Kryptografie verständlich

Ein Fachbuch für
Studierende und Anwender

von
Christof Paar und Jan Pelzl

www.crypto-textbook.com

Kapitel 4

Der Advanced Encryption Standard (AES)

(Version: 1. Dezember 2016)

Übersicht

- Einführung
- Übersicht über den Algorithmus
- Interne Struktur
  - Byte Substitution Schicht
  - Diffusion Schicht
  - Key Addition Schicht
  - Schlüsselfahrplan
- Entschlüsselung
- Praktische Aspekte

Übersicht

- **Einführung**
- Übersicht über den Algorithmus
- Interne Struktur
  - Byte Substitution Schicht
  - Diffusion Schicht
  - Key Addition Schicht
  - Schlüsselfahrplan
- Entschlüsselung
- Praktische Aspekte

AES
Fakten

- AES ist heutzutage die am meisten verwendete symmetrische Chiffre

- Der AES Algorithmus wurde über einen mehrjährigen Auswahlprozess durch die amerikanische NIST (US National Institute of Standards and Technology) bestimmt

- Anforderungen an alle AES Kandidaten:
  - Blockchiffre mit 128 Bit Blockgröße
  - Unterstützung von 3 Schlüssellängen: 128, 192 und 256 bit
  - Sicher ggü. kryptanalytischen Angriffen
  - Effizient in Soft- und Hardware

AES

Historie der AES Entwicklung

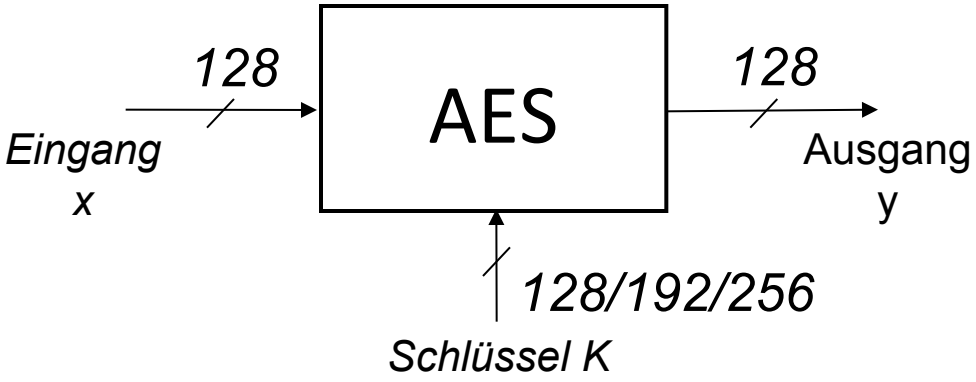
- NIST kündigt im Januar 1997 Auswahlverfahren für neue Blockchiffre an
- 15 Kandidaten wurden im August 1998 akzeptiert
- 5 Finalisten im August 1999 bekannt gegeben:
  - *Mars* – IBM Corporation
  - *RC6* – *RSA* Laboratories
  - *Rijndael* – J. Daemen & V. Rijmen
  - *Serpent* – Eli Biham et al.
  - *Twofish* – B. Schneier et al.
- Im Oktober 2000 wurde *Rijndael* als AES ausgewählt
- AES ist mittlerweile weltweiter Standard

Übersicht

- Einführung
- **Übersicht über den Algorithmus**
- Interne Struktur
  - Byte Substitution Schicht
  - Diffusion Schicht
  - Key Addition Schicht
  - Schlüsselfahrplan
- Entschlüsselung
- Praktische Aspekte

