Block Codes

incl. DES und AES

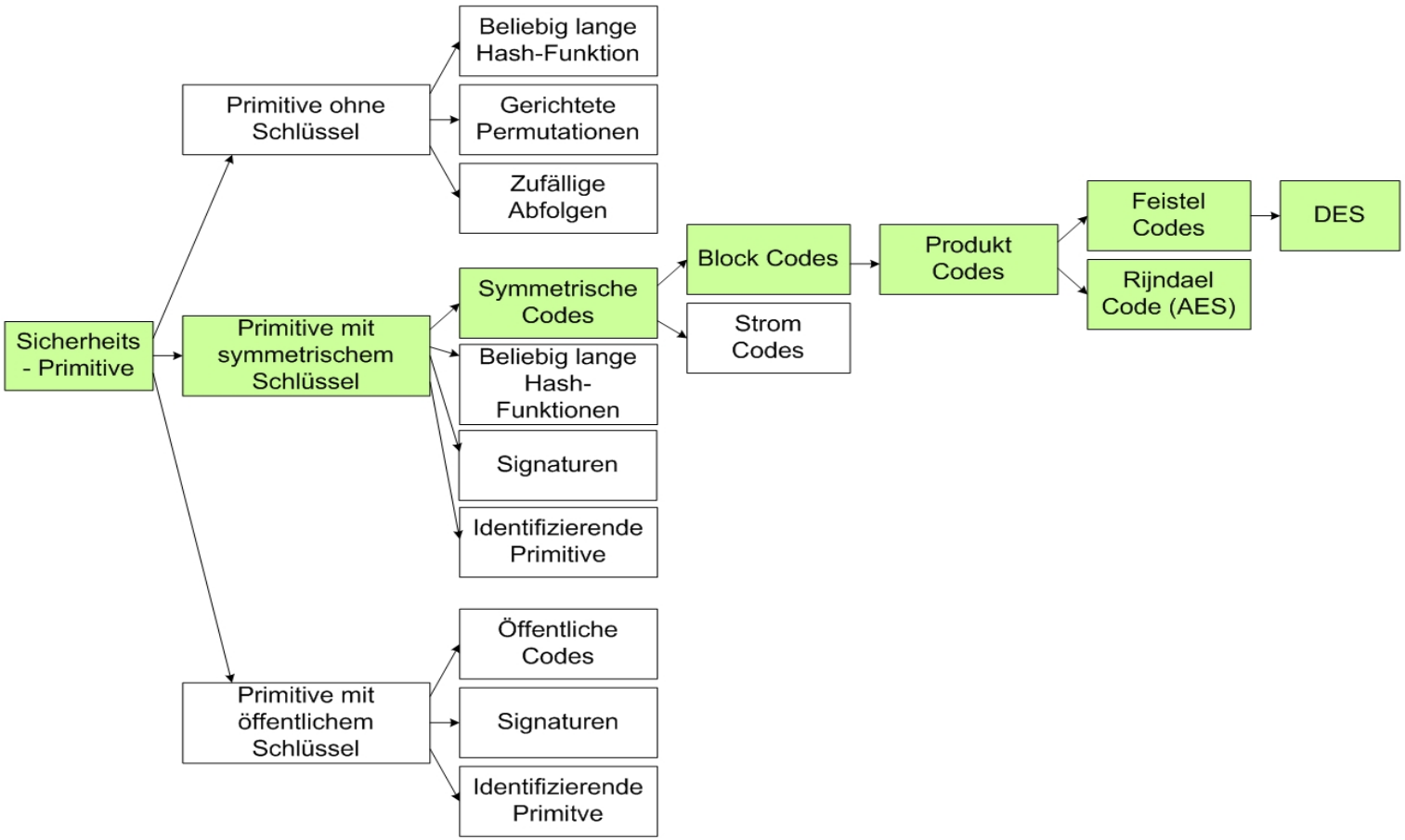
Bastian Brehmer und Jan Kleeßen

Prof. Dr. Kreitz
Kryptographie WiSe0607
Institut für Informatik
Universität Potsdam

8. November 2006

Inhaltsverzeichnis

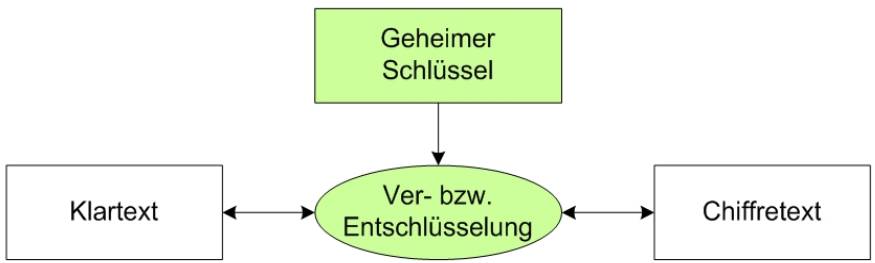
Eingliederung



Symmetrische Verschlüsselung

Symmetrischer Schlüssel

Zum Ver- und Entschlüsseln wird der **gleiche** Schlüssel verwendet.



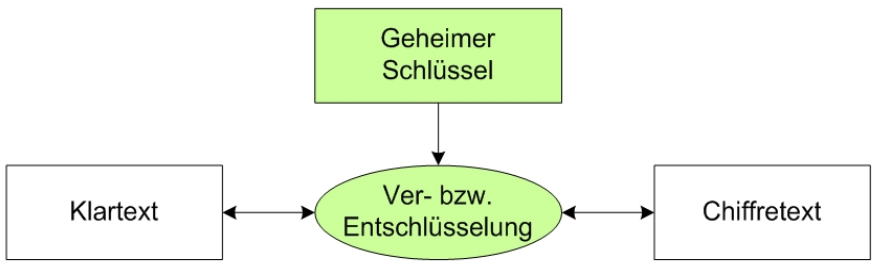
Problem

... liegt in der Übertragung des geheimen Schlüssels.

Symmetrische Verschlüsselung

Symmetrischer Schlüssel

Zum Ver- und Entschlüsseln wird der **gleiche** Schlüssel verwendet.



Problem

... liegt in der Übertragung des geheimen Schlüssels.

Block Codes

- Die Klartextumwandelung erfolgt in Binärcode
- Binärcodeaufteilung in Blöcke gleicher Größe
- Den Vorgang dieser Verschlüsselung nennt man *Blockchiffre*
- Eine Blockchiffre E_n mit einer Blockgröße von n Bits ist eine bijektive Abbildung:

$$E_n : \{0,1\}^n \rightarrow \{0,1\}^n$$

Produktverschlüsselung

Motivation

Einfache Block Codes können nicht nur leicht generiert werden, sondern auch genauso leicht wieder gebrochen werden.

- .. ist die Kombination von Ersetzung *(Substitution)* und Vertauschung *(Transposition bzw. Permutation)*
- Produktverschlüsselung meint allgemein:

$$c_i = E_{e1}\left(E_{e2}\left(...\left(E_{et}\left(m_i\right)...\right)\right)\right) t \geq 2$$

- Dabei wird die Hintereinanderausführung von Substitution und Permutation als Runde bezeichnet.
- Dies erziehlt eine Härtung der Verschlüsselung

Produktverschlüsselung

Motivation

Einfache Block Codes können nicht nur leicht generiert werden, sondern auch genauso leicht wieder gebrochen werden.

- .. ist die Kombination von Ersetzung *(Substitution)* und Vertauschung *(Transposition bzw. Permutation)*
- Produktverschlüsselung meint allgemein:

$$c_i = E_{e1}\left(E_{e2}\left(...\left(E_{et}\left(m_i\right)...\right)\right)\right) t \geq 2$$

- Dabei wird die Hintereinanderausführung von Substitution und Permutation als Runde bezeichnet.
- Dies erziehlt eine Härtung der Verschlüsselung

Eigenschaften der Produkt-Chiffren (1)

- Endomorphie: $\mathcal{P}_1 = \mathcal{C}_1 = \mathcal{P}_2 = \mathcal{C}_2$
- Zusammengesetzte Schlüsselfunktionen:

$$e_{K_1,K_2}(x) = e_{K_1}(e_{K_2}(x))$$
$$d_{K_1,K_2}(x) = d_{K_1}(d_{K_2}(x))$$

- Produkt-Chiffre $S_1 \times S_2$:

$$S_1 = \{\mathcal{P}, \mathcal{C}, \mathcal{K}_1, \mathcal{E}_1, \mathcal{D}_1\}$$
$$S_2 = \{\mathcal{P}, \mathcal{C}, \mathcal{K}_2, \mathcal{E}_2, \mathcal{D}_2\}$$
$$S_1 \times S_2 = \{\mathcal{P}, \mathcal{C}, \mathcal{K}_1 \times \mathcal{K}_2, \mathcal{E}, \mathcal{D}\}$$

Eigenschaften der Produkt-Chiffren (2)

- können *kommutativ* sein: $S_1 \times S_2 = S_2 \times S_1$
- können *idempotent* sein: $S^n = S^{n-1} = ... = S^2 = S \times S = S$
- sind immer *assoziativ*
- Das Produkt zweier idempotenter Kryptosysteme ist wieder idempotent.

$$
\begin{aligned}
(S_1 \times S_2) \times (S_1 \times S_2) &= S_1 \times (S_2 \times S_1) \times S_2 && \mid \text{assoziativ} \\
&= S_1 \times (S_1 \times S_2) \times S_2 && \mid \text{kommutativ} \\
&= (S_1 \times S_1) \times (S_2 \times S_2) && \mid \text{assoziativ} \\
&= S_1 \times S_2 && \mid \text{idempotent}
\end{aligned}
$$

Eigenschaften der Produkt-Chiffren (2)

- können *kommutativ* sein: $S_1 \times S_2 = S_2 \times S_1$
- können *idempotent* sein: $S^n = S^{n-1} = \ldots = S^2 = S \times S = S$
- sind immer *assoziativ*
- Das Produkt zweier idempotenter Kryptosysteme ist wieder idempotent.

$$
\begin{aligned}
(S_1 \times S_2) \times (S_1 \times S_2) &= S_1 \times (S_2 \times S_1) \times S_2 && \mid \text{assoziativ} \\
&= S_1 \times (S_1 \times S_2) \times S_2 && \mid \text{kommutativ} \\
&= (S_1 \times S_1) \times (S_2 \times S_2) && \mid \text{assoziativ} \\
&= S_1 \times S_2 && \mid \text{idempotent}
\end{aligned}
$$

Eigenschaften der Produkt-Chiffren (2)

- können *kommutativ* sein: $S_1 \times S_2 = S_2 \times S_1$
- können *idempotent* sein: $S^n = S^{n-1} = ... = S^2 = S \times S = S$
- sind immer *assoziativ*
- Das Produkt zweier idempotenter Kryptosysteme ist wieder idempotent.

