden in Primfaktoren zerlegt
    - Gleiche Teiler lassen sich somit schnell finden
- Je länger die Zeichenfolgen sind, desto größer ist die Wahrscheinlichkeit, dass der Abstand ein Vielfaches der Schlüsselwortlänge ist
  - Zufällig entstandene Übereinstimmungen sind leicht erkennbar, weil sie aus der Reihe fallen
- Die genaue Schlüssellänge wird nicht bekannt, denn der Kasiski-Test liefert nur Vielfache der Schlüssellänge
  - Zur genauen Betrachtung $\Longrightarrow$ Friedman-Test
- Für eine erfolgreiche Durchführung des Tests muss der verfügbare Geheimtext viel länger sein als die Schlüssellänge

Der Kasiski-Test – Beispiel von Wikipedia

- Länge des Schlüsselwortes mit dem Kasiski-Test bestimmen

```
AXTRX TRYLC TYSZO EMLAF QWEUZ HRKDP NRVWM WXRPI
JTRHN IKMYF WLQIE NNOXW OTVXB NEXRK AFYHW KXAXF
QYAWD PKKWB WLZOF XRLSN AAWUX WTURH RFWLL WWKYF
WGAXG LPCTG ZXWOX RPIYB CSMYF WIKPA DHYBC SMYFW
KGMTE EUWAD LHSLP AVHFK HMWLK
```

```
XTR:      Abstand 3
XRPI:     Abstand 98
YFW:      Abstand 70
YBCSMYFW: Abstand 14
```

- Abstände in Primfaktoren zerlegen

```
 3  =     3
98  = 2 x       7 x 7
70  = 2 x   5 x 7
14  = 2 x       7
```

- Alle Abstände (außer dem ersten) sind Vielfache von 14
  - Der Abstand 3 ist vermutlich ein zufälliges Zusammentreffen
- Vermutung: Die Schlüsselwortlänge ist 2, 7 oder 14
  - Länge 2 kann ausgeschlossen werden
  - Ergebnis: Die Schlüsselwortlänge ist 7 oder 14

Der Friedmann-Test

- 1925 vom amerikanischen Kryptologen William Friedmann erfunden
- Es wird der sogenannte **Koinzidenzindex** K berechnet
  - Die Wahrscheinlichkeit, dass zwei Buchstaben an gleicher Blockposition gleich sind, ist (empirisch) für längere deutsche Texte $\mu \approx 0,0762$
  - Die Wahrscheinlichkeit, dass zwei Buchstaben an verschiedenen Blockpositionen gleich sind, ist $\varphi \approx 0,0385$

$$K = \frac{(m-n) * \mu + m(n-1) * \varphi}{n(m-1)}$$

- Löst man nach n auf, ergibt sich:

$$n = \frac{m(\mu - \varphi)}{K(m-1) + \mu - m * \varphi}$$

- Die vermutete Schlüssellänge des Codewortes ist eine ganze Zahl, die in der Nähe dieser Schätzung liegt

One-Time-Pad (Einmalverschlüsselung)

- Beispiel für **polyalphabetische Verschlüsselung**
- Wurde 1918 von Joseph Mauborgne und Gilbert Vernamn erfunden
- Soll eine Entschlüsselung durch Häufigkeitsanalyse verhindert werden, darf ein Schlüssel nur für eine Nachricht verwendet werden und muss mindestens so lang wie die Nachricht selbst sein
- Vorteil: Kann nicht entziffert werden, wenn der Schlüssel aus Zeichen besteht, die zufällig und unabhängig sind, und wenn er nur einmal zur Verschlüsselung verwendet wird
- Nachteil: Schlüssel müssen auch irgendwie zum Empfänger gelangen
- In der Praxis wird heute binär verschlüsselt

Klartext	0	0	1	0	1	1	0	1	1
Schlüssel (zufällig)	0	1	0	1	0	1	0	1	0
Geheimtext	0	1	1	1	1	0	0	0	1
Schlüssel (zufällig)	0	1	0	1	0	1	0	1	0
Klartext	0	0	1	0	1	1	0	1	1

ENIGMA

- Weiteres Beispiel einer **polyalphabetischen Verschlüsselung**
- Der deutsche Erfinder Arthur Scherbius entwickelte 1918 die Chiffriermaschine ENIGMA
  - Die ENIGMA war zunächst als ziviles Chiffriersystem konzipiert und wurde zum Kauf angeboten
  - Gegen Ende der 1920er Jahre zeigten militärische Stellen Interesse und die Maschine verschwand vom zivilen Markt
- Mit drei Walzen und einer Steckbrettverschaltung ist die Enigma eine angepasste Vigenère-Chiffre mit der maximalen Schlüssellänge $26 * 26 * 26 = 17576$

Bestandteile der ENIGMA (1/5)

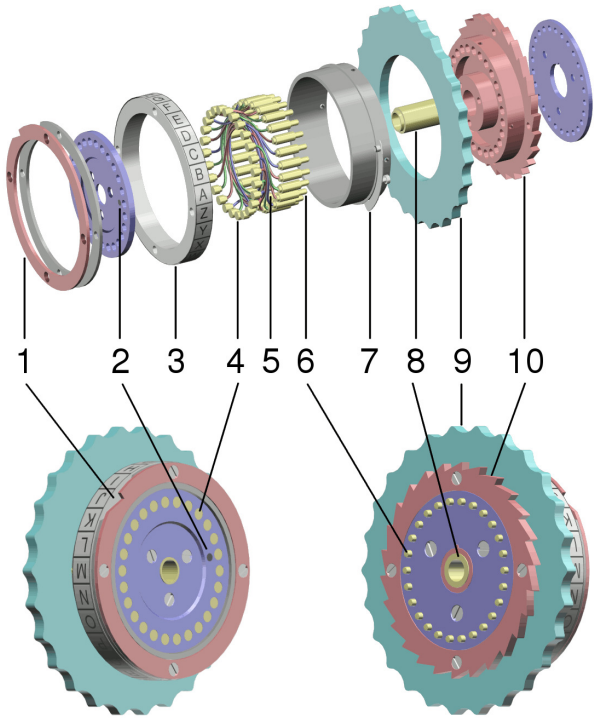
Bildquelle: Wikipedia

- **Tastatur**
  - Genau so wie die einer QUERTZ-Schreibmaschine
  - Hier wird der Klartext eingegeben
- **Lampenfeld**
  - Alle Buchstaben des Alphabets sind als kleines Lämpchen angeordnet
  - Sieht fast genau so aus wie die Tastatur (gleiches Layout)
  - Wird ein Buchstabe zum Verschlüsseln in die Enigma eingegeben, leuchtete der verschlüsselte Buchstabe im Lampenfeld auf

Bestandteile der ENIGMA (2/5)

Bildquelle: Wikipedia

- **Walzen (Rotoren)**
  - Vollziehen die Umwandlung des Klartextes
  - Drei Walzen, auf denen sich entsprechend 26 Buchstaben 26 Verdrahtungen (Nummer 5 im Bild) befinden
  - Die Drähte verlaufen unregelmäßig quer über den Rotor
    - Verhindert eine Gleichmäßigkeit in der Verschlüsselung
  - Die Verdrahtungen in den Rotoren sind auch untschiedlich
  - Jede Walzen stellt eine Permutation des Alphabets (**monoalphabetische Verschlüsselung**) dar
  - Die Walzen sind miteinander gekoppelt
    - Bei jeder Eingabe eines Buchstabens drehen die Walzen um eine Einheit weiter



Bestandteile der ENIGMA (3/5)

- Die Walzen sind hintereinander angereiht
  - D.h. das die erste Walze mit den Drähten der einzelnen Buchstaben zusammentrifft und die letzte mit dem Lampenfeld verbunden ist
- Walzen ändern durch Drehen nach jeder neuen Eingabe eines Buchstabens ihre Lage zueinander
  - Somit verschieben sich die Drahtverbindungen zwischen ihnen und ein neues Geheimtextalphabet wird verwendet (**polyalphabetische Verschlüsselung**)
  - Außerdem ist es möglich, die Walzen untereinander auszutauschen, was (bei 3 Walzen) die Anzahl der Verschlüsselungsmöglichkeiten um den Faktor 6 erhöht
- Um die Sicherheit zu verbessern, kamen immer neue Walzen hinzu
  - Am Ende des 2.Weltkrieges waren es bis zu 10 Walzen
  - Aus diesen 10 Walzen mussten mit einem Codebuch die richtigen ausgewählt werden

Verdrahtungsschema der ENIGMA

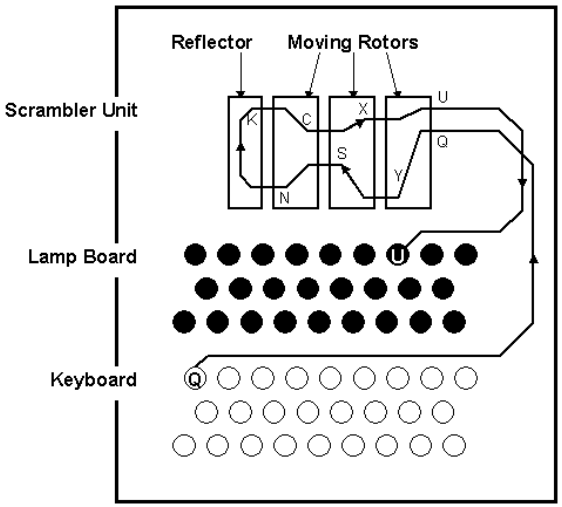
- Die Tabelle zeigt das damals geheime Verdrahtungsschema der bei der ENIGMA I verfügbaren 5 drehbaren Walzen I bis V

	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
I	E	K	M	F	L	G	D	Q	V	Z	N	T	O	W	Y	H	X	U	S	P	A	I	B	R	C	J
II	A	J	D	K	S	I	R	U	X	B	L	H	W	T	M	C	Q	G	Z	N	P	Y	F	V	O	E
III	B	D	F	H	J	L	C	P	R	T	X	V	Z	N	Y	E	I	W	G	A	K	M	U	S	Q	O
IV	E	S	O	V	P	Z	J	A	Y	Q	U	I	R	H	X	L	N	F	T	G	K	D	C	M	W	B
V	V	Z	B	R	G	I	T	Y	U	P	S	D	N	H	L	X	A	W	M	J	Q	O	F	E	C	K