AES
Übersicht

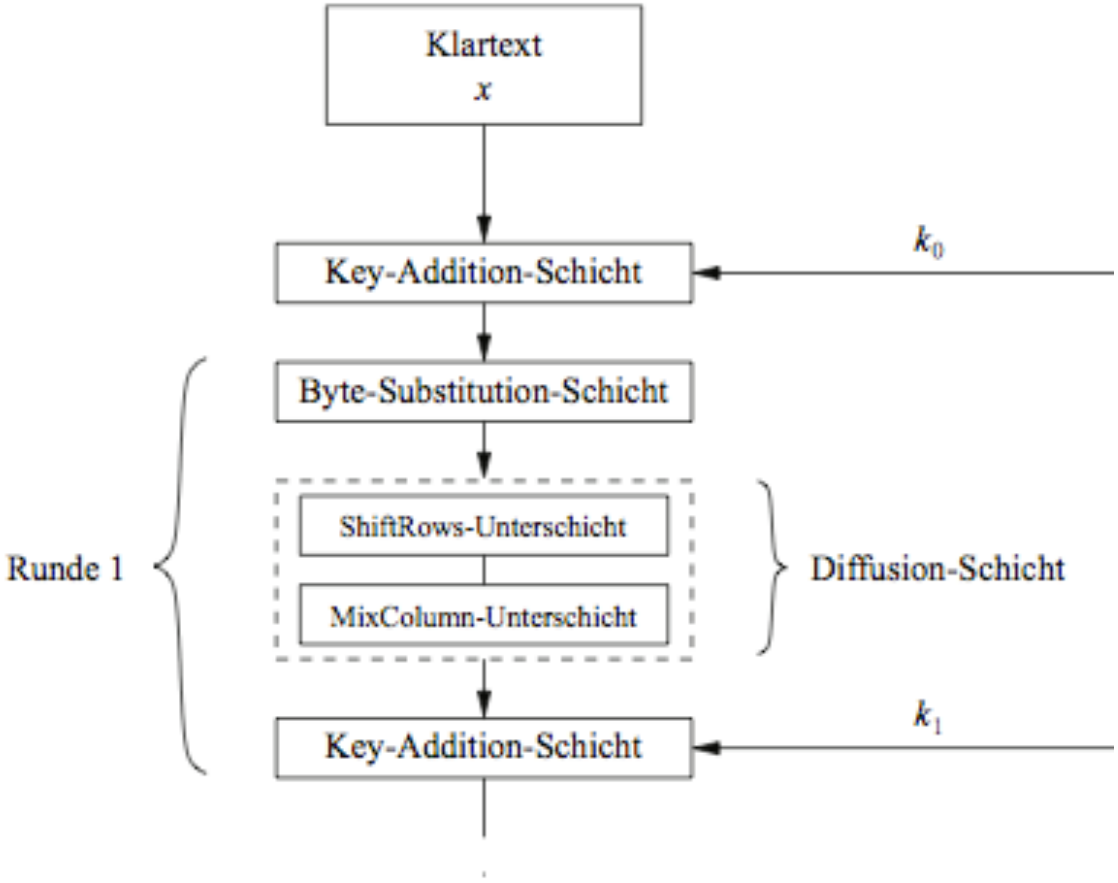
Die Anzahl der Runden (=Iterationen) ist eine Funktion der Schlüssellänge:



Schlüssellänge in Bit	# Runden
128	10
192	12
256	14

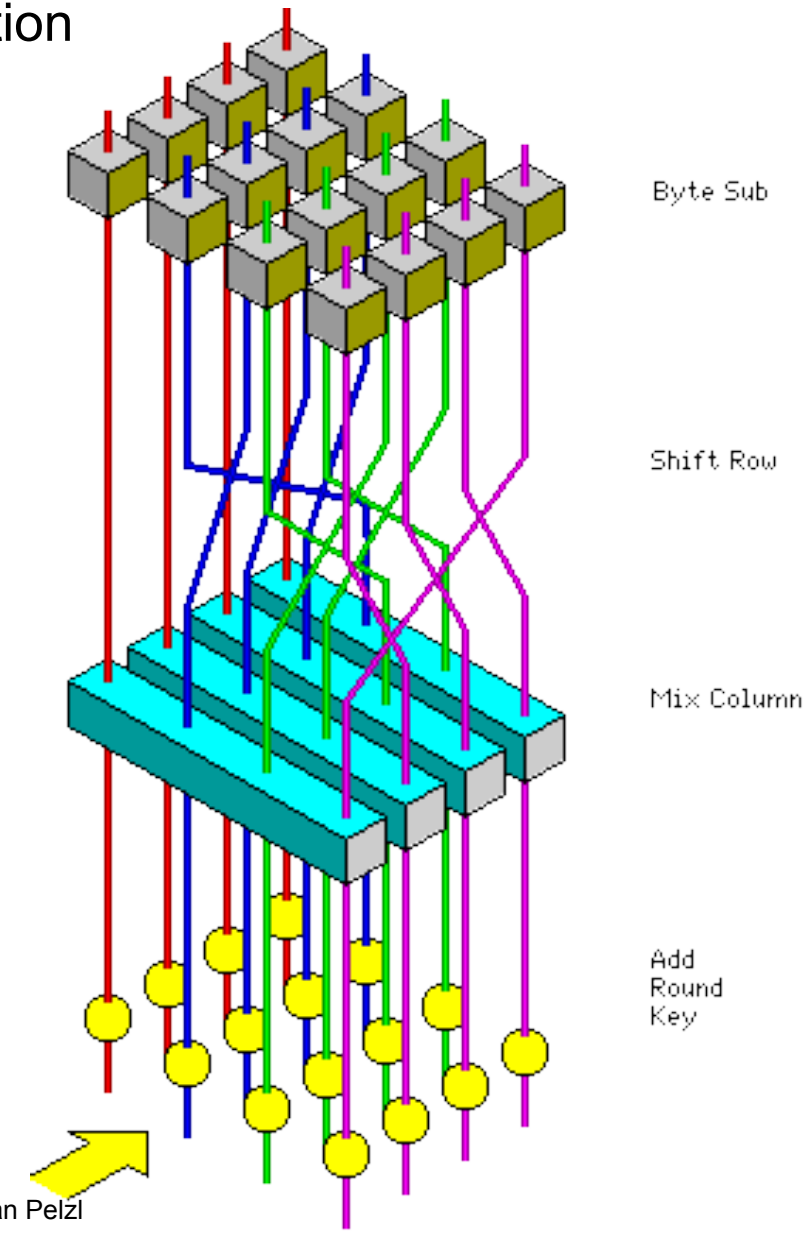
AES
Übersicht

- Iterative Chiffre mit 10/12/14 Runden
- Jede Runde besteht aus Schichten (engl. Layers)



AES

Übersicht: Rundenfunktion



Übersicht

- Einführung
- Übersicht über den Algorithmus
- **Interne Struktur**
  - **Byte Substitution Schicht**
  - **Diffusion Schicht**
  - **Key Addition Schicht**
  - **Schlüsselfahrplan**
- Entschlüsselung
- Praktische Aspekte

AES
Interne Struktur

- AES ist eine Byte-orientierte Chiffre
- Zustand *A* (d.h. der 128 Bit Datenpfad) kann in einer 4x4 Matrix angeordnet werden:

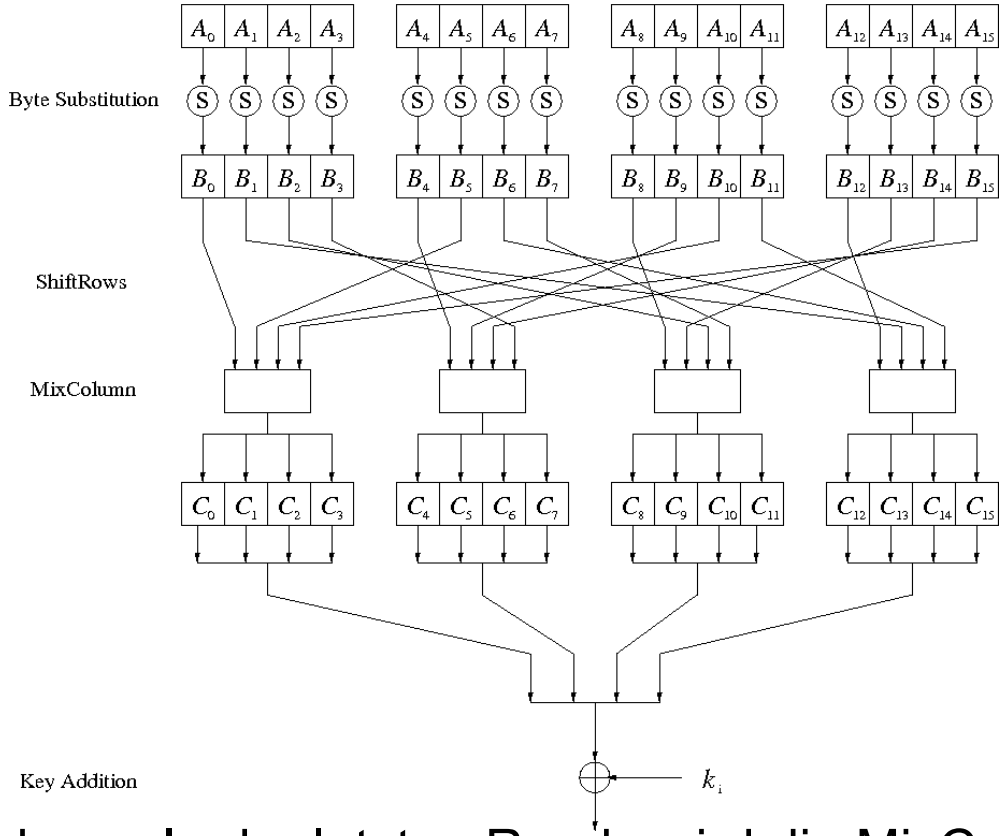
A_0	A_4	A_8	A_{12}
A_1	A_5	A_9	A_{13}
A_2	A_6	A_{10}	A_{14}
A_3	A_7	A_{11}	A_{15}