$$
\begin{aligned}
(S_1 \times S_2) \times (S_1 \times S_2) &= S_1 \times (S_2 \times S_1) \times S_2 && \mid \text{assoziativ} \\
&= S_1 \times (S_1 \times S_2) \times S_2 && \mid \text{kommutativ} \\
&= (S_1 \times S_1) \times (S_2 \times S_2) && \mid \text{assoziativ} \\
&= S_1 \times S_2 && \mid \text{idempotent}
\end{aligned}
$$

Eigenschaften der Produkt-Chiffren (2)

- können *kommutativ* sein: $S_1 \times S_2 = S_2 \times S_1$
- können *idempotent* sein: $S^n = S^{n-1} = \ldots = S^2 = S \times S = S$
- sind immer *assoziativ*
- Das Produkt zweier idempotenter Kryptosysteme ist wieder idempotent.

$$
\begin{aligned}
(S_1 \times S_2) \times (S_1 \times S_2) &= S_1 \times (S_2 \times S_1) \times S_2 && \mid \text{assoziativ} \\
&= S_1 \times (S_1 \times S_2) \times S_2 && \mid \text{kommutativ} \\
&= (S_1 \times S_1) \times (S_2 \times S_2) && \mid \text{assoziativ} \\
&= S_1 \times S_2 && \mid \text{idempotent}
\end{aligned}
$$

Eigenschaften der Produkt-Chiffren (2)

- können *kommutativ* sein: $S_1 \times S_2 = S_2 \times S_1$
- können *idempotent* sein: $S^n = S^{n-1} = ... = S^2 = S \times S = S$
- sind immer *assoziativ*
- Das Produkt zweier idempotenter Kryptosysteme ist wieder idempotent.

$$\begin{aligned}
(S_1 \times S_2) \times (S_1 \times S_2) &= S_1 \times (S_2 \times S_1) \times S_2 && \mid \text{assoziativ} \\
&= S_1 \times (S_1 \times S_2) \times S_2 && \mid \text{kommutativ} \\
&= (S_1 \times S_1) \times (S_2 \times S_2) && \mid \text{assoziativ} \\
&= S_1 \times S_2 && \mid \text{idempotent}
\end{aligned}$$

Eigenschaften Iterativer Chiffren

- Anzahl der Runden Nr
- Hauptschlüssel K zur Erstellung des Schlüsselplans
- Schlüsselplan mit Nr Unterschlüsseln $\left(K^1, ..., K^n\right)$
- Zustände $\left(w^0, ..., w^{Nr}\right)$
- Rundenfunktion $g\left(w^{r-1}, K^r\right)$

Eigenschaften: Substitutions-Permutations Netzwerk

Was sind SPN's

- ein SPN ist eine iterative Chiffre
- Zusammensetzung aus: Substitutions- und Permutationschiffre kombiniert mit Bit-Addition ($\otimes$ XOR)

- Blocklänge, Blockanzahl, Rundenzahl: $l, m, Nr \in \mathbb{N}$
- Eine Bit-Permutation der Länge l: $\pi_S : \{0,1\}^l \rightarrow \{0,1\}^l$
- Eine Permutation der Länge l: $\pi_P : \{1, ..., lm\}^l \rightarrow \{1, ..., lm\}^l$
- Klar- und Chiffretext-Alphabet: $\mathcal{P} = \mathcal{C} = \{0,1\}^{lm}$
- Schlüsselplan als Teilmenge aller mögl. Schlüssel:

$$\mathcal{K} \subseteq \left(\{0,1\}^{lm}\right)^{Nr+1}$$

Eigenschaften: Substitutions-Permutations Netzwerk

Was sind SPN's

- ein SPN ist eine iterative Chiffre
- Zusammensetzung aus: Substitutions- und Permutationschiffre kombiniert mit Bit-Addition ($\otimes$ XOR)

- Blocklänge, Blockanzahl, Rundenzahl: $l, m, Nr \in \mathbb{N}$
- Eine Bit-Permutation der Länge l: $\pi_S : \{0,1\}^l \rightarrow \{0,1\}^l$
- Eine Permutation der Länge l: $\pi_P : \{1, ..., lm\}^l \rightarrow \{1, ..., lm\}^l$
- Klar- und Chiffretext-Alphabet: $\mathcal{P} = \mathcal{C} = \{0, 1\}^{lm}$
- Schlüsselplan als Teilmenge aller mögl. Schlüssel:

$$\mathcal{K} \subseteq \left(\{0, 1\}^{lm}\right)^{Nr+1}$$

Algorithmus: Substitutions-Permutations Netzwerk

Algorithmus

$w^0 \leftarrow x$

for $r \leftarrow 1$ **to** $Nr - 1$

do $\begin{cases} u^r \leftarrow w^{r-1} \otimes K^r \\ \textbf{for } i \leftarrow 1 \textbf{ to } m \\ \quad \textbf{do } v^r_{(i)} \leftarrow \pi_S\left(u^r_{(i)}\right) \\ w^r \leftarrow \left(v^r_{\pi_P(1)}, ..., v^r_{\pi_P(lm)}\right) \end{cases}$

$w^{Nr} \leftarrow w^{Nr-1} \otimes K^{Nr}$

for $i \leftarrow 1$ **to** m

do $v_i^{Nr} \leftarrow \pi_S\left(u^{Nr}_{(i)}\right)$

$y \leftarrow v^{Nr} \otimes K^{Nr+1}$

output (y)

- Klartext einlesen
- 1. bis Nr-te Runde
- Bit-Add. des Unterschlüssels
- Für alle Blöcke:
- substituieren
- permutieren
- Bit-Add. vorletzer Unterschlüssel
- Für alle Blöcke:
- substituieren
- Bit-Add. letzter Unterschlüssel
- Geheimtextausgabe

Beispiel: Substitutions-Permutations Netzwerk

- Substitutionsvorschrift:

z	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$\pi_S(z)$	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

- Permutationsvorschrift:

z	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\pi_P(z)$	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

- Aus dem 32-bit Hauptschlüssel $K = (k, ..., k_{32}) \in \{0,1\}^{32}$ werden 16-bit Unterschlüssel K^r, nach k_{4r-3}, generiert.

$$
\begin{array}{lcccccccc}
K = & 0011 & 1010 & 1001 & 0100 & 1101 & 0110 & 0011 & 1111 \\
K^1 = & 0011 & 1010 & 1001 & 0100 & & & & \\
K^2 = & & 1010 & 1001 & 0100 & 1101 & & & \\
K^3 = & & & 1001 & 0100 & 1101 & 0110 & & \\
K^4 = & & & & 0100 & 1101 & 0110 & 0011 & \\
K^5 = & & & & & 1101 & 0110 & 0011 & 1111
\end{array}
$$

Beispiel: Substitutions-Permutations Netzwerk

- Substitutionsvorschrift:

z	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$\pi_S(z)$	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

- Permutationsvorschrift:

z	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\pi_P(z)$	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

- Aus dem 32-bit Hauptschlüssel $K = (k, ..., k_{32}) \in \{0,1\}^{32}$ werden 16-bit Unterschlüssel K^r, nach k_{4r-3}, generiert.

$K =$ 0011 1010 1001 0100 1101 0110 0011 1111
$K^1 =$ 0011 1010 1001 0100
$K^2 =$ 1010 1001 0100 1101
$K^3 =$ 1001 0100 1101 0110
$K^4 =$ 0100 1101 0110 0011
$K^5 =$ 1101 0110 0011 1111

Beispiel: Substitutions-Permutations Netzwerk

- Substitutionsvorschrift:

z	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$\pi_S(z)$	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

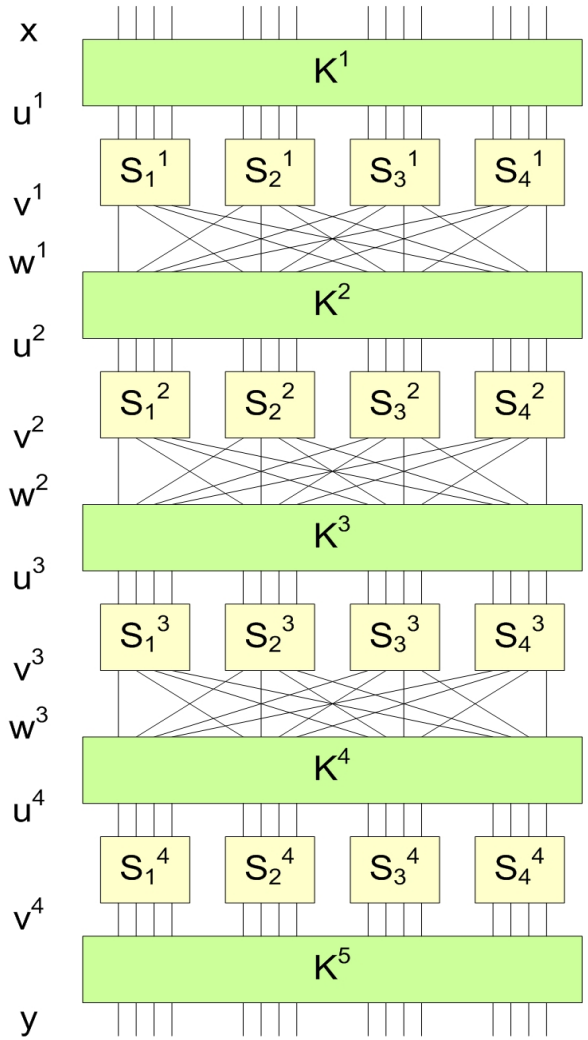
- Permutationsvorschrift:

z	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\pi_P(z)$	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

- Aus dem 32-bit Hauptschlüssel $K = (k, ..., k_{32}) \in \{0,1\}^{32}$ werden 16-bit Unterschlüssel K^r, nach k_{4r-3}, generiert.

