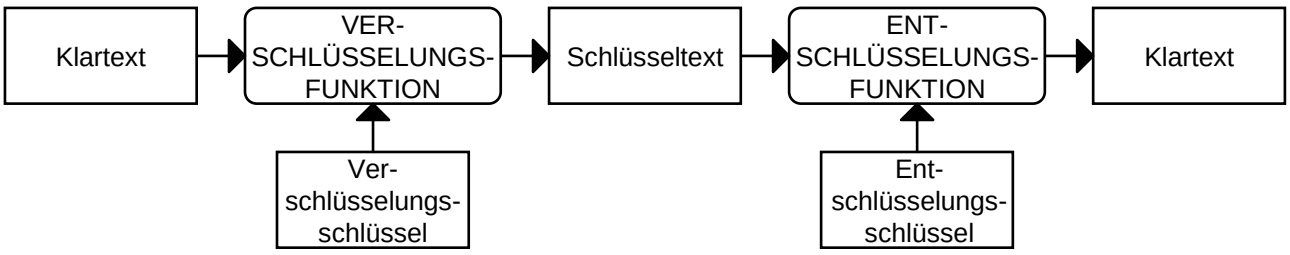
20 Kryptologie

- Die Kryptologie ist die Wissenschaft der Verschlüsselung und der Entschlüsselung von Informationen.
  Unterteilung:
  - Entwurf von Kryptosystemen: Kryptographie
  - Entschlüsselung von Informationen (ohne Kenntnis des Schlüssels): Kryptoanalyse
  - Verstecken von Informationen: Steganographie

- Anwendungen (Beispiele):
  - Verschlüsselung des Passworts beim Abspeichern
  - Verschlüsselung von Dateiinhalten oder Backup-Daten (auf Backup-Medien)
  - Verschlüsselung bei der Datenübertragung (in Netzen)

- Grundbegriffe:
  Die zu verschlüsselnden Daten werden Klartext (Plaintext) genannt. Dieser wird durch eine Funktion f (Abbildung, Verschlüsselungsfunktion) in einen Schlüsseltext (Chiffretext, Ciphertext) umgewandelt (Vorgang: Verschlüsseln, Chiffrieren, Encryption). Die Funktion wird von einem Schlüssel (Key, Verschlüsselungs-schlüssel, Encryption Key) parametrisiert:
  f (Klartext, Verschlüsselungsschlüssel) = Schlüsseltext
  Unter Anwendung der Umkehrfunktion f^{-1} (Entschlüsselungsfunktion) kann (bei bekanntem Schlüssel, Entschlüsselungsschlüssel, Decryption Key) aus dem Schlüsseltext wieder der Klartext abgeleitet werden (Vorgang: Entschlüsseln, Dechiffrieren, Decryption):
  f^{-1} (Schlüsseltext, Entschlüsselungsschlüssel) = Klartext

- Graphische Darstellung:



- Meist werden die Funktionen so gewählt, dass Ver- und Entschlüsselungsschlüssel dergleiche sein kann

- Block-Verschlüsselungsverfahren (Block Cipher): Jeweils ein Block fester Größe des Klartexts wird mit Hilfe der, durch den Schlüssel parametrisierten, Verschlüsselungsfunktion in einen Block des Schlüsseltexts transformiert.
  Problem: Wenn der Klartext Muster mit einer Periode enthält, die ein ganzzahliges Vielfaches oder ein ganzzahliger Bruchteil der verwendeten Blocklänge ist, ist eine Analyse der Häufigkeitsverteilung der Blöcke möglich.