$A_0, \ldots, A_{15}$ sind die 16 Eingangsbyte des AES

AES

Interne Struktur

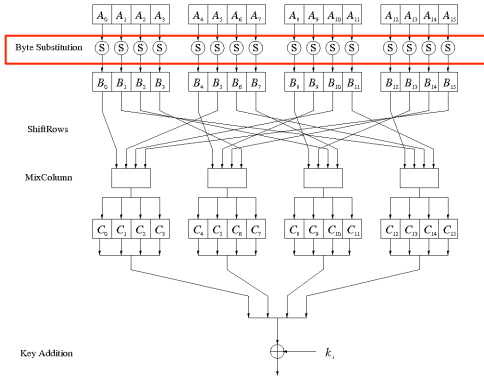
- Rundenfunktion für die Runden 1,2,…,n_{r-1}:



- Anmerkung: In der letzten Runde wird die MixColumn Transformation ausgelassen

AES

Interne Struktur: Die Byte Substitution Schicht

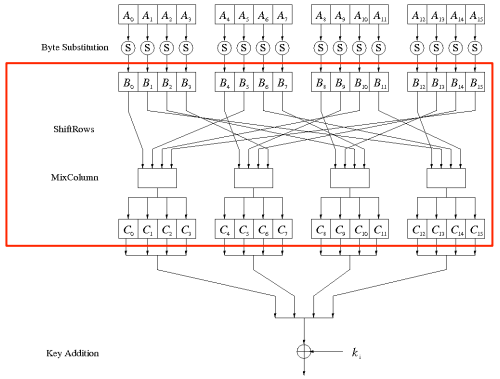


- Besteht aus **16 S-Boxen**
- Die S-Boxen sind
  - **Identisch**
  - Das einzige **nichtlineare** Element des AES, d.h., ByteSub(A_i) + ByteSub(A_j) ≠ ByteSub(A_i + A_j) für i,j = 0,…,15
  - **Bijektiv**, d.h., es existiert eine eineindeutige Abbildung der Eingangsbyte auf die Ausgangsbyte
    ⇒ S-Box ist eindeutig umkehrbar

- Üblicherweise Realisierung als Tabelle in Software

AES

Interne Struktur: Die Diffusionsschicht



- Erzeugt **Diffusion** über allen Eingangsbit

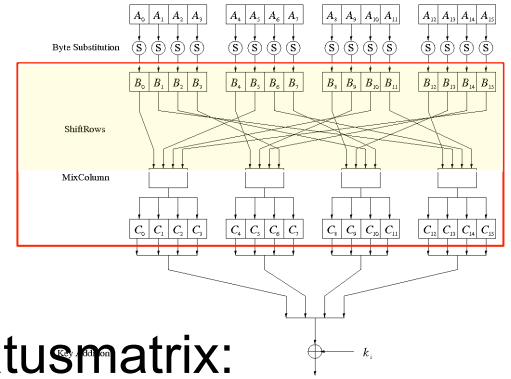
- Besteht aus zwei Unterschichten:
  - **ShiftRows**: Permutation der Daten auf Byte-Level
  - **MixColumn**: Matrix Operation, welche 4-Byte-Blöcke kombiniert ("vermischt")

- Durchführung einer **linearen Transformation** der Statusmatrix *A*, *B*, d.h.,

$$\text{DIFF}(A) + \text{DIFF}(B) = \text{DIFF}(A + B)$$

AES

Interne Struktur: Diffusionsschicht - ShiftRows



- Zyklische Verschiebung der Reihen in der Statusmatrix:

Eingangsmatrix

B_0	B_4	B_8	B_{12}
B_1	B_5	B_9	B_{13}
B_2	B_6	B_{10}	B_{14}
B_3	B_7	B_{11}	B_{15}

Ausgangsmatrix

B_0	B_4	B_8	B_{12}
B_5	B_9	B_{13}	B_1
B_{10}	B_{14}	B_2	B_6
B_{15}	B_3	B_7	B_{11}

keine Verschiebung

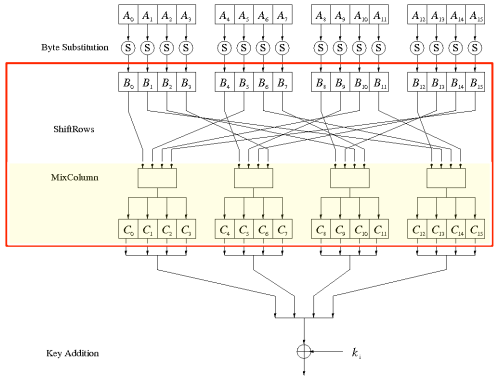
← Linksverschiebung um 1

← Linksverschiebung um 2

← Linksverschiebung um 3

AES

Interne Struktur: Diffusionsschicht - MixColumn



- „Vermischung“ der Spalten der Statusmatrix durch lineare Transformation

- Betrachtung jeder 4-Byte-Spalte als Vektor und Multiplikation mit einer festen 4x4 Matrix:

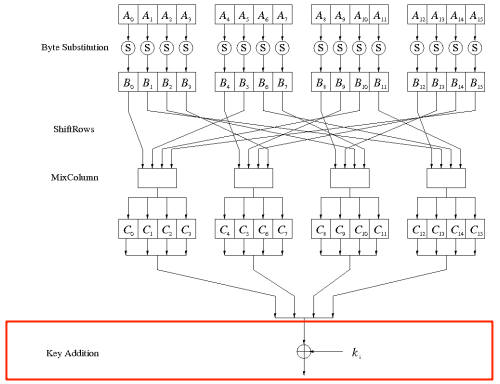
$$\begin{pmatrix} C_0 \\ C_1 \\ C_2 \\ C_3 \end{pmatrix} = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \cdot \begin{pmatrix} B_0 \\ B_5 \\ B_{10} \\ B_{15} \end{pmatrix}$$

  wobei 01, 02 und 03 in hexadezimaler Notation gegeben sind

- Durchführung der Arithmetik im endlichen Körper $GF(2^8)$

AES

Interne Struktur: Key Addition Schicht



- Eingang:
  - 16 Byte Statusmatrix C
  - 16 Byte Unterschlüssel k_i
- Ausgang: $C \oplus k_i$
- Generierung der Unterschlüssel durch Schlüsselfahrplan