$K =$	0011	1010	1001	0100	1101	0110	0011	1111
$K^1 =$	0011	1010	1001	0100				
$K^2 =$		1010	1001	0100	1101			
$K^3 =$			1001	0100	1101	0110		
$K^4 =$				0100	1101	0110	0011	
$K^5 =$					1101	0110	0011	1111

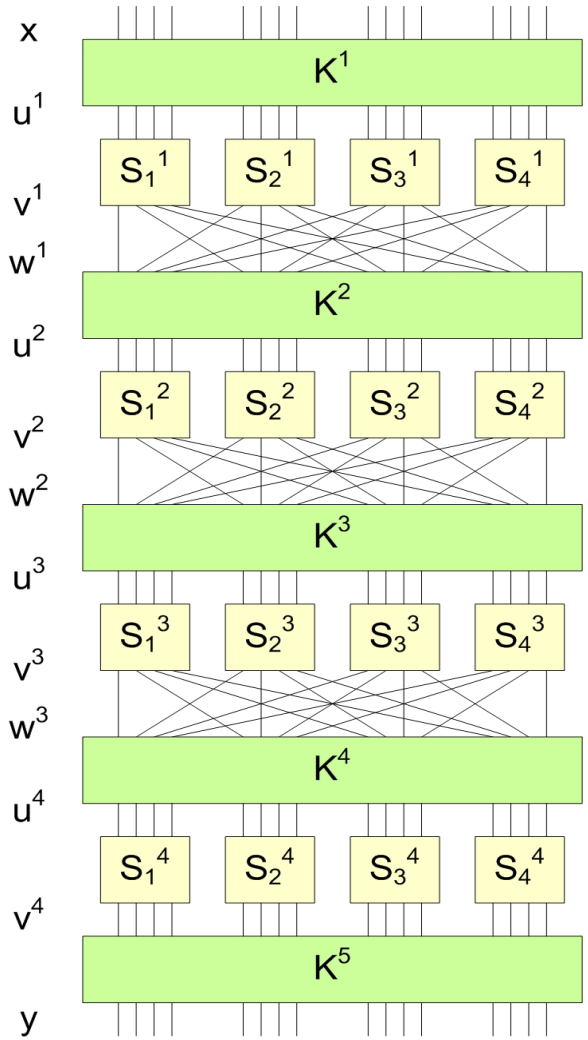
Beispielablaufbild: Substitutions-Permutations Netzwerk



Fazit

- Kombination einfacher Verschlüsselungsformen
- stärkere Verschlüsselung
- Durch Iterationsschritte deutliche Härtung
- Einfache Implementierung (Soft- und Hardware)
- Einziges Geheimnis ist der Schlüssel (*Es soll so wenig wie möglich geheim gehalten werden.*)
- Die Komplexität des mathematische Beweis steigt
- Die Entropie steigt
- Bildet Grundlage für **DES** und **AES**

Beispielablaufbild: Substitutions-Permutations Netzwerk



Fazit

- Kombination einfacher Verschlüsselungsformen
- stärkere Verschlüsselung
- Durch Iterationsschritte deutliche Härtung
- Einfache Implementierung (Soft- und Hardware)
- Einziges Geheimnis ist der Schlüssel (*Es soll so wenig wie möglich geheim gehalten werden.*)
- Die Komplexität des mathematische Beweis steigt
- Die Entropie steigt
- Bildet Grundlage für **DES** und **AES**

Kryptoanalyse

Was ist Kryptanalyse bzw. Kryptoanalyse?

... bezeichnet im ursprünglichen Sinne die Studie von Methoden und Techniken, um Informationen aus verschlüsselten Texten zu gewinnen. Diese Informationen können sowohl der verwendete Schlüssel wie auch der Orginaltext sein. Heutzutage bezeichnet der Begriff Kryptoanalye allgemeiner die Analyse von kryptographischen Verfahren ... mit dem Ziel, diese entweder zu „brechen“ ..., oder ihre Sicherheit nachzuweisen und zu quantifizieren. ...

Quelle:*http://de.wikipedia.org/wiki/kryptanalyse*

Entwicklung von Kryptoanalyse

Wechselspiel zwischen Kryptographie und Kryptoanalyse

- Kryptographen entwickeln Kryptosysteme
- und Kryptoanalytiker versuchen deren Schwachstellen zu finden und die Verfahren nach Möglichkeit zu berechnen
- gewonnene Erfahrungen fließen nachfolgend in die Konstruktion neuer Kryptosysteme ein,
- daraus wächst tendenziell Resistenz gegen Attacken.

Entwicklung von Kryptoanalyse

Wechselspiel zwischen Kryptographie und Kryptoanalyse

- Kryptographen entwickeln Kryptosysteme
- und Kryptoanalytiker versuchen deren Schwachstellen zu finden und die Verfahren nach Möglichkeit zu berechnen
- gewonnene Erfahrungen fließen nachfolgend in die Konstruktion neuer Kryptosysteme ein,
- daraus wächst tendenziell Resistenz gegen Attacken.

Entwicklung von Kryptoanalyse

Wechselspiel zwischen Kryptographie und Kryptoanalyse

- Kryptographen entwickeln Kryptosysteme
- und Kryptoanalytiker versuchen deren Schwachstellen zu finden und die Verfahren nach Möglichkeit zu berechnen
- gewonnene Erfahrungen fließen nachfolgend in die Konstruktion neuer Kryptosysteme ein,
- daraus wächst tendenziell Resistenz gegen Attacken.

Entwicklung von Kryptoanalyse

Wechselspiel zwischen Kryptographie und Kryptoanalyse

- Kryptographen entwickeln Kryptosysteme
- und Kryptoanalytiker versuchen deren Schwachstellen zu finden und die Verfahren nach Möglichkeit zu berechnen
- gewonnene Erfahrungen fließen nachfolgend in die Konstruktion neuer Kryptosysteme ein,
- daraus wächst tendenziell Resistenz gegen Attacken.

Entwicklung von Kryptoanalyse 2

- kryptographische Systeme können prinzipiel gebrochen werden
- jedoch Verschlüsselung nach folgendem Grundsatz:

„Fundamental Tenet of Cryptography“

Wenn es vielen fähigen Leuten bisher nicht gelungen ist, ein bestimmtes Problem zu lösen, dann wird dieses Problem aller Voraussicht nach auch nicht (so bald) gelöst werden.

„Prinzip von Kerckhoffs“

Die Sicherheit eines Kryptosystems darf nicht von der Geheimhaltung des Algorithmus abhängen. Die Sicherheit gründet sich ausschließlich auf die Geheimhaltung des Schlüssels.

Entwicklung von Kryptoanalyse 2

- kryptographische Systeme können prinzipiel gebrochen werden
- jedoch Verschlüsselung nach folgendem Grundsatz:

„Fundamental Tenet of Cryptography“

Wenn es vielen fähigen Leuten bisher nicht gelungen ist, ein bestimmtes Problem zu lösen, dann wird dieses Problem aller Voraussicht nach auch nicht (so bald) gelöst werden.

„Prinzip von Kerckhoffs“

Die Sicherheit eines Kryptosystems darf nicht von der Geheimhaltung des Algorithmus abhängen. Die Sicherheit gründet sich ausschließlich auf die Geheimhaltung des Schlüssels.

Entwicklung von Kryptoanalyse 2

- kryptographische Systeme können prinzipiel gebrochen werden
- jedoch Verschlüsselung nach folgendem Grundsatz:

„Fundamental Tenet of Cryptography“

Wenn es vielen fähigen Leuten bisher nicht gelungen ist, ein bestimmtes Problem zu lösen, dann wird dieses Problem aller Voraussicht nach auch nicht (so bald) gelöst werden.

„Prinzip von Kerckhoffs“

Die Sicherheit eines Kryptosystems darf nicht von der Geheimhaltung des Algorithmus abhängen. Die Sicherheit gründet sich ausschließlich auf die Geheimhaltung des Schlüssels.

Allgemeine Ansätze der Kryptoanalyse

Exhaustion (vollständige Schlüsselsuche)

Hier testet man die gesamte Menge der Schlüssel, um den jeweils verwendeten Key herauszufinden. Sie ist (zumindest theoretisch) immer anwendbar, da sie keine Spezifika der Chiffre bzw. der Daten ausnutzt. (Brute-Force-Attacke)

Strukturanalyse - Algebraischer Angriff

Das Ziel besteht darin, unter Ausnutzung der spezifischen Struktur eines Kryptosystems effiziente Algorithmen zu entwicklen, die ein Berechnen des betreffenden Verfahrens gestatten. (Nutzung unterschiedlicher Hilfsmittel aus der (vorwiegend diskreten) Mathematik)
Erschwert bzw. verhindert wird dies durch die Nichtlinearität der Chiffre.

Statische Angriffe auf versteckte Linearität:

Hierbei werden bekannte statistische Eigenschaften der Klartexte (z. B. Häufigkeiten von Zeichen einer Zeichengruppe) ausgenutzt, um aus der statistischen Analyse des Chiffrats Rückschlüsse auf den Klartext zu ziehen.

- Lineare Kryptoanalyse,
- Differenzielle Kryptoanalyse,
- bzw. Verallgemeinerungen und Mischformen

Trial and Error

Im Gegensatz zur vollständigen Suche werden nur Elemente der im Ergebnis einer vorangegangenen Strukturanalyse des Kryptosystems stark eingeschränkten Menge der möglichen Schlüssel getestet.

Allgemeine Ansätze der Kryptoanalyse

Exhaustion (vollständige Schlüsselsuche)

Hier testet man die gesamte Menge der Schlüssel, um den jeweils verwendeten Key herauszufinden. Sie ist (zumindest theoretisch) immer anwendbar, da sie keine Spezifika der Chiffre bzw. der Daten ausnutzt. (Brute-Force-Attacke)

Strukturanalyse - Algebraischer Angriff

Das Ziel besteht darin, unter Ausnutzung der spezifischen Struktur eines Kryptosystems effiziente Algorithmen zu entwicklen, die ein Berechnen des betreffenden Verfahrens gestatten. (Nutzung unterschiedlicher Hilfsmittel aus der (vorwiegend diskreten) Mathematik)
Erschwert bzw. verhindert wird dies durch die Nichtlinearität der Chiffre.

Statische Angriffe auf versteckte Linearität:

Hierbei werden bekannte statistische Eigenschaften der Klartexte (z. B. Häufigkeiten von Zeichen einer Zeichengruppe) ausgenutzt, um aus der statistischen Analyse des Chiffrats Rückschlüsse auf den Klartext zu ziehen.

- Lineare Kryptoanalyse,
- Differenzielle Kryptoanalyse,
- bzw. Verallgemeinerungen und Mischformen

Trial and Error

Im Gegensatz zur vollständigen Suche werden nur Elemente der im Ergebnis einer vorangegangenen Strukturanalyse des Kryptosystems stark eingeschränkten Menge der möglichen Schlüssel getestet.

Allgemeine Ansätze der Kryptoanalyse

Exhaustion (vollständige Schlüsselsuche)

Hier testet man die gesamte Menge der Schlüssel, um den jeweils verwendeten Key herauszufinden. Sie ist (zumindest theoretisch) immer anwendbar, da sie keine Spezifika der Chiffre bzw. der Daten ausnutzt. (Brute-Force-Attacke)

Strukturanalyse - Algebraischer Angriff

Das Ziel besteht darin, unter Ausnutzung der spezifischen Struktur eines Kryptosystems effiziente Algorithmen zu entwicklen, die ein Berechnen des betreffenden Verfahrens gestatten. (Nutzung unterschiedlicher Hilfsmittel aus der (vorwiegend diskreten) Mathematik)
Erschwert bzw. verhindert wird dies durch die Nichtlinearität der Chiffre.