Bestandteile der ENIGMA (4/5)

Bildquelle: http://www.sebastianlueth.de

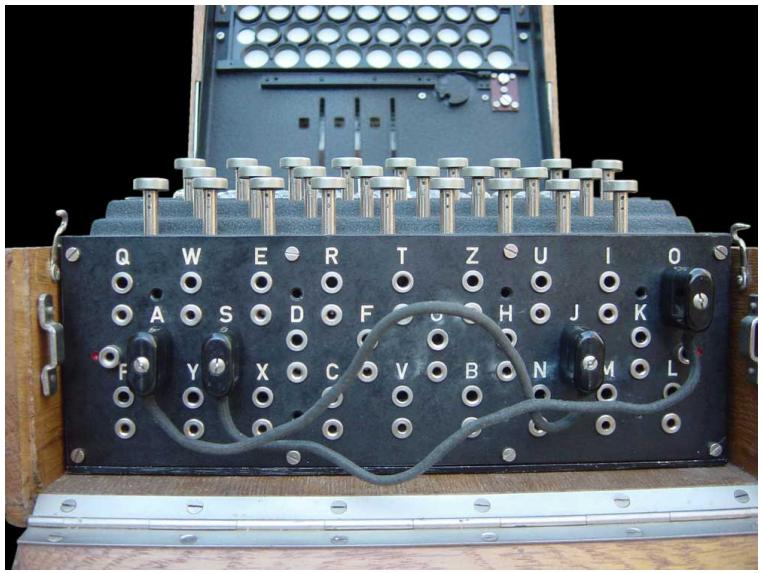
- **Umkehrwalze (Reflektor)**
  - Links vom Walzensatz ist die feststehende Umkehrwalze (UKW)
  - Die UKW hat nur auf ihrer rechten Seite 26 Kontakte, die paarweise miteinander verbunden sind
    - UKW A (bis 1937), UKW B (ab 1937), UKW C (1940 und 1941)
- Der Strom durchfließt den Walzensatz von rechts nach links, wird umgelenkt und durchfließt ihn noch einmal von links nach rechts
  - Der Strom verlässt den Walzensatz, wie er gekommen ist, wieder über die Eintrittswalze
- Die Walzensatz machte die Chiffre umkehrbar
  - Verschlüsselt a nach b und der Empfänger tippt b bei der gleichen Einstellung der Walzen, so erhält er wieder a



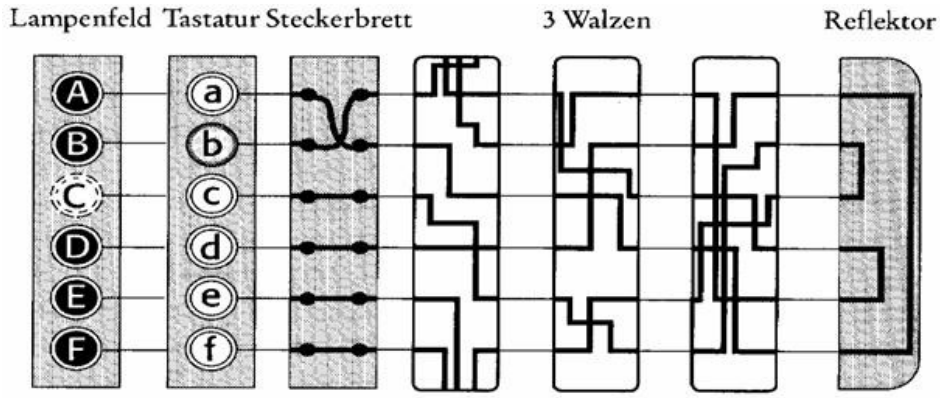
UKW A	AE	BJ	CM	DZ	FL	GY	HX	IV	KW	NR	OQ	PU	ST
UKW B	AY	BR	CU	DH	EQ	FS	GL	IP	JX	KN	MO	TZ	VW
UKW C	AF	BV	CP	DJ	EI	GO	HY	KR	LZ	MX	NW	QT	SU

Bestandteile der ENIGMA (5/5)

- **Steckfeld**
  - Vertauscht vor und nach der Benutzung Buchstabenpaare
  - Dafür verbindet man zwei Buchstaben auf dem Steckfeld
    - Bis zu 13 Verbindungen sind möglich
  - Nicht gesteckten Buchstaben werden ganz normal durch die Walzen verschlüsselt



Bildquelle: Wikipedia



Bildquelle: `http://www.clubkom.org`

- Im Bild ist `A` mit `J` und `S` mit `O` über ein Kabel verbunden
  - Gibt man `A` ein, durchläuft nicht `A` die Walzen, sondern `J`
  - Gibt man `S` ein, durchläuft nicht `S` die Walzen, sondern `O`

Arbeit mit der ENIGMA (1/2) – Quelle: Wikipedia

- Ab 1939 standen fünf Walzen (I, II, III, IV und V) zur Verfügung
- Der Benutzer wählte nach Vorgabe der geheimen *Schlüsseltafel* jeden Tag 3 der 5 Walzen aus und setzte diese nach der im Tagesschlüssel unter der Überschrift *Walzenlage* vorgeschriebenen Anordnung ein
  - Die Schlüsseltafel enthält tabellarisch für einen kompletten Monat die Tagesschlüssel, die um Mitternacht gewechselt wurden
- Im Beispiel sind nur 3 Monatstage dargestellt
  - Die Tage sind absteigend sortiert sind, damit der Benutzer verbrauchte Codes der vergangenen Tage abzuschneiden und vernichten kann

```
Tag UKW  Walzenlage  Ringstellung  ---- Steckerverbindungen ----
 31  B   I   IV III    16 26 08    AD CN ET FL GI JV KZ PU QY WX
 30  B   II  V  I      18 24 11    BN DZ EP FX GT HW IY OU QV RS
 29  B   III I  IV     01 17 22    AH BL CX DI ER FK GU NP OQ TY
```

Arbeit mit der ENIGMA (2/2) – Quelle: Wikipedia

Tag	UKW	Walzenlage	Ringstellung	---- Steckerverbindungen ----
31	B	I IV III	16 26 08	AD CN ET FL GI JV KZ PU QY WX
30	B	II V I	18 24 11	BN DZ EP FX GT HW IY OU QV RS
29	B	III I IV	01 17 22	AH BL CX DI ER FK GU NP OQ TY

- Beispiel für den 31.Tag des Monats:
  - `UKW B` bedeutet, dass als Umkehrwalze die Walze `B` zu wählen ist
  - Walzenlage `I IV III` bedeutet, dass und Walze `I` links, Walze `IV` in der Mitte und Walze `III` rechts einzusetzen ist
  - Die Ringe, die außen am Walzenkörper angebracht sind und den Versatz zwischen der internen Verdrahtung der Walzen und dem Buchstaben bestimmen, zu dem der Übertrag auf die nächste Walze erfolgt, sind auf den 16., 26. beziehungsweise 8.Buchstaben des Alphabets einzustellen, also auf `P`, `Z` und `H`

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
01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26

Schlüsselraum der ENIGMA (1/3)

- **Walzenlage**
  - 3 von 5 Walzen (I bis V) und eine von zwei Umkehrwalzen (B oder C) werden ausgewählt
    - Das ergibt $2 * (5 * 4 * 3) = 120$ mögliche Walzenlagen
- **Ringstellung**
  - 26 Ringstellungen (01 bis 26) für die mittlere und die rechte Walze
  - Ring der linken Walze ist kryptographisch bedeutungslos
    - Ihre Übertragskerbe bewirkt kein Fortschalten einer weiteren Walze
  - Insgesamt sind $26^2 = 676$ Ringstellungen relevant
- **Walzenstellung**
  - Die Walzen haben $26^3 = 17.576$ unterschiedliche Einstellungen
    - 676 Anfangsstellungen sind als kryptographisch redundant zu eliminieren
    - Als relevant übrig bleiben 16.900 Walzenstellungen
- **Steckerverbindungen**
  - Maximal 13 Steckerverbindungen zwischen den 26 Buchstaben
  - Ab 1939 wurden immer genau 10 Steckerverbindungen platziert
    - Das sind 150.738.274.937.250 Steckermöglichkeiten

Schlüsselraum der ENIGMA (2/3)

- 120 Walzenlagen $*$ 676 Ringstellungen $*$ 16.900 Walzenstellungen $*$ 150.738.274.937.250 Steckermöglichkeiten = 206.651.321.783.174.268.000.000
  - Das sind ca. $2 * 10^{23}$ Möglichkeiten
  - Das entspricht einer Schlüssellänge von ca. 77 Bit
- Solange die Maschine die gleiche Einstellung hat, wird kein Buchstabe auf sich selbst abgebildet und der gleiche Geheimtextbuchstabe mit dem gleichen Klartextbuchstaben verschlüsselt
  - Diese Fähigkeit wird als **selbstinverse Abbildung** bezeichnet
  - Aus diesem Grund kann die Maschine in der bekannten Anfangseinstellung ohne Veränderung zum Dechiffrieren benutzt werden

Schlüsselraum der ENIGMA (3/3)

- Die kryptographisch stärkste ENIGMA-Variante ist die M4
  - Sie verwendet vier Walzen (außer der Eintritts- und der Umkehrwalze)
    - Die *Griechenwalze* und die UKW sind unbeweglich
    - Das ergibt $2 * 2 * (8 * 7 * 6) = 1.344$ mögliche Walzenlagen
  - Die Walzen haben $26^4 = 456.976$ unterschiedliche Einstellungen
    - 17.576 Anfangsstellungen sind als kryptographisch redundant zu eliminieren
    - Als relevant übrig bleiben 439.400 Walzenstellungen
- Schlüsselraum der ENIGMA-M4 ergibt sich aus 3 ausgewählten Walzen (I bis VIII), einer von zwei nichtrotierenden *Griechenwalzen* (Beta und Gamma) und einer von zwei Umkehrwalzen sowie bei Verwendung von zehn Steckern
  - 1.344 Walzenlagen $*$ 676 Ringstellungen $*$ 439.400 Walzenstellungen $*$ 150.738.274.937.250 Steckermöglichkeiten = 60.176.864.903.260.346.841.600.000
    - Das sind ca. $6 * 10^{25}$ Möglichkeiten
    - Das entspricht einer Schlüssellänge von ca. 86 Bit