- Strom-Verschlüsselungsverfahren (Stream Cipher): Der ganze Klartext wird als Zeichenstrom aufgefasst und insgesamt verschlüsselt. Aus dem Schlüssel wird nach einem definierten Verfahren ein Schlüsselstrom (Key Stream), derselben Länge wie die des Klartexts, erzeugt. Die Verschlüsselung erfolgt kontinuierlich, aus jeweils einem Teil des Klartexts und des Schlüsselstroms. Eine Untersuchung von Häufigkeitsverteilungen ist damit wesentlich erschwert.
  Problem: Synchronisation des Schlüsselstroms bei der Ver- und Entschlüsselungsfunktion (Sender und Empfänger). Geht durch einen Fehler (bei einer Übertragung) ein Teil des Schlüsseltexts verloren, wird der Rest mit einem anderen Schlüsselstrom entschlüsselt als verschlüsselt.

- Anzahl der Klartextzeichen, die (in einem Verschlüsselungsschritt) verwendet werden
  Eines: Monographische Verschlüsselung
  Mehrere: Polygraphische Verschlüsselung

- Einfachstes Verfahren der Kryptoanalyse ist die Brute-Force-Methode:
  Durchprobieren aller Schlüssel. Schlüssel muss so lang gewählt werden, dass dieses Verfahren wegen der vielen Möglichkeiten viel zu lange dauert.

20.1 Einfache Methoden

- Substitution (Ersetzungsverfahren):
  Jedes Zeichen (oder jede Zeichenfolge) des Klartexts wird durch ein anderes Zeichen (oder andere Zeichenfolge) ersetzt.

  - Monoalphabetische Substitution:
    Gleiche Zeichen des Klartexts werden durch gleiche Zeichen im Schlüsseltext ersetzt.
    - Sicherheit: Durch Analyse der Häufigkeitsverteilungen von einzelnen Zeichen, Zeichenpaaren und -tripel relativ einfach zu entschlüsseln.
    - Beispiel: Cäsar-Code
      Jeder Buchstabe wird durch den drittnächsten Buchstaben im Alphabet (zyklisch) ersetzt (es wird a zu d, b zu e, ..., z zu c).
    - Beispiel: Übersetzungstabelle

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
Schlüsseltext:	q	w	e	r	t	z	u	i	o	p	a	s	d	f	g	h	j	k	l	y	x	c	v	b	n	m

  - Polyalphabetische Substitution:
    Gleiche Zeichen des Klartexts werden durch verschiedene Zeichen im Schlüsseltext ersetzt.
    - Sicherheit: Gegenüber der monoalphabetischen Variante werden die Häufigkeitsverteilungen besser verschleiert, sind durch statistische Analysen aber trotzdem zu entschlüsseln.
    - Beispiel: Vigenère-Verschlüsselung (Blaise de Vigenère 1523-1596)
      Verschlüsselungstabelle besteht aus 26, verschieden angeordneten, Alphabeten. Jedes Zeichen des Verschlüsselungsschlüssels wählt eine Zeile aus, jedes Zeichen des Klartexts eine Spalte, im Kreuzungspunkt steht das Schlüsseltextzeichen. Wenn der Verschlüsselungsschlüssel kürzer als der Klartext ist (trifft üblicherweise zu), muss der Verschlüsselungsschlüssel entsprechend oft wiederholt werden.

	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
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

Klartext:	d	e	r	w	u	n	s	c	h	n	a	c	h	g	e	h	e	i	m	h	a	l	t	u	n	g
Schlüssel:	i	n	f	o	r	m	a	t	i	k	i	n	f	o	r	m	a	t	i	k	i	n	f	o	r	m
Schlüsseltext:	l	r	w	k	l	z	s	v	p	x	i	p	m	u	v	t	e	b	u	r	i	y	y	i	e	s

  - Durch Substitution von ganzen Zeichenfolgen anstatt einzelner Zeichen können Häufigkeitsanalysen zusätzlich erschwert werden (Beispiel: Porta-Verschlüsselung)