Statische Angriffe auf versteckte Linearität:

Hierbei werden bekannte statistische Eigenschaften der Klartexte (z. B. Häufigkeiten von Zeichen einer Zeichengruppe) ausgenutzt, um aus der statistischen Analyse des Chiffrats Rückschlüsse auf den Klartext zu ziehen.

- Lineare Kryptoanalyse,
- Differenzielle Kryptoanalyse,
- bzw. Verallgemeinerungen und Mischformen

Trial and Error

Im Gegensatz zur vollständigen Suche werden nur Elemente der im Ergebnis einer vorangegangenen Strukturanalyse des Kryptosystems stark eingeschränkten Menge der möglichen Schlüssel getestet.

Allgemeine Ansätze der Kryptoanalyse

Exhaustion (vollständige Schlüsselsuche)

Hier testet man die gesamte Menge der Schlüssel, um den jeweils verwendeten Key herauszufinden. Sie ist (zumindest theoretisch) immer anwendbar, da sie keine Spezifika der Chiffre bzw. der Daten ausnutzt. (Brute-Force-Attacke)

Strukturanalyse - Algebraischer Angriff

Das Ziel besteht darin, unter Ausnutzung der spezifischen Struktur eines Kryptosystems effiziente Algorithmen zu entwicklen, die ein Berechnen des betreffenden Verfahrens gestatten. (Nutzung unterschiedlicher Hilfsmittel aus der (vorwiegend diskreten) Mathematik)
Erschwert bzw. verhindert wird dies durch die Nichtlinearität der Chiffre.

Statische Angriffe auf versteckte Linearität:

Hierbei werden bekannte statistische Eigenschaften der Klartexte (z. B. Häufigkeiten von Zeichen einer Zeichengruppe) ausgenutzt, um aus der statistischen Analyse des Chiffrats Rückschlüsse auf den Klartext zu ziehen.

- Lineare Kryptoanalyse,
- Differenzielle Kryptoanalyse,
- bzw. Verallgemeinerungen und Mischformen

Trial and Error

Im Gegensatz zur vollständigen Suche werden nur Elemente der im Ergebnis einer vorangegangenen Strukturanalyse des Kryptosystems stark eingeschränkten Menge der möglichen Schlüssel getestet.

Elementare kryptoanalytischer Attacken

Ciphertext-only Attack:
Kryptoanalytiker ...

- ... verfügt über mehrere Geheimtextnachrichten (mit dem selben Algorithmus verschlüsselt)
- ... besitz einer groben Vorstellung der Struktur des Klartextes zur Verifikation des gesuchten Schlüssels

Known-plaintext Attack:
Kryptoanalytiker ...

- ... verfügt über einige Geheimtextnachrichten mit zugehörigen Klartextteilen

Chosen-plaintext Attack:
Kryptoanalytiker ...

- ... ist in der Lage sich seinen eigenen Klartext mit einem bestimmten ihm unbekannten Schlüssel verschlüsseln zu lassen
- ... kennt dadurch Paare von Geheim- und zugehörigem Klartext

Elementare kryptoanalytischer Attacken

Ciphertext-only Attack:
Kryptoanalytiker ...

- ... verfügt über mehrere Geheimtextnachrichten (mit dem selben Algorithmus verschlüsselt)
- ... besitz einer groben Vorstellung der Struktur des Klartextes zur Verifikation des gesuchten Schlüssels

Known-plaintext Attack:
Kryptoanalytiker ...

- ... verfügt über einige Geheimtextnachrichten mit zugehörigen Klartextteilen

Chosen-plaintext Attack:
Kryptoanalytiker ...

- ... ist in der Lage sich seinen eigenen Klartext mit einem bestimmten ihm unbekannten Schlüssel verschlüsseln zu lassen
- ... kennt dadurch Paare von Geheim- und zugehörigem Klartext

Elementare kryptoanalytischer Attacken

Ciphertext-only Attack:
Kryptoanalytiker ...

- ... verfügt über mehrere Geheimtextnachrichten (mit dem selben Algorithmus verschlüsselt)
- ... besitz einer groben Vorstellung der Struktur des Klartextes zur Verifikation des gesuchten Schlüssels

Known-plaintext Attack:
Kryptoanalytiker ...

- ... verfügt über einige Geheimtextnachrichten mit zugehörigen Klartextteilen

Chosen-plaintext Attack:
Kryptoanalytiker ...

- ... ist in der Lage sich seinen eigenen Klartext mit einem bestimmten ihm unbekannten Schlüssel verschlüsseln zu lassen
- ... kennt dadurch Paare von Geheim- und zugehörigem Klartext

Idee der Lineare Kryptoanalyse

Idee:

- Bit-Block Chiffre: $F : \mathbb{F}_2^n \times \mathbb{F}_2^l \longrightarrow \mathbb{F}_2^n$
- Linearformen: $\alpha : \mathbb{F}_2^n \times \mathbb{F}_2^l \longrightarrow \mathbb{F}_2$, $\beta : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$
- Aussagen über die Wahrscheinlichkeit und das Potential der linearen Relation (α, β) ableiten

Phasen:

1. Sammeln: n Klartext- Geheimtextpaare
2. Auszählung: Potential der linearen Relation wird berechnet
3. Mehrheitsentscheidung: Auswahl der linearen Relation mit dem größten Potential

Folgerung

- Verkleinerung des Schlüsselraums (einige Bits entschlüsselt)
- Exhaustion

Je größer die Dimension des Suchraums, desto mehr Klar- Geheimtextpaare (N) werden benötigt.

Idee der Lineare Kryptoanalyse

Idee:

- Bit-Block Chiffre: $F : \mathbb{F}_2^n \times \mathbb{F}_2^l \longrightarrow \mathbb{F}_2^n$
- Linearformen: $\alpha : \mathbb{F}_2^n \times \mathbb{F}_2^l \longrightarrow \mathbb{F}_2$, $\beta : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$
- Aussagen über die Wahrscheinlichkeit und das Potential der linearen Relation (α, β) ableiten

Phasen:

1. Sammeln: n Klartext- Geheimtextpaare
2. Auszählung: Potential der linearen Relation wird berechnet
3. Mehrheitsentscheidung: Auswahl der linearen Relation mit dem größten Potential

Folgerung

- Verkleinerung des Schlüsselraums (einige Bits entschlüsselt)
- Exhaustion

Je größer die Dimension des Suchraums, desto mehr Klar- Geheimtextpaare (N) werden benötigt.

Idee der Differentielle Kryptoanalyse

Phasen:

1. Analog zu Linearen Kryptoanalyse werden Approximationen aus der linearen Strukturen abgearbeitet
2. Berechnung eines Differenzenvektors aus Klartext-Geheimtext-Paar vor jeder Runde
3. Suchen von sich wiederholenden Differenzvektoren in den Folgerunden des iterativen Chiffres (*differenzieller Pfad bzw. Charakteristik*)
4. Berechnung des Potentials eines differentiellen Pfades als Produkt der Potentiale der Einzelschritte
5. Der beste (dominanteste) differenzielle Pfad hat auch das beste Potential
6. Herleitung von Schlüsselbits aus der Wahrscheinlichkeit des Potentials
7. Restliche Ermittlung durch Exhaustion

Idee der Differentielle Kryptoanalyse

Phasen:

1. Analog zu Linearen Kryptoanalyse werden Approximationen aus der linearen Strukturen abgearbeitet
2. Berechnung eines Differenzenvektors aus Klartext-Geheimtext-Paar vor jeder Runde
3. Suchen von sich wiederholenden Differenzvektoren in den Folgerunden des iterativen Chiffres (*differenzieller Pfad bzw. Charakteristik*)
4. Berechnung des Potentials eines differentiellen Pfades als Produkt der Potentiale der Einzelschritte
5. Der beste (dominanteste) differenzielle Pfad hat auch das beste Potential
6. Herleitung von Schlüsselbits aus der Wahrscheinlichkeit des Potentials
7. Restliche Ermittlung durch Exhaustion

Idee der Differentielle Kryptoanalyse

Phasen:

1. Analog zu Linearen Kryptoanalyse werden Approximationen aus der linearen Strukturen abgearbeitet
2. Berechnung eines Differenzenvektors aus Klartext-Geheimtext-Paar vor jeder Runde
3. Suchen von sich wiederholenden Differenzvektoren in den Folgerunden des iterativen Chiffres (*differenzieller Pfad bzw. Charakteristik*)
4. Berechnung des Potentials eines differentiellen Pfades als Produkt der Potentiale der Einzelschritte
5. Der beste (dominanteste) differenzielle Pfad hat auch das beste Potential
6. Herleitung von Schlüsselbits aus der Wahrscheinlichkeit des Potentials
7. Restliche Ermittlung durch Exhaustion

Idee der Differentielle Kryptoanalyse

Phasen:

1. Analog zu Linearen Kryptoanalyse werden Approximationen aus der linearen Strukturen abgearbeitet
2. Berechnung eines Differenzenvektors aus Klartext-Geheimtext-Paar vor jeder Runde
3. Suchen von sich wiederholenden Differenzvektoren in den Folgerunden des iterativen Chiffres (*differenzieller Pfad bzw. Charakteristik*)
4. Berechnung des Potentials eines differentiellen Pfades als Produkt der Potentiale der Einzelschritte
5. Der beste (dominanteste) differenzielle Pfad hat auch das beste Potential
6. Herleitung von Schlüsselbits aus der Wahrscheinlichkeit des Potentials
7. Restliche Ermittlung durch Exhaustion

Idee der Differentielle Kryptoanalyse

Phasen:

1. Analog zu Linearen Kryptoanalyse werden Approximationen aus der linearen Strukturen abgearbeitet
2. Berechnung eines Differenzenvektors aus Klartext-Geheimtext-Paar vor jeder Runde
3. Suchen von sich wiederholenden Differenzvektoren in den Folgerunden des iterativen Chiffres (*differenzieller Pfad bzw. Charakteristik*)
4. Berechnung des Potentials eines differentiellen Pfades als Produkt der Potentiale der Einzelschritte
5. Der beste (dominanteste) differenzielle Pfad hat auch das beste Potential
6. Herleitung von Schlüsselbits aus der Wahrscheinlichkeit des Potentials
7. Restliche Ermittlung durch Exhaustion

Idee der Differentielle Kryptoanalyse

Phasen:

1. Analog zu Linearen Kryptoanalyse werden Approximationen aus der linearen Strukturen abgearbeitet
2. Berechnung eines Differenzenvektors aus Klartext-Geheimtext-Paar vor jeder Runde
3. Suchen von sich wiederholenden Differenzvektoren in den Folgerunden des iterativen Chiffres (*differenzieller Pfad bzw. Charakteristik*)
4. Berechnung des Potentials eines differentiellen Pfades als Produkt der Potentiale der Einzelschritte
5. Der beste (dominanteste) differenzielle Pfad hat auch das beste Potential
6. Herleitung von Schlüsselbits aus der Wahrscheinlichkeit des Potentials
7. Restliche Ermittlung durch Exhaustion