Gründe, warum die ENIGMA entziffert werden konnte

- Die Maschine baute auf einem käuflichen Chiffriergerät auf
- Die Einstellungen wurden über das Heeresschlüsselbuch festgelegt und waren 24 (später 8) Stunden für alle verschlüsselten Nachrichten gleich
- Menschliches Versagen: Es wurden auch Wetterberichte verschlüsselt, in fast jedem Funkspruch kamen die gleichen Wörter wie *Vaterland* vor
- Ein deutscher Funker tippte z.B. aus Langeweile immer wieder den gleichen Buchstaben auf der ENIGMA
  - Daraus konnten die Experten die Einstellung der Walzen erkennen
- Schwäche des Algorithmus: Kein Buchstabe wird auf sich selbst abgebildet und die gesamte Abbildung ist selbstinvers, was die Anzahl der Möglichkeiten einschränkt
  - Selbstinvers = Algorithmus zum Verschlüsseln und zum Entschlüsseln ist identisch

Entzifferung der ENIGMA

- **1928** Das polnische Militär gründete eine eigene Abteilung, die sich mit der Entschlüsselung der Enigma beschäftigte
- **1932** gelang es einer polnischen Mathematikergruppe um Marian Rejewski erstmals in das Enigma-System einzubrechen
- Die polnische Mathematikergruppe entwickelte die **Bomba**
  - Der Kern der Bomba bestand aus 6 polnischen Enigmas (Nachbauten)
  - Innerhalb 2 Stunden konnte eine Bomba alle 17.576 Permutationen (Walzeneinstellungen) testen und den Tagesschlüssel finden
- **1938** kamen bei den Deutschen zwei austauschbare Walzen hinzu
  - Das erhöhte die Anzahl der möglichen Walzenlagen von sechs auf sechzig
  - Damit war das Verfahren wieder sicher
- **1939** übergaben die Polen ihren Verbündeten alle Unterlagen
- Britischen Kryptoanalytiker um Alan Turing entwickelten in Bletchley Park die **Turing-Bombe**, die auf der polnischen Bomba aufbaute
- **Ab 1940** konnten alle mit der ENIGMA I verschlüsselten Nachrichten entschlüsselt werden

Stromchiffren und Blockchiffren

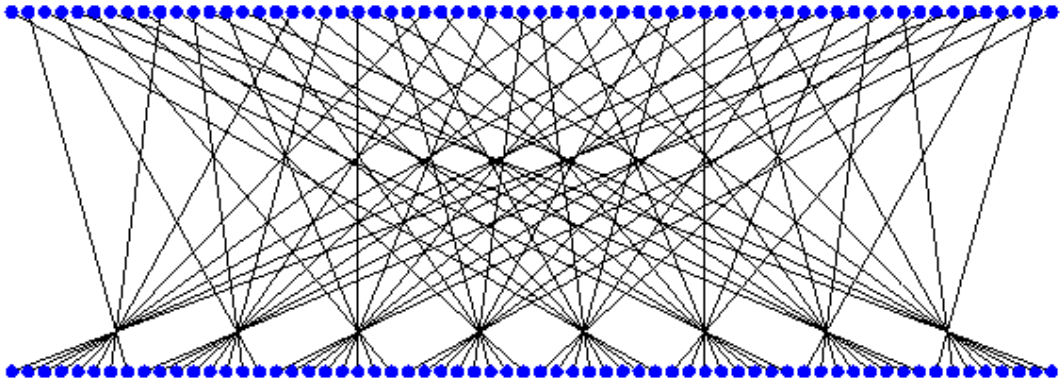
- Symmetrische Verschlüsselungsverfahren kann man in 2 Klassen einteilen
  - **Stromchiffren**
    - Es wird ein Zeichen nach dem anderen verschlüsselt
    - Sobald zu verschlüsselte Daten vorliegen, kann verschlüsselt werden
    - Besonders für Echtzeitübertragungen geeignet (z.B. Mobilfunk)
    - Beispiele: HC-256, PANAMA, RC4 und RABBIT
  - **Blockchiffren**
    - Die Nachricht wird in Blöcke zerteilt
    - Danach wird ein Block nach dem anderen verschlüsselt
    - Vor dem Verschlüsseln müssen sich erst genug zu verschlüsselnde Daten angesammelt haben, bis sie die Größe für einen Eingabeblock erreicht haben
    - Beispiele: AES, Blowfish, DES, 3DES, RC2 und RC6

Data Encryption Standart (DES)

- Von IBM entwickelt und 1975 veröffentlicht
- Seit 1976 offizieller Standard für alle US-Bundesbehörden
- Kommerziell am häufigsten eingesetzte Verschlüsselungsalgorithmus
- Die verwendeten Schlüssellänge von nur 56 Bit ist heute nicht mehr ausreichend sicher
  - Während der Entwicklung überzeugte die NSA IBM davon, dass eine reduzierte Schlüssellänge von 56 Bit statt 128 Bit ausreichend sei
- 64 Bit Blocklänge
  - Der Klartext wird in Blöcke von je 64 Bit unterteilt
  - Jeder 64-Bit-Block Klartext wird in einen 64-Bit-Block Chiffretext transformiert
- 64 Bit Schlüssellänge
  - Davon 8 Bit für den Paritäts-Check $\Longrightarrow$ 56 Bit effektive Schlüssellänge
  - der Schlüsselraum ist $2^{56} = 72.057.594.037.927.936$

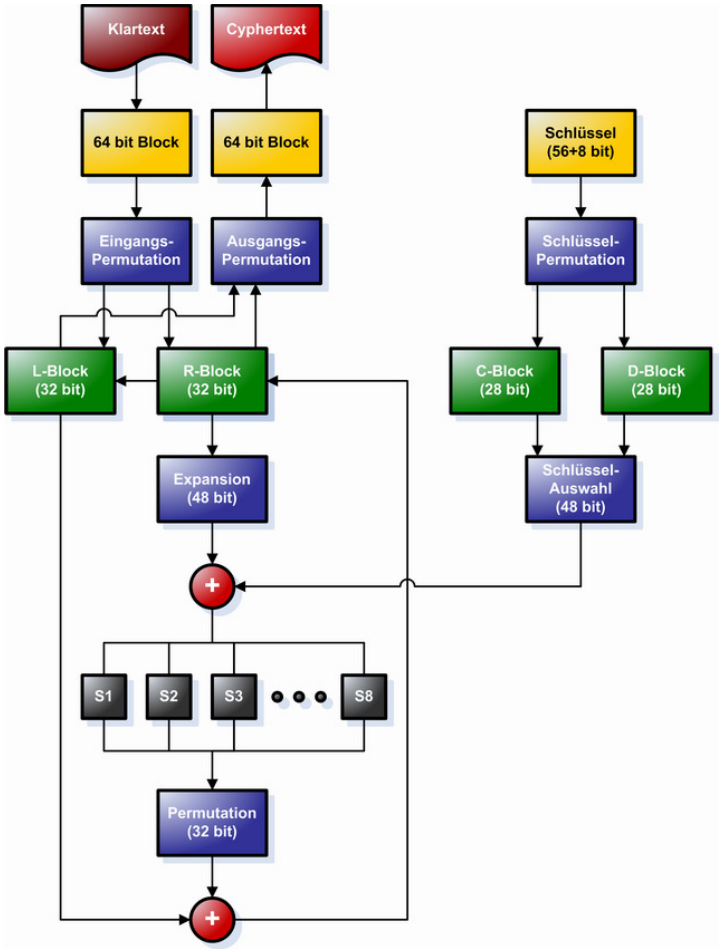
Arbeitsweise von DES (1/3)

- Eingangspermutation mischt den Klartext



Bildquelle: `http://me-lrt.de`

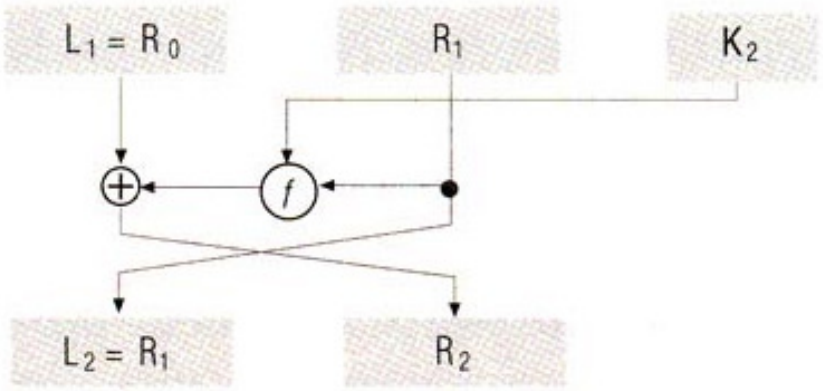
- Zerlegen jedes Eingabeblocks in eine linke und eine rechte Hälfte (je 32 Bit Länge)
- 16 Chiffrierrunden (Iterationen)
  - Bei jeder Iteration wird auf die rechte Hälfte die **Feistel-Funktion** angewandt