- Transposition (Permutation, Versetzungsverfahren):
  Die Reihenfolge der Zeichen (oder Zeichengruppen) des Klartexts wird verändert.
  - Sicherheit: Da Transpositionen nur jeweils auf (relativ kurze) Zeichenfolgen (Teilfolgen) des Klartexts angewendet werden können, ist durch entsprechendes Analysieren eine Entschlüsselung möglich.
  - Beispiel: Skytala der Spartaner (ca. 500 v.Chr.)
    Ein fingerbreiter Pergamentstreifen wurde auf einen Holzstab (die Skytala) gewickelt und mit dem Klartext in Längsrichtung des Stabes beschrieben. Der Klartext konnte nur wieder von einem Empfänger gelesen werden, der einen Holzstab des gleichen Durchmessers besaß.
  - Beispiel: Der Verschlüsselungsschlüssel ist ein Wort ohne Buchstabenwiederholungen (z.B. megabuck). Die relativen Stellungen der Buchstaben dieses Worts im Alphabet stellen die Transposition dar.

```
m e g a b u c k
_______________
7 4 5 1 2 8 3 6
_______________
d e r w u n s c
h n a c h a u f
l o e s u n g x
```

  Klartext: d e r w u n s c h n a c h a u f l o e s u n g x
  Schlüsseltext: w c s u h u s u g e n o r a e c f x d h l n a n

- Vernam-Verfahren (Einmalschlüssel, One-Time-Pad) ca. 1925, Gilbert Vernam (1890-1960):
  Verschlüsselungsverfahren, das theoretisch absolut sicher ist.
  Es wird ein Schlüssel gewählt, der so lang wie der Klartext ist und der aus einer zufälligen Bitfolge besteht. Der Schlüsseltext wird durch bitweises Exklusiv-Oder von Klartext und Schlüssel gebildet. Die Bitfolge im entstandenen Schlüsseltext stellt wieder eine reine Zufallsfolge dar und kann daher nicht aufgebrochen werden. In der Praxis besteht die Schwierigkeit in der Verwaltung von beliebig langen, zufälligen Bitfolgen. Pseudo-Zufallszahlenfolgen, wie sie von Rechnern algorithmisch erzeugt werden, haben eine (wenn auch große, aber trotzdem) endliche Periode und sind daher nicht absolut sicher. Um nicht immer diegleiche Bit-Folge vom Pseudo-Zufallszahlengenerator erzeugen zu lassen, kann der Startwert des Generators mit einem Schlüssel parametrisiert werden.