Idee der Differentielle Kryptoanalyse

Phasen:

1. Analog zu Linearen Kryptoanalyse werden Approximationen aus der linearen Strukturen abgearbeitet
2. Berechnung eines Differenzenvektors aus Klartext-Geheimtext-Paar vor jeder Runde
3. Suchen von sich wiederholenden Differenzvektoren in den Folgerunden des iterativen Chiffres (*differenzieller Pfad bzw. Charakteristik*)
4. Berechnung des Potentials eines differentiellen Pfades als Produkt der Potentiale der Einzelschritte
5. Der beste (dominanteste) differenzielle Pfad hat auch das beste Potential
6. Herleitung von Schlüsselbits aus der Wahrscheinlichkeit des Potentials
7. Restliche Ermittlung durch Exhaustion

Geschichtliche Entwicklung des DES

1973 Ausschreibung zur Standardisierung eines kryptografischen Algorithmus durch das „National Bureau of Standard (NBS)“

1974 IBM: symmetrischer Blockchiffre, beruht auf *Lucifer-Algorithmus*

1975 Modifizierung durch NBS (heute: *NIST*), IBM und NSA: Schlüssellänge von 128 auf 56 Bit, veränderte S-Boxen

1977 Veröffentlichung des Standard

Geschichtliche Entwicklung des DES

1973 Ausschreibung zur Standardisierung eines kryptografischen Algorithmus durch das „National Bureau of Standard (NBS)“

1974 IBM: symmetrischer Blockchiffre, beruht auf *Lucifer-Algorithmus*

1975 Modifizierung durch NBS (heute: *NIST*), IBM und NSA: Schlüssellänge von 128 auf 56 Bit, veränderte S-Boxen

1977 Veröffentlichung des Standard

Geschichtliche Entwicklung des DES

1973 Ausschreibung zur Standardisierung eines kryptografischen Algorithmus durch das „National Bureau of Standard (NBS)“

1974 IBM: symmetrischer Blockchiffre, beruht auf *Lucifer-Algorithmus*

1975 Modifizierung durch NBS (heute: *NIST*), IBM und NSA: Schlüssellänge von 128 auf 56 Bit, veränderte S-Boxen

1977 Veröffentlichung des Standard

Geschichtliche Entwicklung des DES

1973 Ausschreibung zur Standardisierung eines kryptografischen Algorithmus durch das „National Bureau of Standard (NBS)“

1974 IBM: symmetrischer Blockchiffre, beruht auf *Lucifer-Algorithmus*

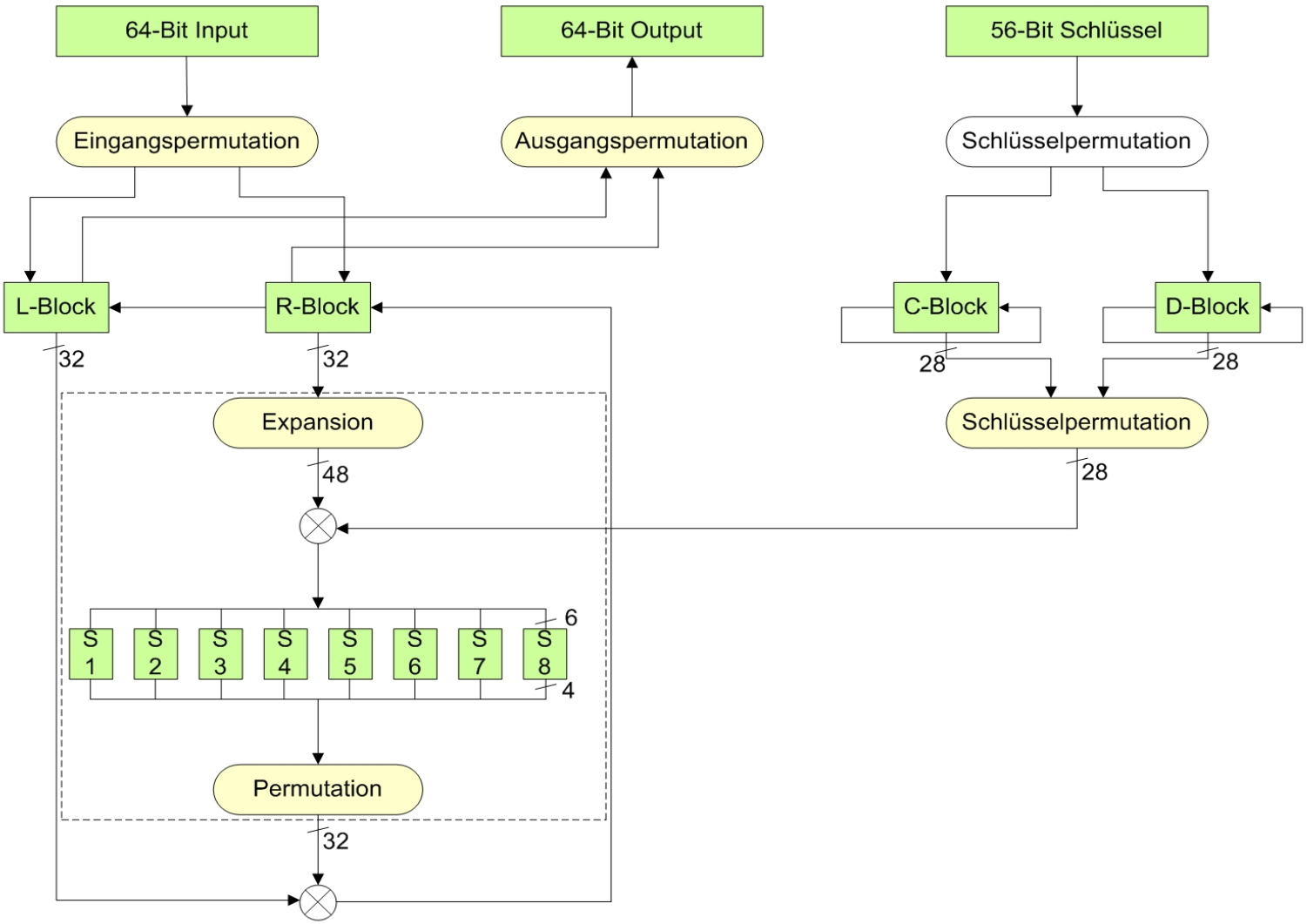
1975 Modifizierung durch NBS (heute: *NIST*), IBM und NSA: Schlüssellänge von 128 auf 56 Bit, veränderte S-Boxen

1977 Veröffentlichung des Standard

Allgemeine Eigenschaften DES

- Symmetrischer Blockchiffre
- 64 Bit Blockweise Chiffrierierung mittels des geheimem Schlüssels (56 Bit + 8 Bit Parität $\rightarrow 2^{56}$ Schlüssel)
- 16 Verschlüsselungsrunden (Substitution und Permutation)

Detailierter Aufbau DES



DES Permutationen IP und IP^{-1}

- Permutationen haben kyptologisch **keine** Auswirkungen

Initial Permutation IP L-Block und R-Block

58	50	42	34	26	18	10	2	60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6	64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1	59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5	63	55	47	39	31	23	15	7

Ausgangspermutation IP^{-1}

40	8	48	16	56	24	64	32	39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30	37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28	35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26	33	1	41	9	49	17	57	25

DES Permutationen IP und IP^{-1}

- Permutationen haben kyptologisch **keine** Auswirkungen

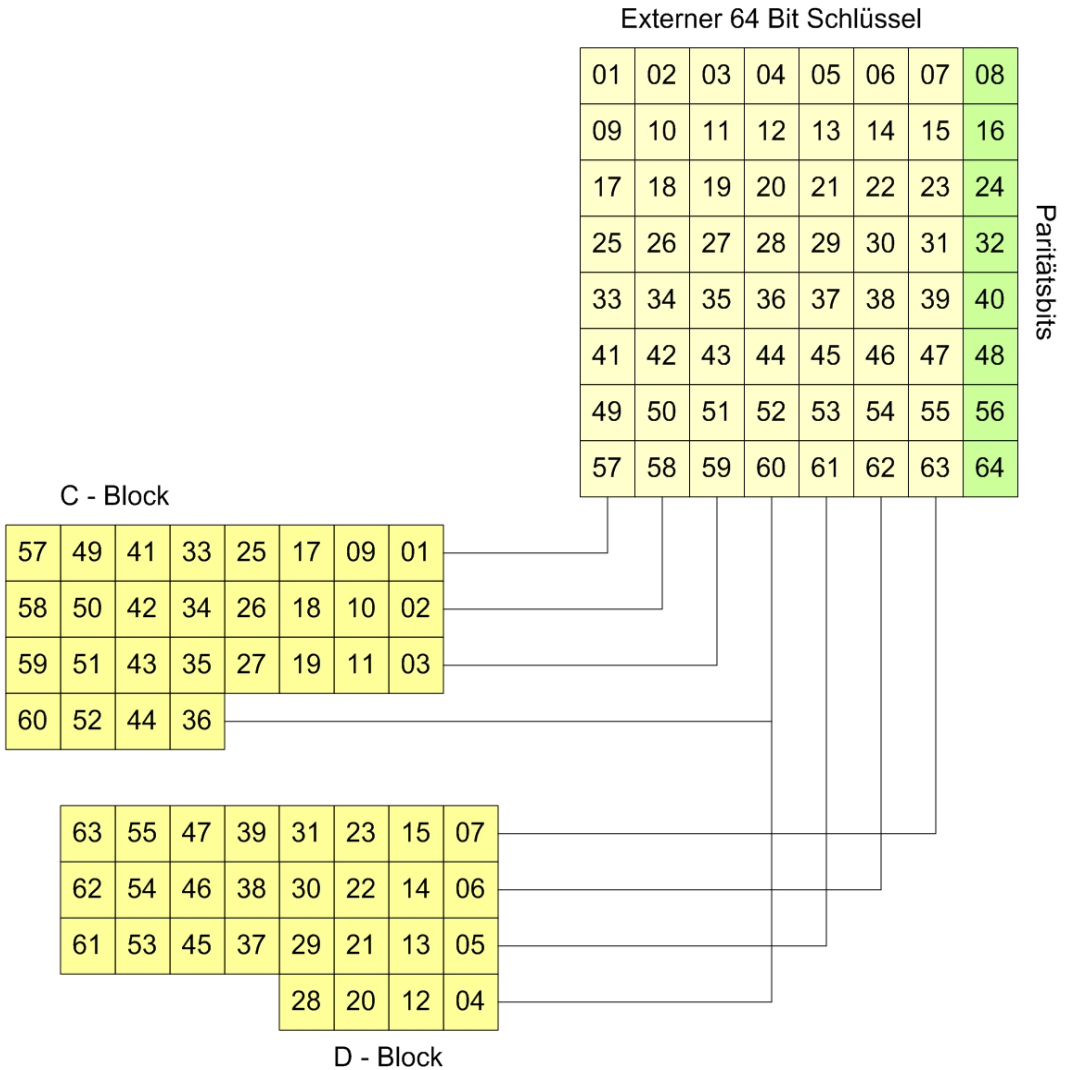
Initial Permutation IP L-Block und R-Block