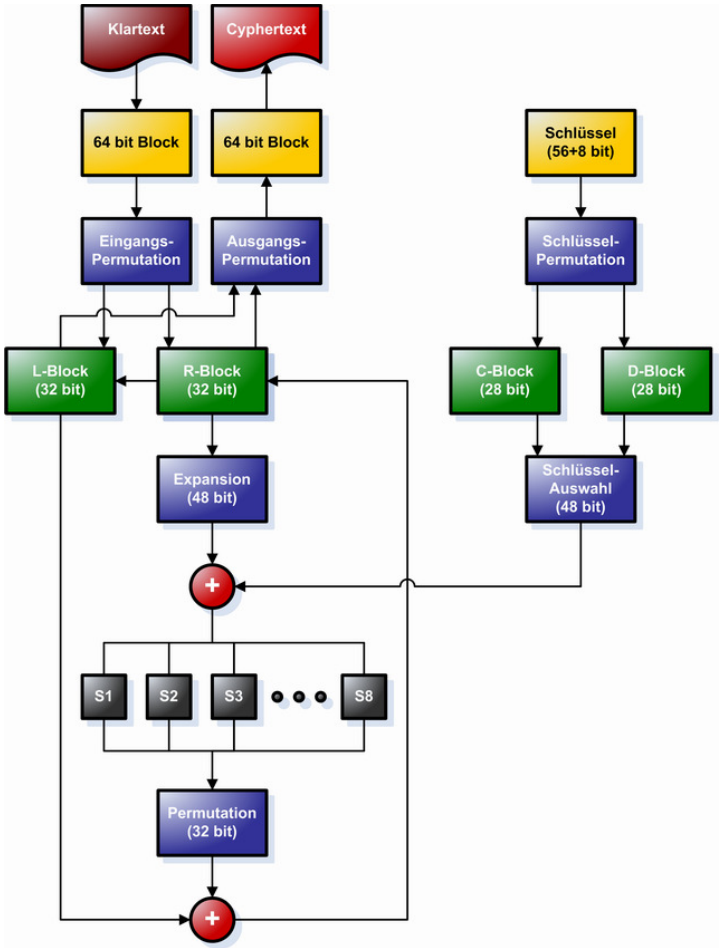
Bildquelle: Wikipedia

Arbeitsweise von DES (2/3)

- Am Ende jeder Iteration wird die rechte mit der linken Hälfte XOR-verknüpft und das Ergebnis im Register der rechte Hälfte für die nächsten Runde gespeichert
- Die rechte Blockhälfte wird in das linke Register der nächsten Runde kopiert
- Die Abbildung zeigt eine DES-Runde



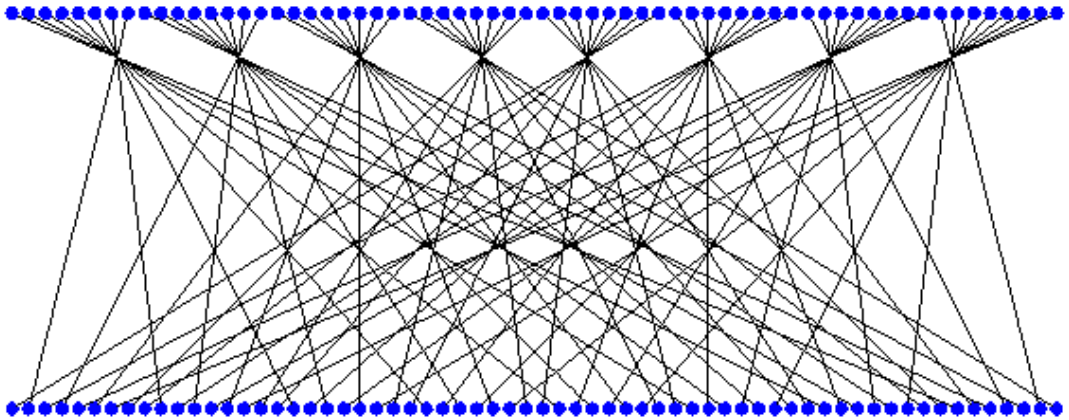
Bildquelle: `http://www-lehre.informatik.uni-osnabrueck.de`



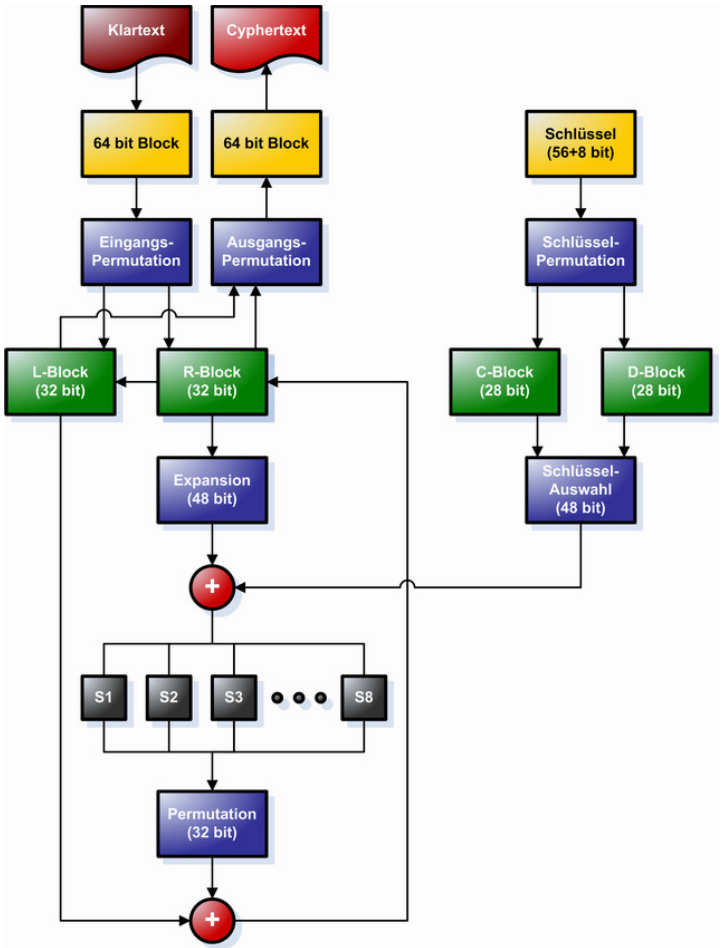
Bildquelle: Wikipedia

Arbeitsweise von DES (3/3)

- Nach dem Ende dem der letzten Runde werden die beiden Hälften wieder zusammengeführt und es folgt die Ausgangspermutation
  - Die Ausgangspermutation ist invers zur Eingangspermutation
  - Eingangs- und Ausgangspermutation tragen nicht zur Sicherheit bei



Bildquelle: `http://me-lrt.de`

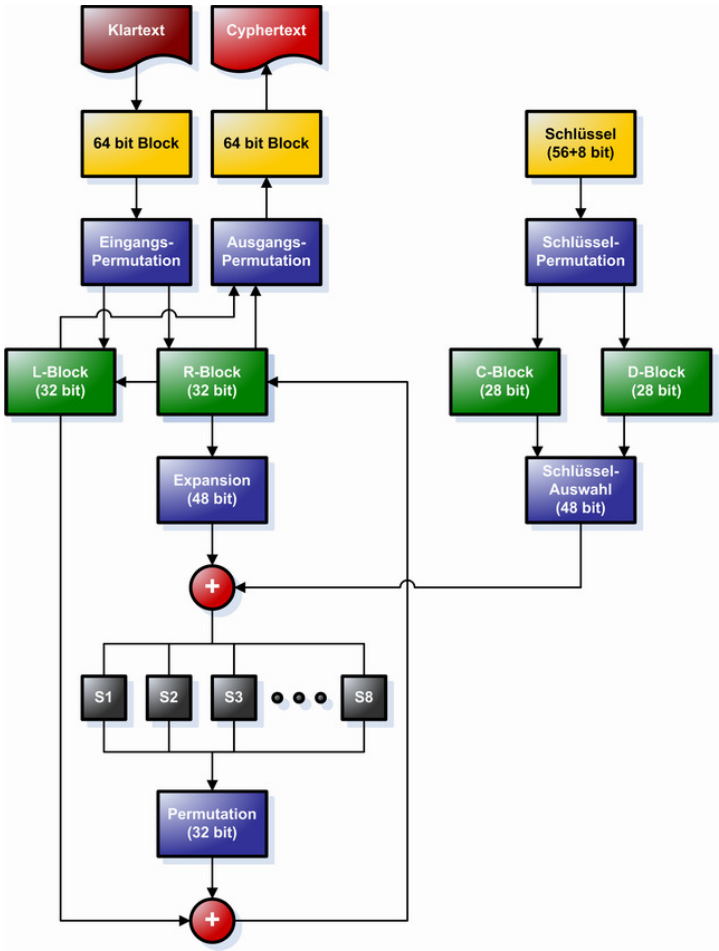


Bildquelle: Wikipedia

Die Feistel-Funktion von DES

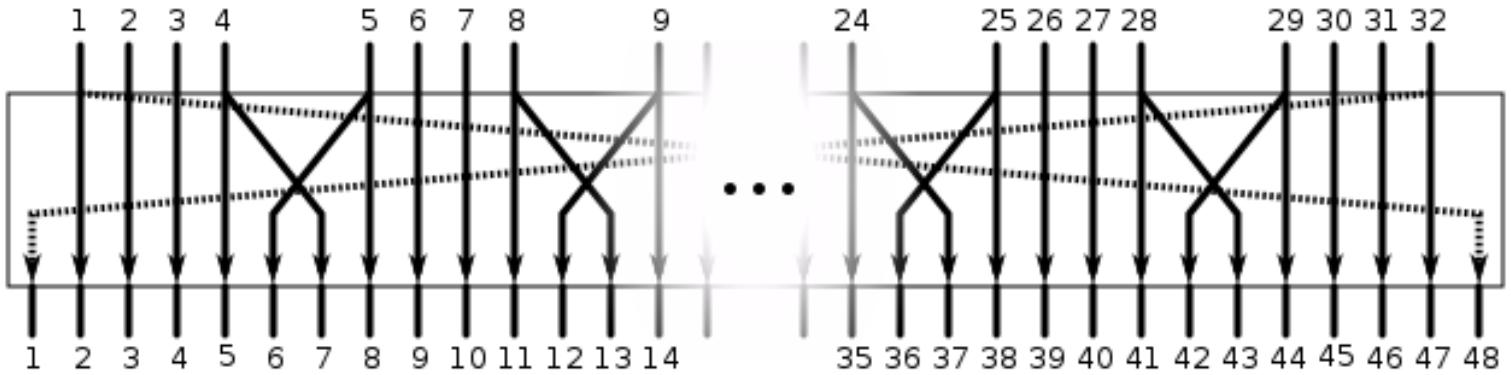
Bildquelle: Wikipedia

- Die Feistel-Funktion besteht aus 4 Phasen
  1. Erweiterung (**Expansion**) von 32 Bit auf 48 Bit
     - Einzelne Bits werden mehrfach verwendet
  2. Ergebnis mit einem Teilschlüssel der Runde XOR-verknüpfen
     - Für jede der 16 Runden wird hierzu ein anderer 48-Bit Teilschlüssel aus dem Hauptschlüssel erzeugt
  3. Ergebnis in acht 6-Bit-Stücke zerteilen
     - Jedes 6-Bit-Stücke mit **Substitution durch S-Boxen** (Substitutionsboxen) auf 4 Bit komprimieren
  4. Die Ausgabe (32 Bit) der S-Boxen mit einer festen Permutation (P-Box-Permutation) rearrangieren