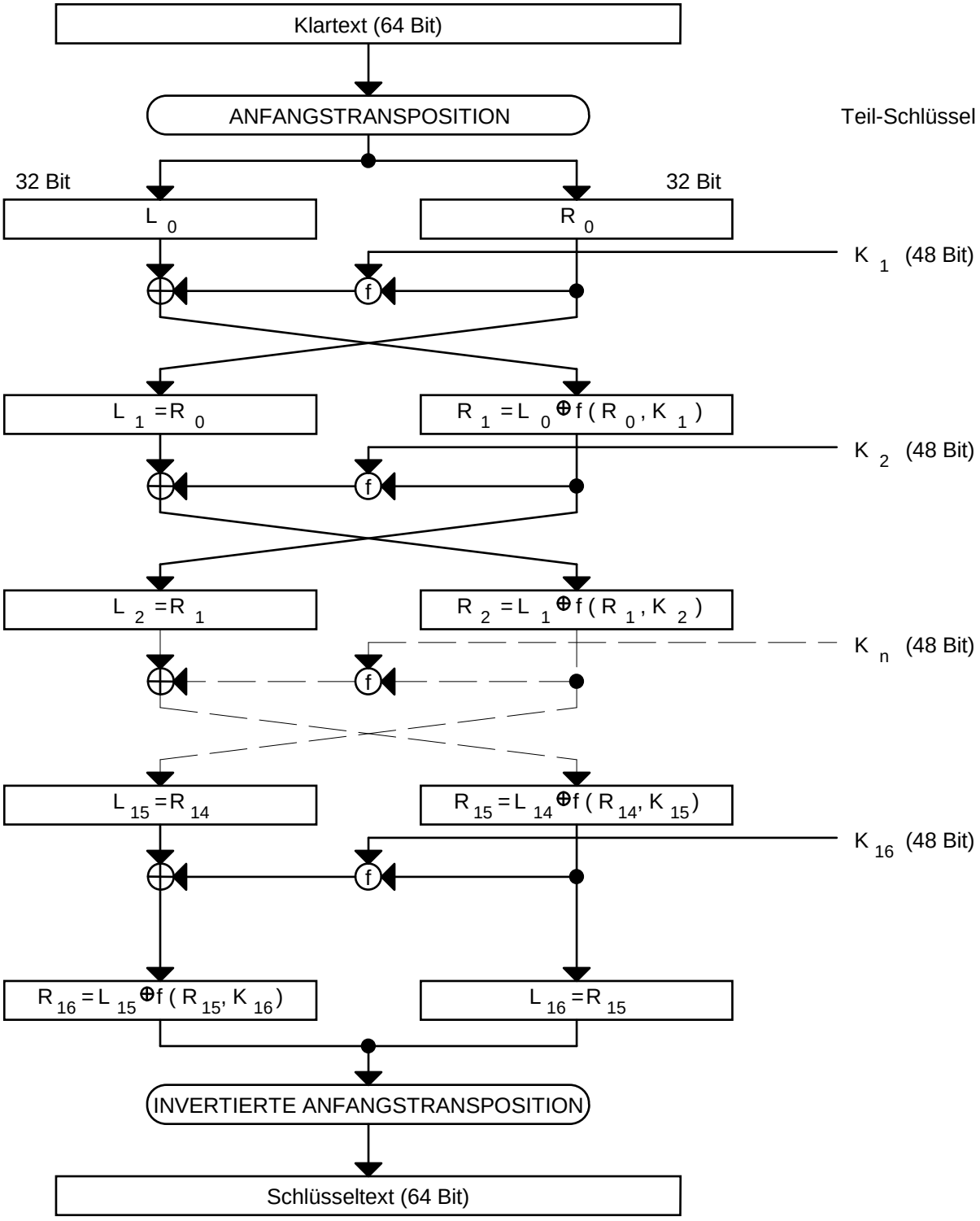
20.2 Die Datenverschlüsselungsnorm DES (Data Encryption Standard)

- Von IBM entwickelt, 1977 vom National Bureau of Standards veröffentlicht, auch gut für hardwaremäßige Implementierung geeignet (En-/Decryption-Chips, Geschwindigkeit!)

- Besteht im Wesentlichen aus wiederholten Anwendungen von Substitutionen und Transpositionen