58	50	42	34	26	18	10	2	60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6	64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1	59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5	63	55	47	39	31	23	15	7

Ausgangspermutation IP^{-1}

40	8	48	16	56	24	64	32	39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30	37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28	35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26	33	1	41	9	49	17	57	25

DES Schlüsselpermutation (Permutated Choice 1)



Bitverschiebung und Schlüsselauswahl (PC - 2)

Zyklische Linksverschiebung der Register C und D

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Permutated Choise 2

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

Bitverschiebung und Schlüsselauswahl (PC - 2)

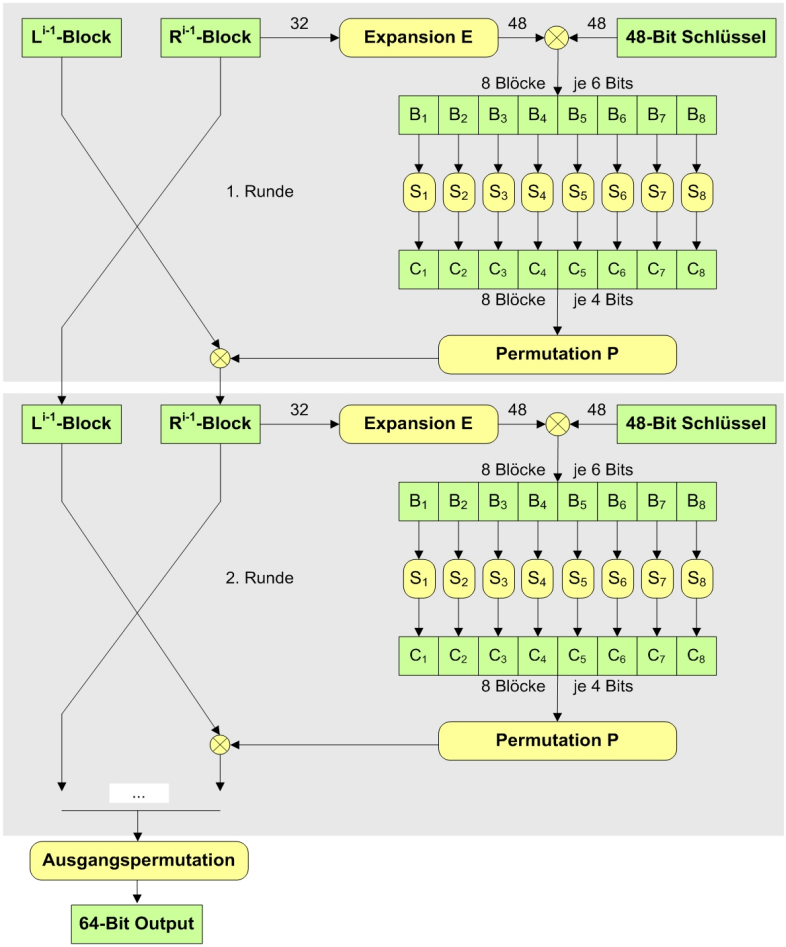
Zyklische Linksverschiebung der Register C und D

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Permutated Choise 2

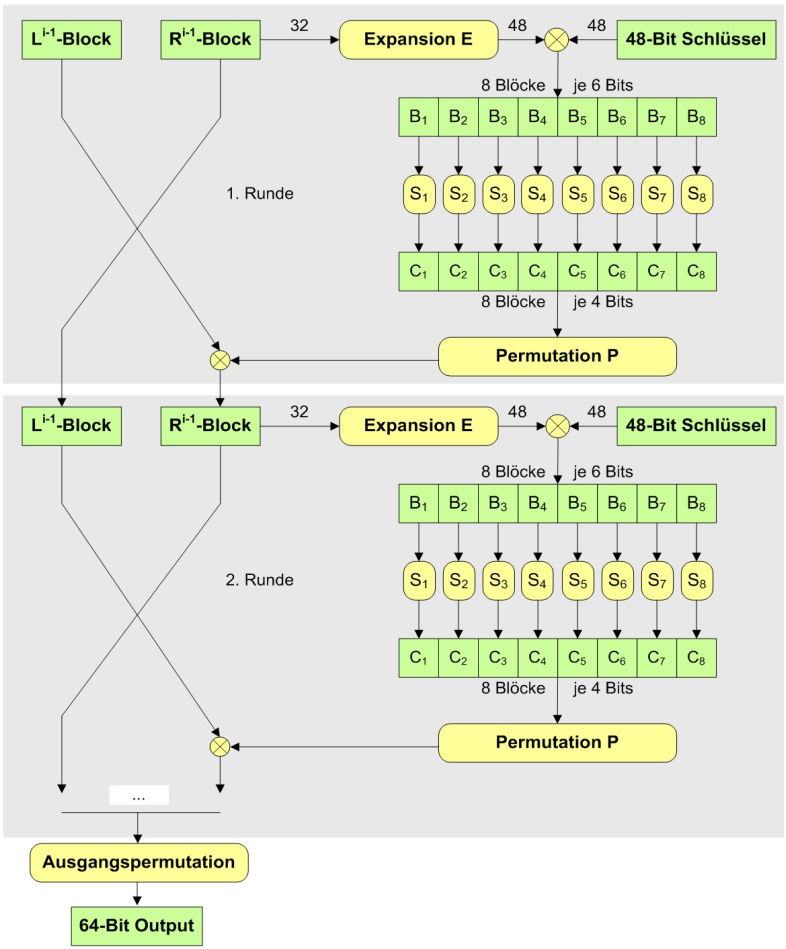
14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

DES Funktion *f*



- Expansion E
- $E(R) \oplus$ 48-Bit Schlüssel
  Ergebnis: $B = B_1 B_2 B_3 B_4 B_5 B_6 B_7 B_8$
- S-Boxen (4 x 16 Matrix)
  $S_j : \{0,1\}^6 \rightarrow \{0,1\}^4$
  $S_j(B_j) = C_j \quad (1 \leq j \leq 8)$
- $C = C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8$
  (8 x 4 Bit)
  $\rightarrow P(C) = f(R) \oplus$ 48 Bit Schlüssel)
- Nach 16 Runden:
  64-Bit Output $= IP^{-1}\left(L^{16} R^{16}\right)$

DES Funktion *f*



- Expansion E
- $E(R) \oplus$ 48-Bit Schlüssel
  Ergebnis: $B = B_1 B_2 B_3 B_4 B_5 B_6 B_7 B_8$
- S-Boxen (4 x 16 Matrix)
  $S_j : \{0,1\}^6 \rightarrow \{0,1\}^4$
  $S_j(B_j) = C_j \quad (1 \leq j \leq 8)$
- $C = C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8$
  (8 x 4 Bit)
  $\rightarrow P(C) = f(R) \oplus$ 48 Bit Schlüssel)
- Nach 16 Runden:
  64-Bit Output $= IP^{-1}\left(L^{16} R^{16}\right)$

DES S-Boxen

- 8 Eingabeblöcke je 6 Bits = 48 Bits
- Eingabe und Ausgabe **nicht linear**, da 6 Bit → 4 Bit

S-Box 1

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

...

S-Box 8

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
3	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

DES S-Boxen

- 8 Eingabeblöcke je 6 Bits = 48 Bits
- Eingabe und Ausgabe **nicht linear**, da 6 Bit → 4 Bit

S-Box 1

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

...

S-Box 8

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
3	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

Permutation in jeder Runde

- Ergebnis der 8 S-Boxen: 32 Bit ($p_1, p_2, p_3, ..., p_{32}$)
- Permutation P: Abbildung von 32 Bits auf 32 Bits

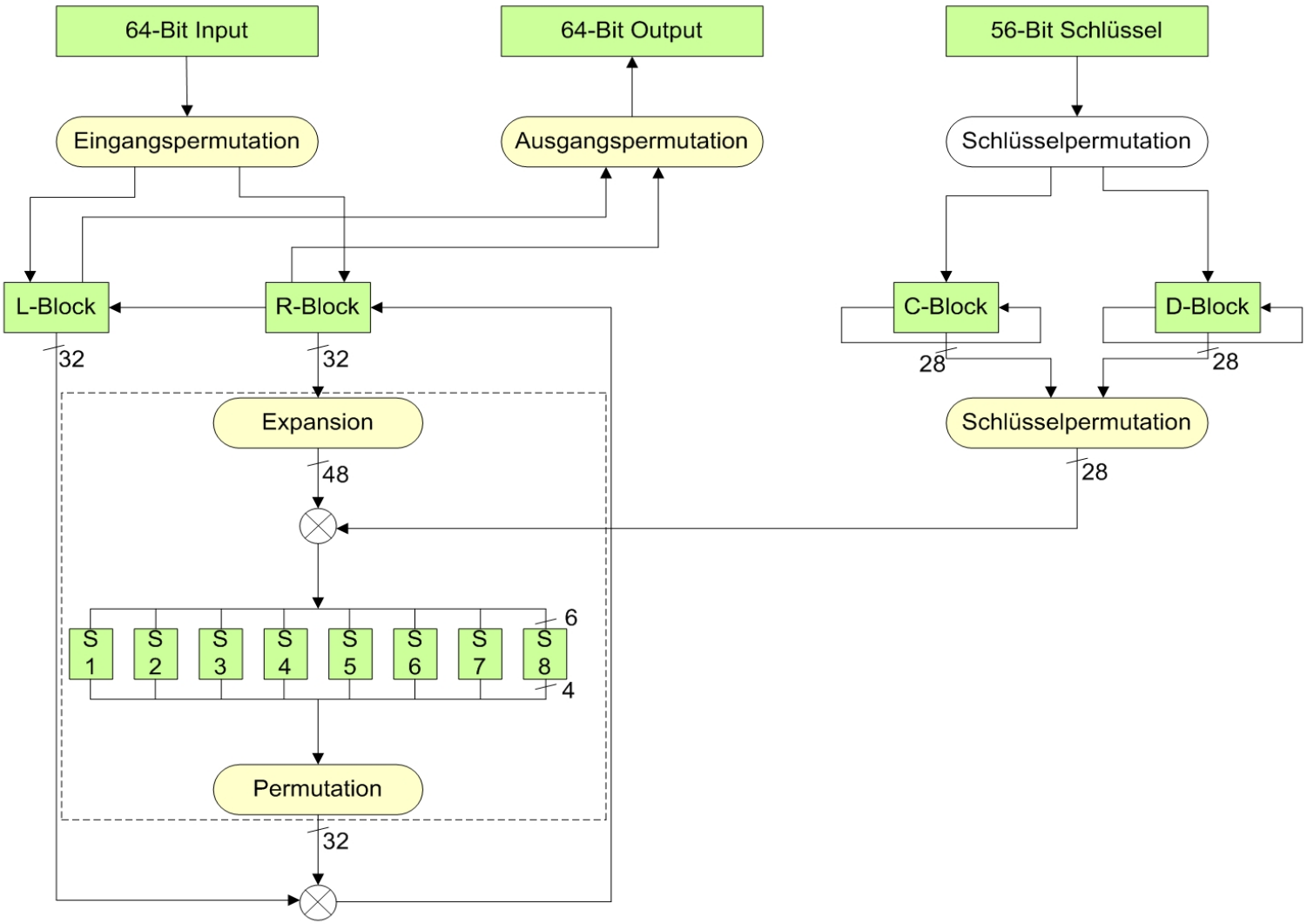
16	7	20	21	29	12	28	17	1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9	19	13	30	6	22	11	4	25