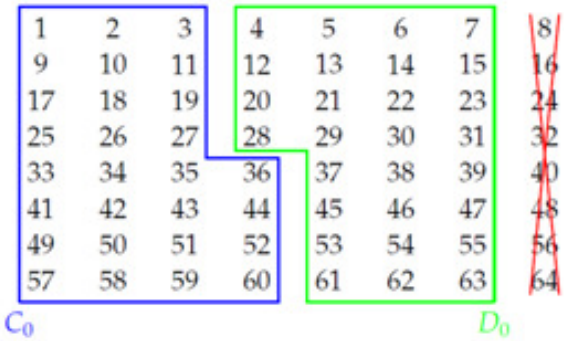
Die Expansion der Feistel-Funktion von DES

- Erweiterung eines Halbblocks von 32 Bit auf 48 Bit
  - Halbblock in 4-Bit-Gruppen aufteilen
  - Duplikate der Bits am Rand jeder 4-Bit-Gruppe werden vorn, beziehungsweise hinten an die benachbarte 4-Bit-Gruppe angehängt



Teilschlüssel/Rundenschlüssel bei DES berechnen (1/2)

- DES verwendet einen 64-Bit-Schlüssel
  - Von diesem sind nur 56 Bit nutzbar, weil jedes achte Bit ein Paritätsbit ist
- Zunächst werden mit der **Permuted-Choice-Funktion (PC-1)** die Paritätsbits entfernt und der Schlüssel in zwei 28-Bit-Teilschlüssel C_0 und D_0 aufgeteilt
- Beide Blöcke werden jeweils von unten nach oben durchlaufen
  - C_0 wird von links nach rechts durchlaufen
  - D_0 wird von rechts nach links durchlaufen
- Nun liegt der 56-Bit-Schlüssel vor



Bildquelle: `http://me-lrt.de`

Bildquelle: `http://www-lehre.informatik.uni-osnabrueck.de`

PC1						
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

Teilschlüssel/Rundenschlüssel bei DES berechnen (2/2)

- Nun werden die 16 Rundenschlüssel K_1 bis K_{16} erzeugt
- Diese bestehen aus den 16 Rundenteilschlüssel C_i und D_i
- Hierzu werden die Teilschlüssel C_0 und D_0 in jeder Runde um 1 oder 2 Bit zyklisch nach links geshiftet

PC1						
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

PC2					
14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

- Jeder Rundenteilschlüssel C_i und D_i ist Ergebnis des Linksshifts des vorherigen Teilschlüssels C_{i-1} und D_{i-1}
  - In den Runden 1, 2, 9 und 16 wird um 1 Bit linksgeshiftet
  - In den übrigen Runden wird je um 2 Bit linksgeshiftet
- Aus den Rundenteilschlüsseln C_i und D_i wird mit der **Permuted-Choice-Funktion (PC-2)** der jeweilige Rundenschlüssel K_i erzeugt

Die Substitution der Feistel-Funktion von DES

- Die 48 Bit werden zuerst in Blöcke a 6 Bit aufgeteilt
- Jede der 8 S-Boxen nimmt 6 Bit Eingabe entgegen und gibt 4 Bit Ausgabe zurück
  - Die 8 Substitutionsboxen (S-Boxen) bei DES sind standardisiert
- Bit 1 und 6 eines Blocks zusammen ergibt die Zeile in der zugehörigen S-Box
- Die von Bit 2 bis 5 dargestellte Zahl wird zur Auswahl der Spalte benutzt
- Der an dieser Stelle in der S-Box gefundene Eintrag ist eine Zahl zwischen 0 und 15, die als 4-Bit Binärzahl ausgegeben wird
  - Das Ergebnis ist $8 * 4$ Bit, insgesamt also 32 Bit an Daten
    - Danach folgt die P-Box-Permutation

$1\,|\,1010\,|\,0$

$(10)_2 = 2_{10} \Rightarrow$ die 3.Zeile

$(1010)_2 = 10_{10}$

Substitutionsboxen (S-Boxen) 1-4 bei DES

S_1		Mittlere 4 Bits des Eingabewertes															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Äußere Bits	**00**	1110	0100	1101	0001	0010	1111	1011	1000	0011	1010	0110	1100	0101	1001	0000	0111
	01	0000	1111	0111	0100	1110	0010	1101	0001	1010	0110	1100	1011	1001	0101	0011	1000
	10	0100	0001	1110	1000	1101	0110	0010	1011	1111	1100	1001	0111	0011	1010	0101	0000
	11	1111	1100	1000	0010	0100	1001	0001	0111	0101	1011	0011	1110	1010	0000	0110	1101

S_2		Mittlere 4 Bits des Eingabewertes															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Äußere Bits	**00**	1111	0001	1000	1110	0110	1011	0011	0100	1001	0111	0010	1101	1100	0000	0101	1010
	01	0011	1101	0100	0111	1111	0010	1000	1110	1100	0000	0001	1010	0110	1001	1011	0101
	10	0000	1110	0111	1011	1010	0100	1101	0001	0101	1000	1100	0110	1001	0011	0010	1111
	11	1101	1000	1010	0001	0011	1111	0100	0010	1011	0110	0111	1100	0000	0101	1110	1001

S_3		Mittlere 4 Bits des Eingabewertes															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Äußere Bits	**00**	1010	0000	1001	1110	0110	0011	1111	0101	0001	1101	1100	0111	1011	0100	0010	1000
	01	1101	0111	0000	1001	0011	0100	0110	1010	0010	1000	0101	1110	1100	1011	1111	0001
	10	1101	0110	0100	1001	1000	1111	0011	0000	1011	0001	0010	1100	0101	1010	1110	0111
	11	0001	1010	1101	0000	0110	1001	1000	0111	0100	1111	1110	0011	1011	0101	0010	1100

S_4		Mittlere 4 Bits des Eingabewertes															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Äußere Bits	**00**	0111	1101	1110	0011	0000	0110	1001	1010	0001	0010	1000	0101	1011	1100	0100	1111
	01	1101	1000	1011	0101	0110	1111	0000	0011	0100	0111	0010	1100	0001	1010	1110	1001
	10	1010	0110	1001	0000	1100	1011	0111	1101	1111	0001	0011	1110	0101	0010	1000	0100
	11	0011	1111	0000	0110	1010	0001	1101	1000	1001	0100	0101	1011	1100	0111	0010	1110

Substitutionsboxen (S-Boxen) 5-8 bei DES

S_5		Mittlere 4 Bits des Eingabewertes															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Äußere Bits	**00**	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

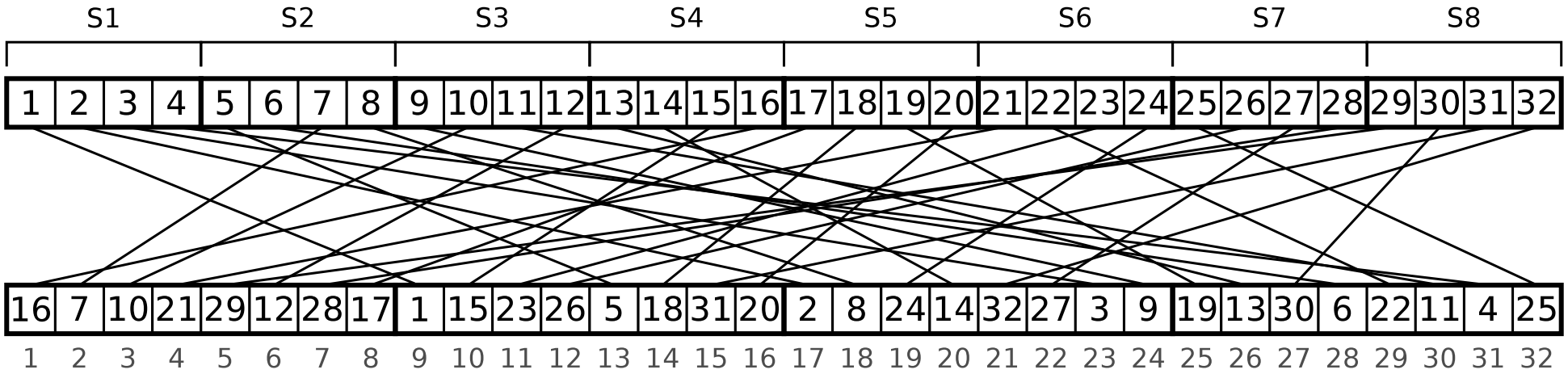
S_6		Mittlere 4 Bits des Eingabewertes															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Äußere Bits	**00**	1100	0001	1010	1111	1001	0010	0110	1000	0000	1101	0011	0100	1110	0111	0101	1011
	01	1010	1111	0100	0010	0111	1100	1001	0101	0110	0001	1101	1110	0000	1011	0011	1000
	10	1001	1110	1111	0101	0010	1000	1100	0011	0111	0000	0100	1010	0001	1101	1011	0110
	11	0100	0011	0010	1100	1001	0101	1111	1010	1011	1110	0001	0111	0110	0000	1000	1101

S_7		Mittlere 4 Bits des Eingabewertes															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Äußere Bits	**00**	0100	1011	0010	1110	1111	0000	1000	1101	0011	1100	1001	0111	0101	1010	0110	0001
	01	1101	0000	1011	0111	0100	1001	0001	1010	1110	0011	0101	1100	0010	1111	1000	0110
	10	0001	0100	1011	1101	1100	0011	0111	1110	1010	1111	0110	1000	0000	0101	1001	0010
	11	0110	1011	1101	1000	0001	0100	1010	0111	1001	0101	0000	1111	1110	0010	0011	1100