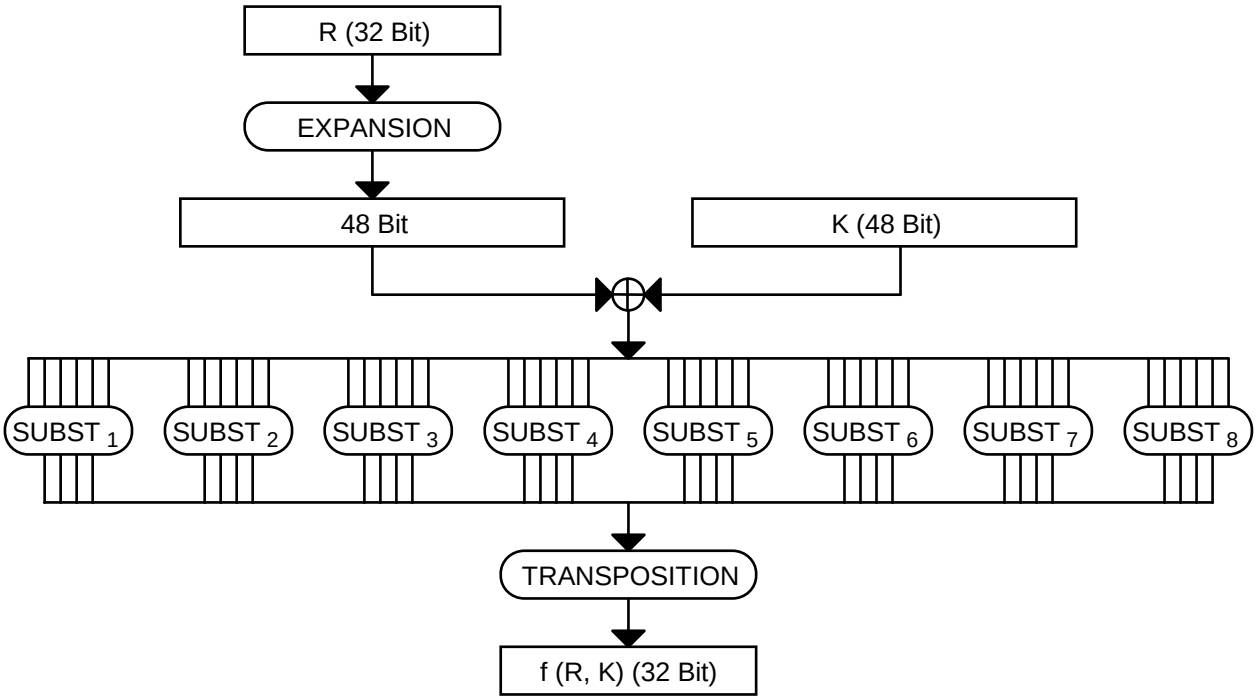
- Jeweils 8 Zeichen (64Bit) des Klartexts werden mit einem, für die Verschlüsselung des gesamten Klartexts gültigen, Schlüssel von 7 Zeichen (56Bit) in 8 Zeichen (64Bit) Schlüsseltext umgesetzt (der Schlüssel wird bei seiner Anwendung um 8 Paritäts-Bits verlängert)

- Schema:



  - Zunächst erfolgt eine, vom Schlüssel unabhängige, Transposition des Klartexts
  - Dann werden 16 Transformationen durchgeführt, die (bis auf die letzte) in ihrer Funktionsweise identisch sind; für jede Stufe wird allerdings ein, aus dem Schlüssel abgeleiteter, unterschiedlicher Teilschlüssel verwendet. Die Komplexität liegt in der Funktion f (R, K), die aus dem rechten Teil der Vorstufe (R, 32Bit) und dem entsprechenden Teilschlüssel (K, 48Bit) ein Ergebnis mit 32Bit liefert.
  - Zum Schluss wird die, zur Transposition des ersten Schrittes, inverse Transposition angewendet

- Berechnung der f (R, K):



- Folgende Informationen sind im DES tabellenmäßig definiert:
  - 1 Anfangstransposition (64Bit)
  - Auswahl der 16 Teilschlüssel aus dem Schlüssel (48Bit aus 64Bit)
  - 1 Expansion in f (32Bit auf 48Bit)
  - 8 Substitutionen in f (6Bit in 4Bit)
  - 1 Transposition in f (32Bit)

- Die Entschlüsselung erfolgt mit demselben Schlüssel und demgleichen Verfahren wie die Verschlüsselung, jedoch wird mit der invertierten Anfangstransposition begonnen und werden die 16 Teilschlüssel in umgekehrter Reihenfolge angewendet