Permutation in jeder Runde

- Ergebnis der 8 S-Boxen: 32 Bit ($p_1, p_2, p_3, ..., p_{32}$)
- Permutation P: Abbildung von 32 Bits auf 32 Bits

16	7	20	21	29	12	28	17	1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9	19	13	30	6	22	11	4	25

Detailierter Aufbau DES (Wiederholung)



DES Entschlüsselung

- Umgekehrte Reihenfolge der 16 Rundenschlüssel
- Permutation und Substitution in umgekehrter Reihenfolge
- Zyklische Rechtsverschiebung der Register C und D durch $PC - 2^{-1}$

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

DES Entschlüsselung

- Umgekehrte Reihenfolge der 16 Rundenschlüssel
- Permutation und Substitution in umgekehrter Reihenfolge
- Zyklische Rechtsverschiebung der Register C und D durch $PC - 2^{-1}$

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Geschichtliche Entwicklung des AES

1990er galt DES mit seinen 56 Bit als nicht mehr sicher (z. B. *Brute-Force-Angriffe*), auch das 3DES mit seinen 112 Bit nicht mehr

1997-1998 Ankündigung und Ausschreibung eines neuen Verschlüsselungsstandards (AES)

1998-1999 15 Kandidaten als Ergebnis der Vorrunde

1999-2000 Die letzten 5 Finalisten (*MARS*, *RC6*, *Rijndael*, *Serpent*, *Twofish*) aus der 1. Entscheidungsrunde stellen sich der technischen Analyse

Okt. 2000 Sieger: Rijndael-Algorithmus
(**Überlegenheit** bei Sicherheitsaspekten in Kombination mit Eigenschaften der **Performance**, **Effiziens**, **Implementierbarkeit** und **Flexibilität**)

Nov. 2000 Verabschiedung des AES (FIPS PUB 197)

Geschichtliche Entwicklung des AES

1990er galt DES mit seinen 56 Bit als nicht mehr sicher (z. B. *Brute-Force-Angriffe*), auch das 3DES mit seinen 112 Bit nicht mehr

1997-1998 Ankündigung und Ausschreibung eines neuen Verschlüsselungsstandards (AES)

1998-1999 15 Kandidaten als Ergebnis der Vorrunde

1999-2000 Die letzten 5 Finalisten (*MARS*, *RC6*, *Rijndael*, *Serpent*, *Twofish*) aus der 1. Entscheidungsrunde stellen sich der technischen Analyse

Okt. 2000 Sieger: Rijndael-Algorithmus
(**Überlegenheit** bei Sicherheitsaspekten in Kombination mit Eigenschaften der **Performance**, **Effiziens**, **Implementierbarkeit** und **Flexibilität**)

Nov. 2000 Verabschiedung des AES (FIPS PUB 197)

Geschichtliche Entwicklung des AES

1990er galt DES mit seinen 56 Bit als nicht mehr sicher (z. B. *Brute-Force-Angriffe*), auch das 3DES mit seinen 112 Bit nicht mehr

1997-1998 Ankündigung und Ausschreibung eines neuen Verschlüsselungsstandards (AES)

1998-1999 15 Kandidaten als Ergebnis der Vorrunde

1999-2000 Die letzten 5 Finalisten (*MARS*, *RC6*, *Rijndael*, *Serpent*, *Twofish*) aus der 1. Entscheidungsrunde stellen sich der technischen Analyse

Okt. 2000 Sieger: Rijndael-Algorithmus
(**Überlegenheit** bei Sicherheitsaspekten in Kombination mit Eigenschaften der **Performance**, **Effiziens**, **Implementierbarkeit** und **Flexibilität**)

Nov. 2000 Verabschiedung des AES (FIPS PUB 197)

Geschichtliche Entwicklung des AES

1990er galt DES mit seinen 56 Bit als nicht mehr sicher (z. B. *Brute-Force-Angriffe*), auch das 3DES mit seinen 112 Bit nicht mehr

1997-1998 Ankündigung und Ausschreibung eines neuen Verschlüsselungsstandards (AES)

1998-1999 15 Kandidaten als Ergebnis der Vorrunde

1999-2000 Die letzten 5 Finalisten (*MARS*, *RC6*, *Rijndael*, *Serpent*, *Twofish*) aus der 1. Entscheidungsrunde stellen sich der technischen Analyse

Okt. 2000 Sieger: Rijndael-Algorithmus
(**Überlegenheit** bei Sicherheitsaspekten in Kombination mit Eigenschaften der **Performance**, **Effiziens**, **Implementierbarkeit** und **Flexibilität**)

Nov. 2000 Verabschiedung des AES (FIPS PUB 197)

Geschichtliche Entwicklung des AES

1990er galt DES mit seinen 56 Bit als nicht mehr sicher (z. B. *Brute-Force-Angriffe*), auch das 3DES mit seinen 112 Bit nicht mehr

1997-1998 Ankündigung und Ausschreibung eines neuen Verschlüsselungsstandards (AES)

1998-1999 15 Kandidaten als Ergebnis der Vorrunde

1999-2000 Die letzten 5 Finalisten (*MARS, RC6, Rijndael, Serpent, Twofish*) aus der 1. Entscheidungsrunde stellen sich der technischen Analyse

Okt. 2000 Sieger: Rijndael-Algorithmus (**Überlegenheit** bei Sicherheitsaspekten in Kombination mit Eigenschaften der **Performance**, **Effiziens**, **Implementierbarkeit** und **Flexibilität**)

Nov. 2000 Verabschiedung des AES (FIPS PUB 197)

Geschichtliche Entwicklung des AES

1990er galt DES mit seinen 56 Bit als nicht mehr sicher (z. B. *Brute-Force-Angriffe*), auch das 3DES mit seinen 112 Bit nicht mehr

1997-1998 Ankündigung und Ausschreibung eines neuen Verschlüsselungsstandards (AES)

1998-1999 15 Kandidaten als Ergebnis der Vorrunde

1999-2000 Die letzten 5 Finalisten (*MARS, RC6, Rijndael, Serpent, Twofish*) aus der 1. Entscheidungsrunde stellen sich der technischen Analyse

Okt. 2000 Sieger: Rijndael-Algorithmus (**Überlegenheit** bei Sicherheitsaspekten in Kombination mit Eigenschaften der **Performance**, **Effiziens**, **Implementierbarkeit** und **Flexibilität**)

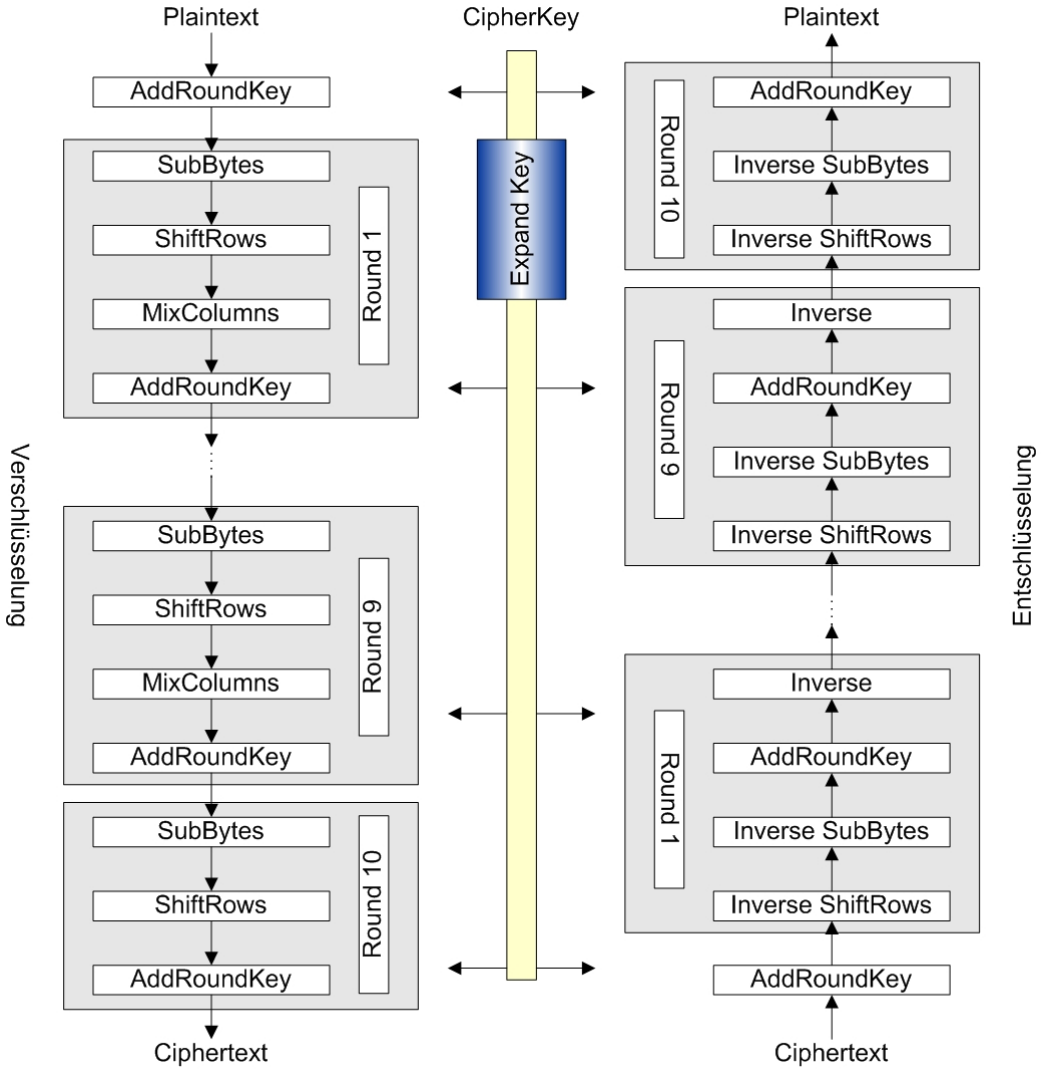
Nov. 2000 Verabschiedung des AES (FIPS PUB 197)