S_8		Mittlere 4 Bits des Eingabewertes															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Äußere Bits	**00**	1101	0010	1000	0100	0110	1111	1011	0001	1010	1001	0011	1110	0101	0000	1100	0111
	01	0001	1111	1101	1000	1010	0011	0111	0100	1100	0101	0110	1011	0000	1110	1001	0010
	10	0111	1011	0100	0001	1001	1100	1110	0010	0000	0110	1010	1101	1111	0011	0101	1000
	11	0010	0001	1110	0111	0100	1010	1000	1101	1111	1100	1001	0000	0011	0101	0110	1011

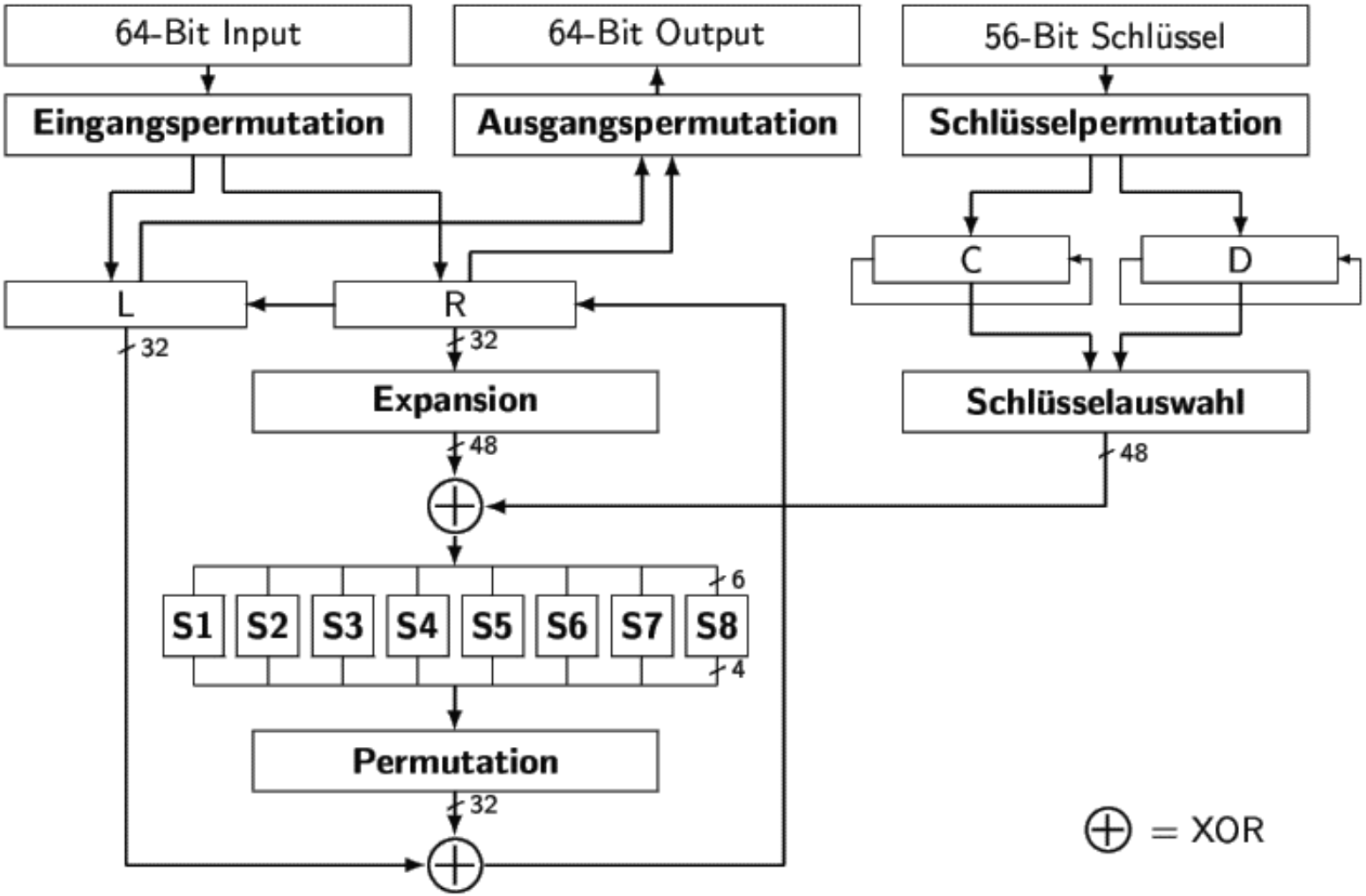
P-Box-Permutation von DES

- Auf den 32 Bit (Ausgabe der Substitution) wird eine weitere Permutation (P-Box-Permutation) angewandt
  - Wird auch einfach als P-Permutation bezeichnet



- Dann wird die Ausgabe mit der linken Hälfte XOR (bitweise binäre Addition) verknüpft

Gute Zusammenfassung des Ablaufs von DES



Bildquelle: `http://atterer.org/uni/crypto.html`

Tripple-DES

- Bei **Triple-DES** (auch 3DES genannt) ist eine Verbesserung von DES
  - Mehrfache Ausführung von DES mit drei verschiedenen Schlüsseln
- Bei 3DES wird jeder Datenblock...
  - mit dem 1.Schlüssel chiffriert
  - mit dem 2.Schlüssel dechiffriert
  - und mit dem 3.Schlüssel chiffriert
- Dieses Verfahren wird auch EDE (Encrypt-Decrypt-Encrypt) genannt

$$3\mathrm{DES}_{(K_1,K_2,K_3)} := \mathrm{DES}_{K_3} \circ \mathrm{DES}_{K_2}^{-1} \circ \mathrm{DES}_{K_1}$$

- Die Schlüssellänge ist mit 168 Bit dreimal so groß wie bei DES (56 Bit)
  - Dadurch vergrößert sich der Schlüsselraum um 2^{112}
  - Bedingt durch kryptographische Schwächen ist die effektive Schlüssellänge nur bei 112 Bit
- Ähnlich sicher wie Verfahren mit 128 Bit Schlüssellänge
- Hoher Rechenaufwand wegen dreimaliger Verschlüsselung $\Longrightarrow$ Relativ langsam

Advanced Encryption Standard (AES)

- Nachfolger von DES und 3DES
- Patentfrei
- Seit 2000 als US-Standard zur Sicherung staatlicher Dokumente der höchsten Geheimhaltungsstufe
- Wird heute u.a. genutzt, für:
  - Wireless LAN (802.11i, WPA2)
  - SSH
  - IP-Telefonie (z.B: Skype)
  - Festplattenverschlüsselung (MacOS X, TrueCrypt)
  - Sicherung von komprimierten Archiven (7-zip, RAR)
  - TLS/SSL (sichere Datenübertragung im Internet)
- Symmetrisches Verschlüsselungsverfahren
- Das Verfahren basiert auf dem Prinzip der Substitution
- Schlüssellänge ist 128, 192 oder 256 Bit
- Die Blockgröße von AES ist 128 Bit
- 10, 12 oder 14 Verschlüsselungsrunden (Schlüssellängenabhängig)

AES – Prinzip

- AES ist eine Blockchiffre
- Jeder Block mit 128 Bit wird zunächst in eine zweidimensionale Tabelle mit 4 Zeilen und 4 Spalten geschrieben
  - Jede Zelle der Tabelle ist 1 Byte groß
- Jeder Block wird nun innerhalb mehrerer Runden nacheinander bestimmten Transformationen unterzogen
- Die Anzahl der Verschlüsselungsrunden ist abhängig vom der Schlüsselgröße:
  - r(128Bit) = 10
  - r(192Bit) = 12
  - r(256Bit) = 14
  - R = r + 1 (Die Schlussrunde zählt auch als Runde)
- Dabei werden verschiedene Teile des erweiterten Originalschlüssels nacheinander auf den Klartext-Block angewendet

Sehr gute Beschreibung von AES

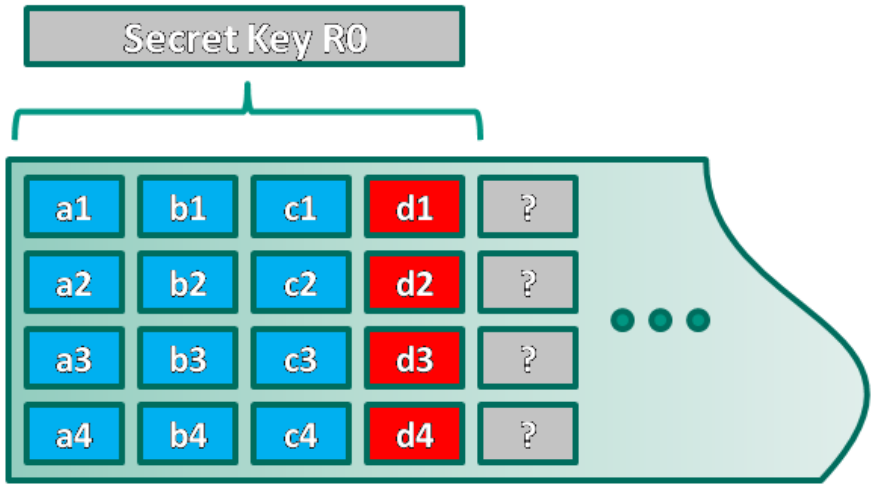
`http://www.codeplanet.eu/tutorials/cpp/51-advanced-encryption-standard.html`