- Betriebsarten des DES:
  - ECB-Modus (Electronic Code Book, Elektronisches Codebuch): Der Klartext wird in Blöcke von je 8 Zeichen eingeteilt und jeder Block separat verschlüsselt (Block-Verschlüsselung). Diese Methode produziert identische Schlüsseltextblöcke bei gleichen Klartextblöcken.
  - CFM-Modus (Cipher Feedback Mode, Schlüsselrückführung): 8-Bit (oder mehr) eines jeden Schlüsseltextblocks werden durch eine Exklusiv-Oder-Funktion mit dem nächsten Klartextblock vor dessen Verschlüsselung verknüpft. Derselbe Klartext erscheint damit jedesmal als unterschiedlicher Schlüsseltext.
  - CBC-Modus (Cipher Block Chain, Schlüsselblockkette): Jeder Schlüsseltextblock wird als Ganzes zum nächsten Klartextblock rückgekoppelt. Damit ist jeder verschlüsselte Datenblock eine auch Funktion des vorhergegangenen Datenblocks.

- Zeitverhalten: 2^{56} (ca. 70 Billiarden) verschiedene Schlüssel gibt es; wenn pro Sekunde 1 Million Überprüfungen durchgeführt werden können und im Durchschnitt die Hälfte der möglichen Schlüssel ausprobiert werden, dann dauert dies mehr als 1000 Jahre (10000 solcher Chips, in einem Parallelcomputer vereinigt, würden ca. 40 Tage brauchen)

- Kritik:
  - Geringe Länge des Schlüssels. Ursprünglicher IBM-Vorschlag war 128 Bits / 16 Zeichen, dieser wurde vom US-Geheimdienst NSA (National Security Agency) verhindert.
  - Blocklänge nur 64 Bits
  - Es sind eine Anzahl 'Schwacher Schlüssel' bekannt, bei deren Anwendung eine Entschlüsselung möglich ist.

- 3DES (Triple-DES): DES mehrmals hintereinander mit zwei verschiedenen Schlüsseln
  (Schlüssel praktisch 112 Bit lang)

- Als neuer Standard im Jahr 2000 festgelegt:
  AES (Advanced Encryption Standard), nach seinen Entwicklern Joan Daemen und Vincent Rijmen (Belgier) auch Rijndael-Algorithmus genannt. Symmetrisches Blockverschlüsselungsverfahren.
  Schlüssellänge: 128, 192 oder 256 Bit; Blocklänge: 128 Bit

20.3 Systeme mit öffentlichen Schlüsseln (Public-Key Systems)

- Die bisher besprochenen Verfahren haben den Nachteil, dass die verwendeten Schlüsseln zwischen dem, der den Klartext verschlüsselt (Sender) und dem, der den Schlüsseltext entschlüsselt (Empfänger), ausgetauscht werden müssen. Dieser Schlüsselaustausch stellt naturgemäß ein Sicherheitsrisiko dar, vor allem, wenn die Schlüssel zur Gewährleistung der Geheimhaltung häufig geändert werden müssen.

- Im Prinzip fuktioniert ein Public-Key Verfahren folgendermaßen (Idee von Diffie und Hellman, 1976):
  Jedem Teilnehmer sind zwei verschiedene, jedoch nach mathematischen Regeln zusammengehörige, Schlüssel zugeordnet:
  - ein öffentlicher Schlüssel: Steht jedem zur Verfügung, der dem betreffenden Teilnehmer eine Nachricht zukommmen lassen möchte. Die öffentlichen Schlüssel aller Teilnehmer sind in einem allgemein zugänglichen Verzeichnis vermerkt ('Telefonbuch'). Dieser Schlüssel wird üblicherweise zum Verschlüsseln benutzt.
  - ein privater Schlüssel (Geheimschlüssel): Kennt nur der Teilnehmer selbst. Dieser Schlüssel wird üblicherweise zum Entschlüsseln benutzt.

  Aufgrund dieser Tatsache werden Public-Key Systeme auch als Zweischlüssel-Systeme bezeichnet (die bisher besprochenen Verfahren können demnach als Einschlüssel-Systeme bezeichnet werden).