Designkriterien des AES

Designkriterien:

1. muss symmetrischer Schlüssel, und zwar ein Blockchiffre sein
2. muss mindestens 128 Bit lange Blöcke verwenden und Schlüssel von 128, 196 und 256 Bit Länge einsetzen können
3. soll gleichermaßen leicht in Hard- und in Software umzusetzen sein
4. soll in Hard- und Software eine überdurchschnittliche Performance besitzen
5. soll Methoden der Kryptoanalyse wiederstehen können, insbesondere Power- und Timing-Attacken
6. soll nur wenige Ressourcen nutzen müssen
7. es sollen keine Patentansprüche geltend gemacht werden
8. muss weltweit frei ferfügbar sein

AES Aufbau


Plaintext
AddRoundKey
SubBytes
ShiftRows
MixColumns
AddRoundKey
Round 1
SubBytes
ShiftRows
MixColumns
AddRoundKey
Round 9
SubBytes
ShiftRows
AddRoundKey
Round 10
Ciphertext
Verschlüsselung
CipherKey
Expand Key
Plaintext
AddRoundKey
Inverse SubBytes
Inverse ShiftRows
Round 10
Inverse
AddRoundKey
Inverse SubBytes
Inverse ShiftRows
Round 9
Inverse
AddRoundKey
Inverse SubBytes
Inverse ShiftRows
Round 1
AddRoundKey
Ciphertext
Entschlüsselung

AES Schlüsselexpansion

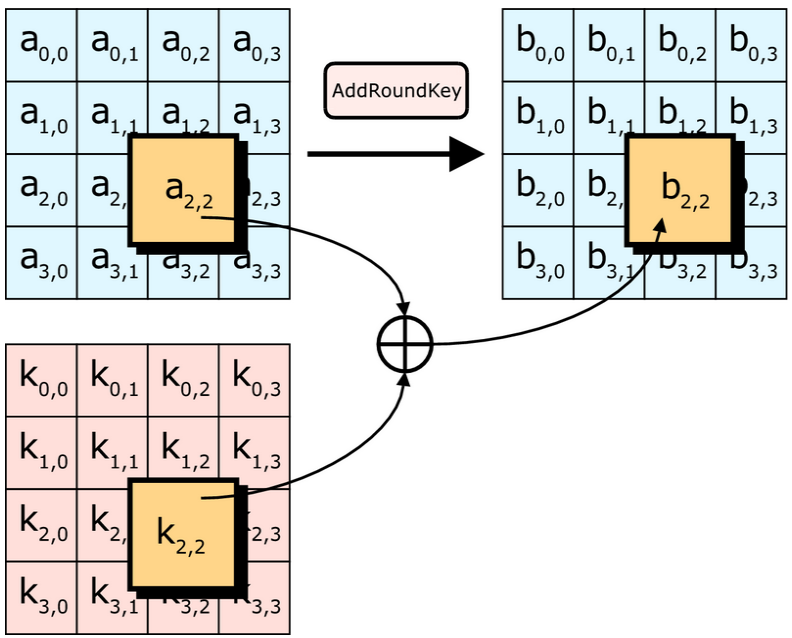
- Benötigt werden $(r + 1)$ Teilschlüssel der Länge b (r..Rundenanzahl und b..Blockgröße)

Expansionsalgorithmus

```
proc KeyExpansion(K, w);
begin external RotWord; SubWord;
RCon[1] := 01000000; RCon[2] := 02000000;
RCon[3] := 04000000; RCon[4] := 08000000;
RCon[5] := 10000000; RCon[6] := 20000000;
RCon[7] := 40000000; RCon[8] := 80000000;
RCon[9] := 1B000000; RCon[10] := 36000000;
for i := 0 to 3 do
   w[i] := (K[4i]; K[4i+1]; K[4i+2]; K[4i+3])
endfor;
for i := 4 to 43 do
  temp := w[i-1];
  if (i mod 4 = 0)
     then temp := SubWord(RotWord(temp)) XOR RCon[i/4];
  w[i] := w[i-4] XOR temp
  endfor;
return (w[0] ... w[43]);
end
```

AES KeyAddition

- $\otimes$ - Verknüpfung zuwischen Datenblock und aktuellem Rundenschlüssel



AES SubBytes

- Substitution durch AES S-Box $\Pi_S : \{0,1\}_8 \rightarrow \{0,1\}_8$
- 1 Byte (2 x 4 Bit) werden auf 1 Byte abgebildet (monoalphabetische Verschlüsselung)

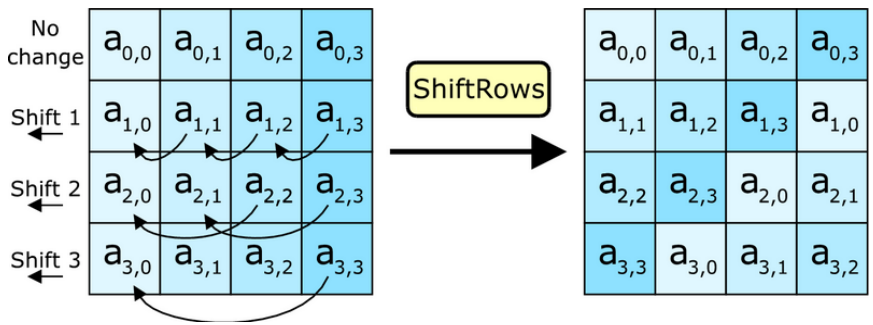
Beispiel $\Pi_S\,(0111\ 0011) = \Pi_S\,(7\ 3) = 8\ F = 1000\ 1111$

	0	·····	2	3	4	5	·····	F
0	63	·····	77	7B	F2	6B	·····	76
⋮	⋮			⋮				⋮
5	53		00	ED	20	FC		CF
6	D0		AA	EB	43	4D		8A
7	51	·····	40	8F	92	9D	·····	D2
8	CD		13	EC	5F	97		73
9	60		4F	DC	22	2A		DB
⋮	⋮			⋮				⋮
F	8C	·····	89	0D	BF	E6	·····	16

AES ShiftRow

- Linksverschiebung der Zeilen um eine bestimmte Anzahl von Spalten.
- Abhängig von Zeilenindex und Blocklänge

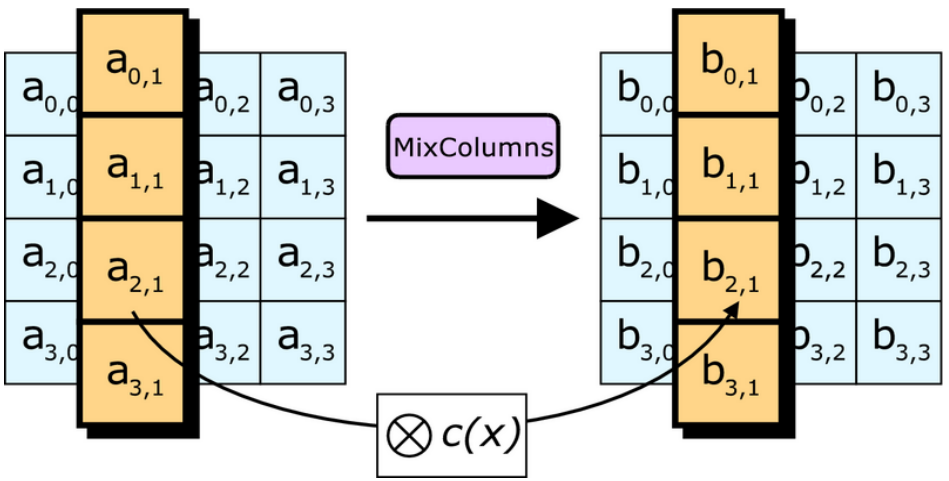
r b	128	192	256
Zeile 0	0	0	0
Zeile 1	1	1	1
Zeile 2	2	2	3
Zeile 3	3	3	4



AES MixColumn

- Multiplikation jeder Zelle mit einer Konstanten und $\otimes$ -Verknüpfung der Ergebnisse
- Ziel: Vermischung der Spaltenelemente zu einer neuen Spalte

Zeile 1	2
Zeile 2	3
Zeile 3	1
Zeile 4	1



AES Entschlüsselung

- Gleicher Ablauf in rückwertiger Abfolge
- Daten und Schlüssel in 2-dim. Tabellen einlesen
- Rundenschlüssel generieren
- Durch XOR-Vershlüsselung unterscheiden sich die meisten Funktionen zum entschlüsseln nicht mit denen zum Verschlüsseln
- Jedoch muss andere S-Box verwendet werden
- Zusätzlich erfolgen die Zeilenverschiebungen in die entgegengesetzte Richtung

Fazit

Vor- und Nachteile:

- Blockcodes sind am weitesten verbreitet und wissenschaftlich untersucht
- Effizient auch bei großen Datenmengen (symmetrische Verschlüsselung)
- AES offene Ausschreibung
- AES und DES sind einfach in Hardware und Software zu implementieren

- DES nicht öffentlich ausgeschrieben
- DES wurde 1997 schon genackt und geht heute binnen weniger Stunden
- AES nur bedingt sicher - Fundamental Tenet of Cryptography
- Fehlende Authentifizierung

Fazit

Vor- und Nachteile:

- Blockcodes sind am weitesten verbreitet und wissenschaftlich untersucht
- Effizient auch bei großen Datenmengen (symmetrische Verschlüsselung)
- AES offene Ausschreibung
- AES und DES sind einfach in Hardware und Software zu implementieren
- DES nicht öffentlich ausgeschrieben
- DES wurde 1997 schon genackt und geht heute binnen weniger Stunden
- AES nur bedingt sicher - Fundamental Tenet of Cryptography
- Fehlende Authentifizierung

Quellen

Douglas R. Stinson: *Cryptography*: Theory and Practice [1. Edition, Prentice Hall 1995]

Douglas R. Stinson: *Cryptography*: Theory and Practice [2. Edition, Chapman & Hall/CRC 2002]

http://csrc.nist.gov/encryption Stand 09.11.2006

http://de.wikipedia.org/wiki/Kryptanalyse Stand 09.11.2006

http://de.wikipedia.org/wiki/Advanced_Encryption_Standard Stand 09.11.2006

http://de.wikipedia.org/wiki/Data_Encryption_Standard Stand 09.11.2006

Seminar Kryptographie Vortragsfolien der vergangen Jahre