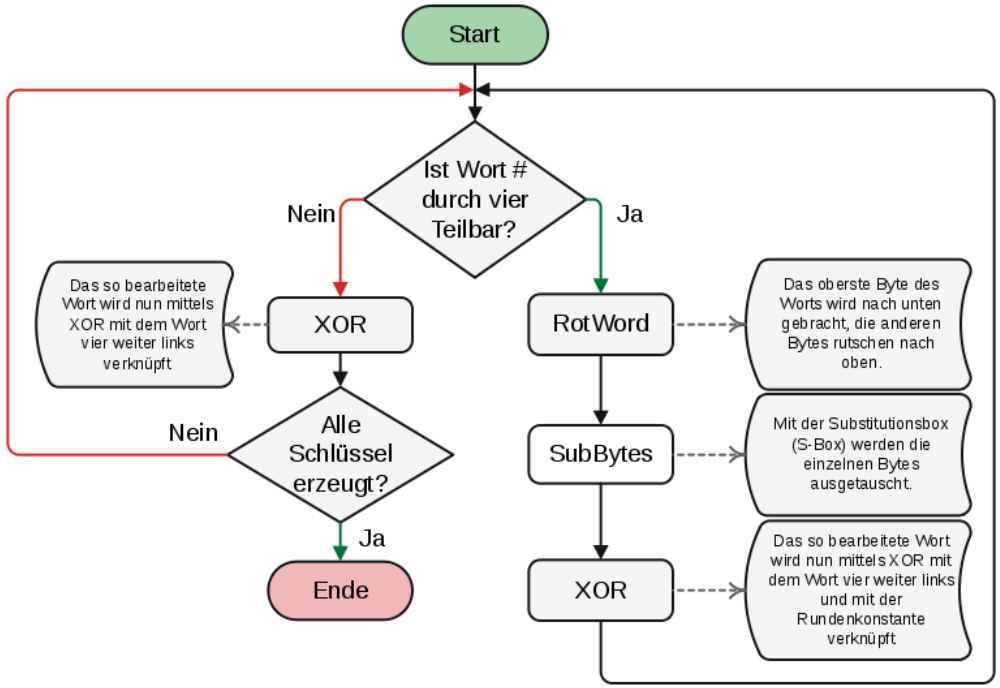
AES – Algorithmus

- Schlüsselexpansion
  - Aus dem geheimem Schlüssel werden rekursiv R weitere Rundenschlüssel erzeugt
- Vorrunde:
  - `KeyAddition(Rundenschlüssel[0])`
- Verschlüsselungsrunden (wiederhole solange Runde $<$ R):
  - `SubBytes()` $\Longleftarrow$ Hier findet die Substitution statt
  - `ShiftRows()`
  - `MixColumns()`
  - `KeyAddition(Rundenschlüssel[Runde])`
- Schlussrunde:
  - `SubBytes()` $\Longleftarrow$ Hier findet die Substitution statt
  - `ShiftRows()`
  - `KeyAddition(Rundenschlüssel[R])`

AES – Schlüsselexpansion (1/2)

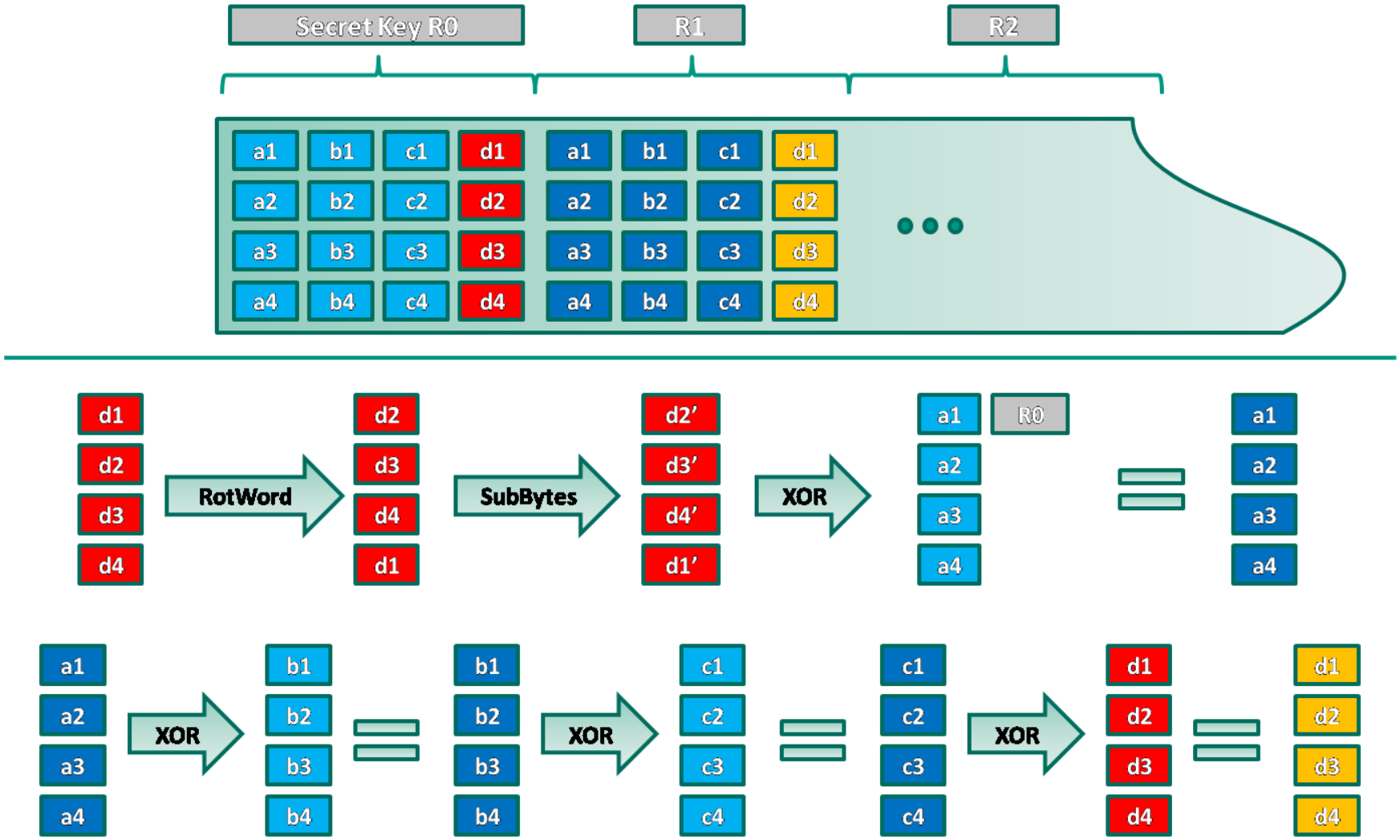
- Der Schlüssel wird in $R + 1$ Teilschlüssel (Rundenschlüssel) aufgeteilt
  - Die Rundenschlüssel müssen die gleiche Länge wie die Blöcke erhalten
  - Der Benutzerschlüssel muss auf die Länge $b * (R + 1)$ expandiert werden
    - b ist die Blockgröße
  - Schlüssel werden in Tabellen (4 Zeilen und Zellen mit je 1 Byte) gebildet
  - Die ersten Spalten der Tabelle werden mit dem Benutzerschlüssel gefüllt
    - Die weiteren Spalten werden wie folgt rekursiv berechnet





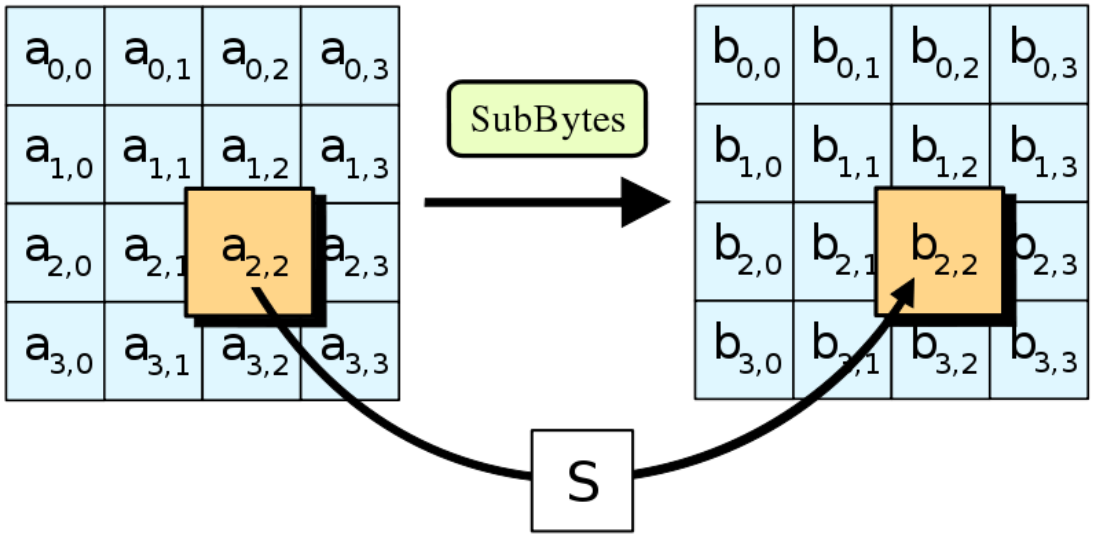
AES – Schlüsselexpansion (2/2)

- Es wird eine starke Diffusionswirkung auf Benutzerschlüssel ausgeübt



AES – SubBytes()