- Alle Methoden, die aus der mathematischen Klasse der Falltür-Probleme (Trap-Door-Problems) stammen, sind bei der Anwendung von Public-Key Systemen geeignet. In der Folge wird das verbreitetste Verfahren, der RSA-Code, beschrieben.

- RSA-Code:

  - Benannt nach den Anfangsbuchstaben seiner drei Erfinder:
    Ronald Rivest, Adi Shamir, Leonard Adleman (1978)
  - Das System beruht auf der Multiplikation zweier sehr großer Primzahlen (je ca. 100 Stellen). Es ist vergleichsweise einfach, zwei große Primzahlen zu multiplizieren, jedoch sehr aufwendig, die beiden Primzahlen zu ermitteln, wenn nur das Produkt (ca. 200 Stellen) bekannt ist (Faktorisierungsproblem).
    Beispiel: Um 29083 manuell in seine beiden Primfaktoren zu zerlegen wird man etwa eine halbe Stunde brauchen. Die Multiplikation von 127 mit 229 dauert manuell etwa eine Minute.

  - Zwei verschiedene, möglichst (je ca. 100 Stellen) große, Primzahlen p und q werden ausgewählt:
    $n=p*q$, $z=(p-1)*(q-1)$
  - Eine möglichst große Zahl e wird ausgewählt, die teilerfremd zu z ist
    (z.B. eine Primzahl, die sowohl größer als p, als auch größer als q ist)
  - d wird bestimmt als das Multiplikative Inverse von e modulo z ($d * e$ modulo $z = 1$)
    Methode: Erweiterter Euklidscher Algorithmus (günstiges Zeitverhalten)
  - Öffentlicher Schlüssel: e,n
  - Privater Schlüssel: d,n (n nicht notwendig, ist ohnehin öffentlich bekannt)
  - Verschlüsselung: $C = P^e$ modulo n
  - Entschlüsselung: $P = C^d$ modulo n
  - Die Sicherheit der Methode hängt damit zusammen, dass die beiden Faktoren p und q des bekannten n (innerhalb eines brauchbaren Zeitrahmens) nicht ermittelt werden können
    (sonst könnten auch z, sowie aus z und e auch d berechnet werden)

- Beispiel (mit sehr kleinen Zahlen):
  p=3, q=5; daher n=15, z=8; e=11; daher d=3; P=13

  $C = 13^{11}$ modulo 15 = 1792160394037 modulo 15 = 7

  $P = 7^{3}$ modulo 15 = 343 modulo 15 = 13

  Rechengang: Zerlegen des Exponenten in Zweierpotenzen, es gilt
  (a*b) mod n = ((a mod n) * (b mod n)) mod n
  Module der Faktoren multiplizieren, statt Modul des Produkts ermitteln

  $C = 13^{11}$ modulo 15 = 13^{8} * (13^{4}) * 13^{2} * 13^{1} mod 15 =
  1 * (1) * 4 * 13 mod 15 = 7

  $P = 7^{3}$ modulo 15 = 7^{2} * 7^{1} mod 15 =
  4 * 7 mod 15 = 13

- Beispiel (mit kleinen Zahlen):
  p=73, q=151; daher n=11023, z=10800; e=11; daher d=5891;
  P = H o w _ a r e _ y o u ?
  33 14 22 62 00 17 04 62 24 14 20 66
  a,b,...,z $\rightarrow$ 00, 01, ..., 25
  A,B,...,Z $\rightarrow$ 26, 27, ..., 51
  Sonderzeichen $\rightarrow$ 52, ..., 66
  Block: 2 Zeichen, da 6666 kleiner 11023, jedoch 666666 nicht mehr kleiner 11023 ist.

  $C1 = 3314^{11}$ modulo 11023 = 10260 $C2 = 2262^{11}$ modulo 11023 = 9489

  $C3 = 17^{11}$ modulo 11023 = 1782 $C4 = 462^{11}$ modulo 11023 = 727

  $C5 = 2414^{11}$ modulo 11023 = 10032 $C6 = 2066^{11}$ modulo 11023 = 2253

  $P1 = 10260^{5891}$ modulo 11023 = 3314 $P2 = 9489^{5891}$ modulo 11023 = 2262

  $P3 = 1782^{5891}$ modulo 11023 = 17 $P4 = 727^{5891}$ modulo 11023 = 462

  $P5 = 10032^{5891}$ modulo 11023 = 2414 $P6 = 2253^{5891}$ modulo 11023 = 2066

- Zeitverhalten: Berechnung von p,q und d aus n und e

Stellen von n	Rechenoperationen	Rechenzeit (1 Operation in 10^{-6} Sekunden)	
50	$1{,}4*10^{10}$	3,9	Stunden
70	$9{,}0*10^{12}$	104	Tage
100	$2{,}3*10^{15}$	74	Jahre
200	$1{,}2*10^{23}$	$3{,}8*10^{9}$	Jahre

- Anzahl großer Primzahlen: Es gibt ca. $10^{60} / \ln(10^{60})$ Primzahlen mit 60 Stellen (Gaußsches Primzahlentheorem). Damit hätte jedes Atom der Erde (ca. 10^{50}) seine eigenen zwei Primzahlen.

- Digitale Unterschrift (Authentifikation):

  - Bei Verwendung von Public-Key Systemen kann (auf Grund der Eigenschaften der verwendeten mathematischen Funktionen) die Nachricht P vom Sender A so verschlüsselt werden ('unterschrieben werden'), dass der Empfänger B sicher sein kann, die verschlüsselte Nachricht C tatsächlich vom Sender A erhalten zu haben (A kann im Streitfalle nicht leugnen, dass die Nachricht von ihm stammt).
  - Sei $E_X(Y)$ das Ergebnis des Verschlüsselungsvorgangs mit dem öffentlichen Schlüssel des Teilnehmers X angewendet auf den Text Y und $D_X(Y)$ das Ergebnis des Entschlüsselungsvorgangs mit dem privaten Schlüssel des Teilnehmers X angewendet auf den Text Y.
  - Verschlüsselung durch A: $C = E_B(D_A(P))$; wendet vor der eigentlichen Verschlüsselung die Entschlüsselung mit seinem privaten Schlüssel an.
  - Entschlüsselung durch B: $P = E_A(D_B(C))$; wendet nach der eigentlichen Entschlüsselung die Verschlüsselung mit dem öffentlichen Schlüssel von A an.
  - Beispiel (RSA-Code):
    Öffentlicher Schlüssel von A: (3,55=5*11) Privater Schlüssel von A: (27,55)
    Öffentlicher Schlüssel von B: (5,119=7*17) Privater Schlüssel von B: (77,119)

    Verschlüsselung durch A: P = 19; C' = 19^{27} modulo 55 = 24; C = 24^5 modulo 119 = 96

    Entschlüsselung durch B: P' = 96^{77} modulo 119 = 24; P = 24^3 modulo 55 = 19
  - Anwendungsgebiete: Telebanking, Juristische Anwendungen

- Kritik: Public-Key Systeme sind relativ langsam, da es sich um reine Softwareimplementierungen handelt (z.B. 80 Zeichen zu verschlüsseln dauert mit RSA 100 mal so lange als die entsprechende DES-Verschlüsselung)

- PGP - Pretty Good Privacy: Konkretes System, das auch die RSA-Methode verwendet

- Neueres Verfahren für Public-Key Verschlüsselung:
  Elliptic Curve Cryptography (ECC), Neal Koblitz und Victor Miller (1986):
  - Angriff erfordert noch wesentlich mehr Rechenaufwand als bei RSA
  - Es können kleinere Zahlen als beim RSA eingesetzt werden (Smartcards, PDAs, etc.)
  - 1024 Bit RSA entspreicht 160 Bit ECC (allerdings höherer Rechenaufwand)