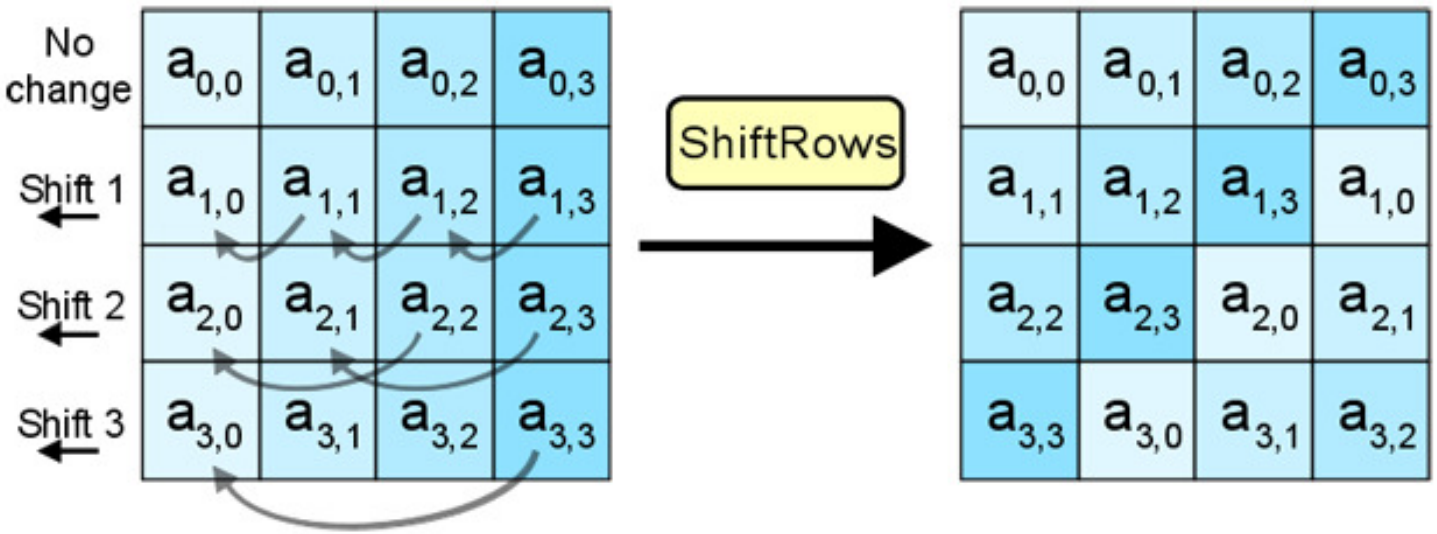
- Monoalphabetische Verschlüsselung mit einer Substitutionsbox (S-Box)
- Jedes Byte wird mit der S-Box in ein anderes Byte umgewandelt
- Im Gegensatz zu DES gibt es bei AES nur eine S-Boxen
  - Die Einträge der S-Box sind standardisiert




```
/* Forward S-Box */
static const uint8_t SBox[256] = {
    // 0     1     2     3  ...     F
    0x63, 0x7c, 0x77, 0x7b, ..., 0x76, // 0
    0xca, 0x82, 0xc9, 0x7d, ..., 0xc0, // 1
    0xb7, 0xfd, 0x93, 0x26, ..., 0x15, // 2
    0x04, 0xc7, 0x23, 0xc3, ..., 0x75, // 3
    0x09, 0x83, 0x2c, 0x1a, ..., 0x84, // 4
    0x53, 0xd1, 0x00, 0xed, ..., 0xcf, // 5
    0xd0, 0xef, 0xaa, 0xfb, ..., 0xa8, // 6
     ...,  ...,  ...,  ..., ...,  ...,
    0x8c, 0xa1, 0x89, 0x0d, ..., 0x16  // F
};
```

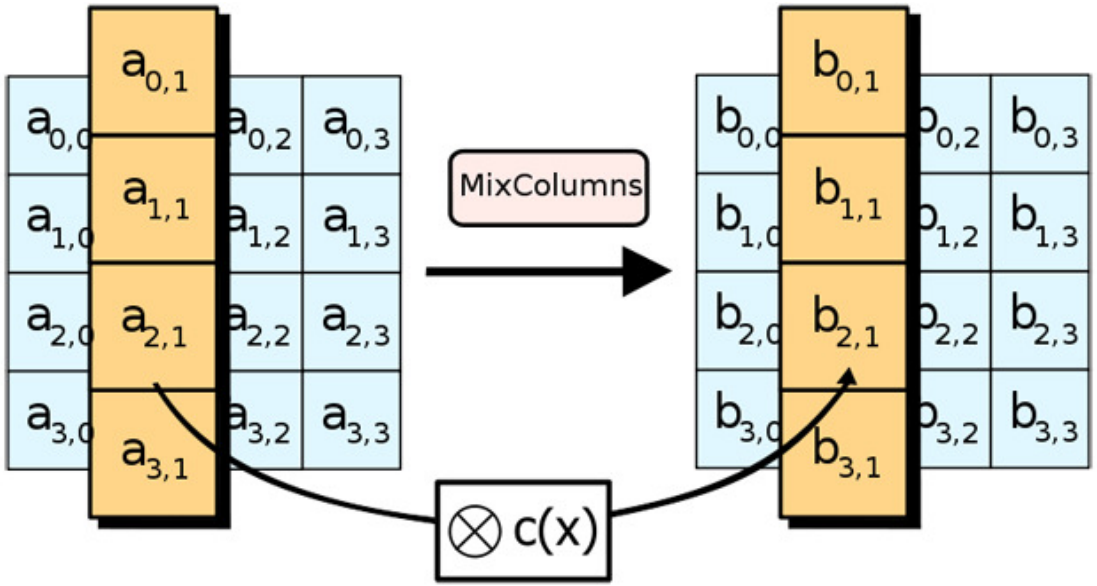
AES – ShiftRows()

- ShiftRows-Transformation, bei der die Bytes des Blocks permutiert werden
  - Die Zeilen werden um eine bestimmte Anzahl von Spalten nach links verschoben
  - Überlaufende Zellen werden von rechts fortgesetzt
  - Die Anzahl der Verschiebungen entspricht dem Zeilenindex



AES – MixColumns()

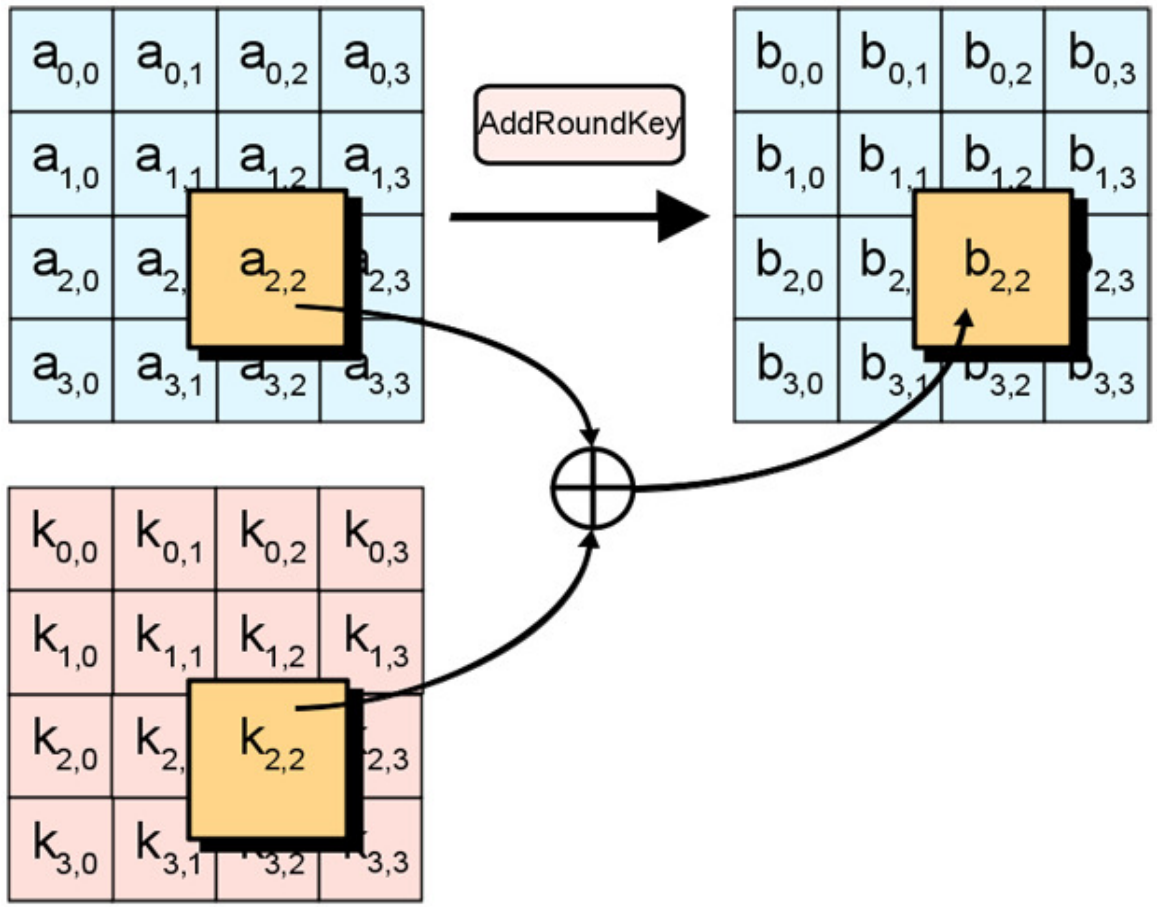
- Vermischung der Spalten
- Es findet eine Matrizenmultiplikation eines Spaltenvektors mit einer speziellen Matrix statt, damit alle 4 Eingabebytes jedes Ausgabebyte beeinflussen
- Ablauf: Jede Zelle einer Spalte wird mit einer Konstanten aus der Matrix multipliziert
  - Anschließend werden die Ergebnisse XOR-verknüpft



	Spalte			
	1	**2**	**3**	**4**
Zeile 1	2	3	1	1
Zeile 2	1	2	3	1
Zeile 3	1	1	2	3
Zeile 4	3	1	1	2

AES – KeyAddition()

- Bitweise XOR-Verknüpfung von Block und Schlüssel
- Einzige Funktion im AES-Algorithmus, die vom Schlüssel abhängig ist



Zusammenfassung (Symmetrische Verfahren)

Verschlüsselung	**Substitution/ Transposition**	**Monographisch/ Bigraphisch**	**Monoalphabetisch/ Polyalphabetisch**
Skytala	Transposition	—	—
Freimaurerchiffre	Substitution	Monographisch	Monoalphabetisch
Caesar	Substitution	Monographisch	Monoalphabetisch
Chiffrierscheibe	Substitution	Monographisch	je nach Anwendung
Polybius	Substitution	Monographisch	Monoalphabetisch
Playfair	Substitution	Bigraphisch	Monoalphabetisch
Vigenère	Substitution	Monographisch	Polyalphabetisch
One-Time-Pad	Substitution	—	Polyalphabetisch
ENIGMA	Substitution	Monographisch	Polyalphabetisch
DES	Substitution	—	Monoalphabetisch
Tripple-DES	Substitution	—	Monoalphabetisch
AES	Substitution	—	Monoalphabetisch