AES

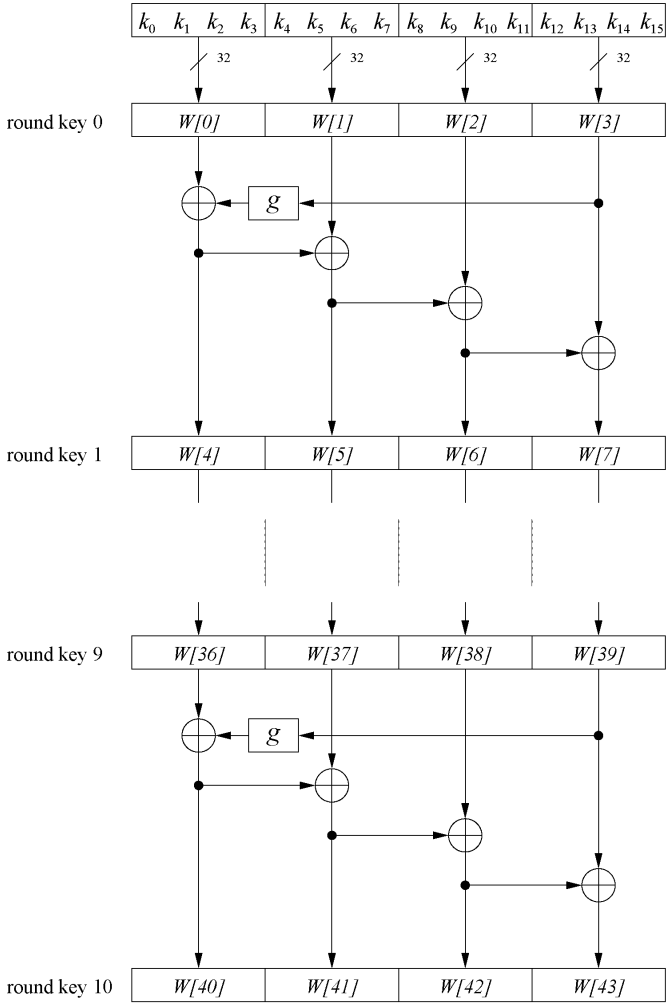
Interne Struktur: Schlüsselfahrplan

- Rekursive Ableitung der Unterschlüssel aus dem 128/192/256 Bit Eingangsschlüssel
- Jede Runde hat einen Unterschlüssel, zzgl. ein Unterschlüssel am Anfang des AES

Schlüssellänge in Bit	Anzahl der Unterschlüssel
128	11
192	13
256	15

- Key Whitening: Unterschlüssel werden sowohl am Anfang als auch am Ende des AES verwendet ⇒ # Unterschlüssel = # Runden + 1
- Je nach Anzahl der Runden unterschiedliche Schlüsselableitung

AES
Interne Struktur: Schlüsselfahrplan



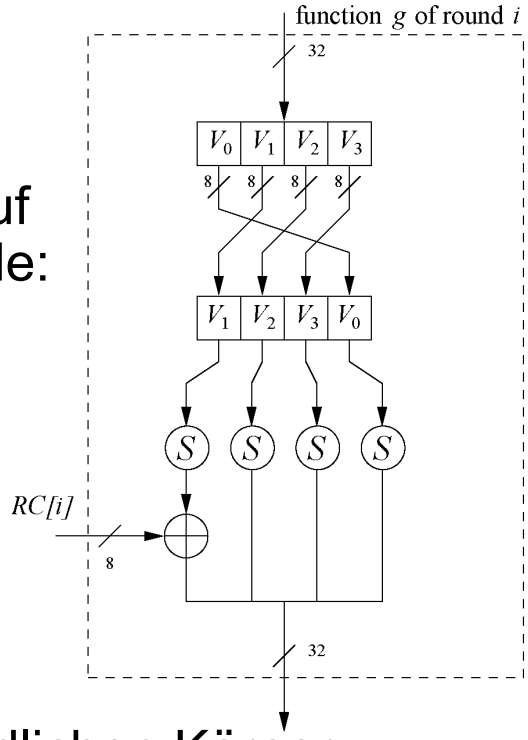
Beispiel: Schlüsselfahrplan für 128 Bit AES Schlüssel

- Wort-orientiert: 1 Wort = 32 Bit
- Speicherung der 11 Unterschlüssel in *W[0]…W[3], W[4]…W[7], … , W[40]… W[43]*
- Erster Unterschlüssel *W[0]…W[3]* ist der Eingangsschlüssel des AES

AES

Interne Struktur: Schlüsselfahrplan

- *g*-Funktion rotiert die vier Eingangsbyte und führt eine byteweise S-Box Substitution durch
  $\Rightarrow$ Nichtlinearität



- Addition des Rundenkoeffizienten *RC* auf das Byte links aussen, Variation je Runde:

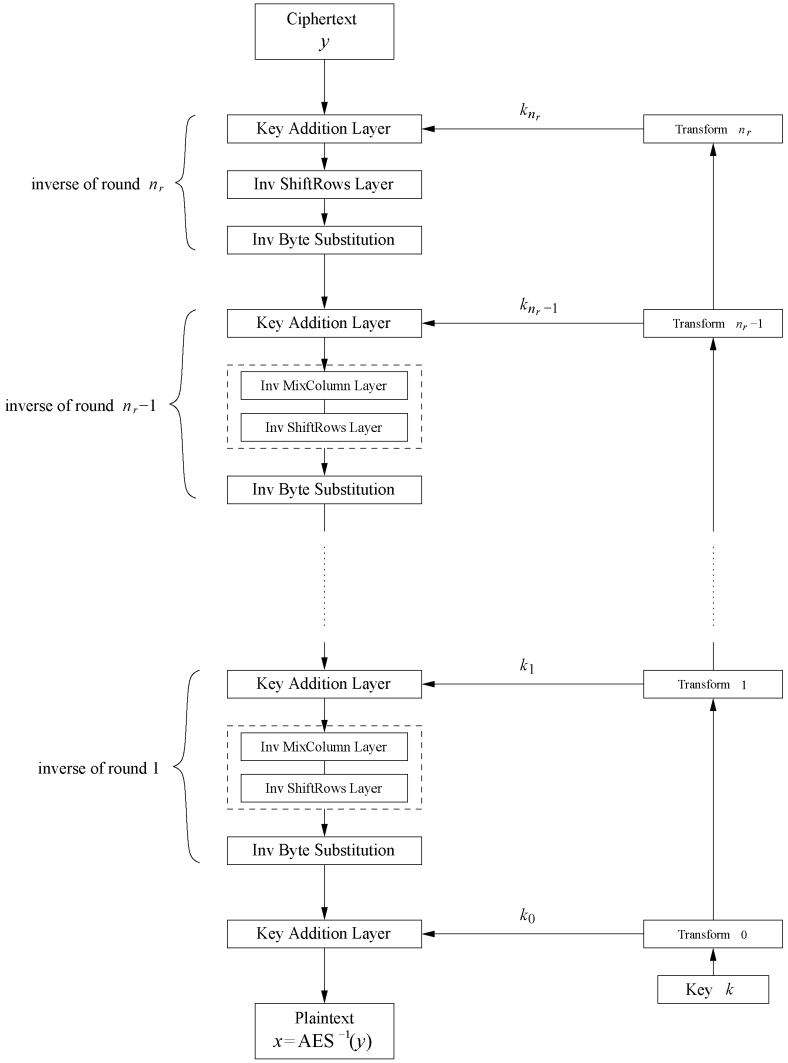
  $RC[1] = x^0 = (00000001)_2$
  $RC[2] = x^1 = (00000010)_2$
  $RC[3] = x^2 = (00000100)_2$
  ...
  $RC[10] = x^9 = (00110110)_2$

- x^i repräsentiert ein Element in einem endlichen Körper

Übersicht

- Einführung
- Übersicht über den Algorithmus
- Interne Struktur
  - Byte Substitution Schicht
  - Diffusion Schicht
  - Key Addition Schicht
  - Schlüsselfahrplan
- **Entschlüsselung**
- Praktische Aspekte

AES
Entschlüsselung



- AES basiert nicht auf einem Feistelnetzwerk

⇒ Alle Schichten müssen bei der Entschlüsselung umgekehrt werden:

  - MixColumn Schicht
    → **Inverse MixColumn Schicht**
  - ShiftRows Schicht
    → **Inverse ShiftRows Schicht**
  - Byte Substitution Schicht
    → **Inverse Byte Substitution Schicht**
  - Key Addition Schicht ist ihre eigene Inverse

AES

Entschlüsselung: Inverse MixColumn Schicht

- Zur Inversion der MixColumn Operation muss jede Spalte der Statusmatrix *C* mit der **Inversen 4x4 Matrix** multipliziert werden, d.h.

$$\begin{pmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \end{pmatrix} = \begin{pmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{pmatrix} \cdot \begin{pmatrix} C_0 \\ C_1 \\ C_2 \\ C_3 \end{pmatrix}$$

  wobei 09, 0*B, 0D* und 0*E* in hexadezimaler Notation gegeben sind

- Alle Arithmetik wird wiederum im endlichen Körper $GF(2^8)$ durchgeführt

AES

Entschlüsselung: Inverse ShiftRows Schicht

- Verschiebung aller Zeilen der Status Matrix *B* in die entgegengesetzte Richtung:

Eingangsmatrix

B_0	B_4	B_8	B_{12}
B_1	B_5	B_9	B_{13}
B_2	B_6	B_{10}	B_{14}
B_3	B_7	B_{11}	B_{15}

Ausgangsmatrix

B_0	B_4	B_8	B_{12}
B_{13}	B_1	B_5	B_9
B_{10}	B_{14}	B_2	B_6
B_7	B_{11}	B_{15}	B_3

keine Verschiebung

→ Rechtsverschiebung um 1

→ Rechtsverschiebung um 2

→ Rechtsverschiebung um 3

AES
Entschlüsselung

- **Inverse Byte Substitution Schicht**:
  - Konstruktion einer inversen S-Box durch

    $$A_i = S^{-1}(B_i) = S^{-1}(S(A_i))$$

    $\Rightarrow$ Verwendung der inversen S-Box für die Entschlüsselung. Realisierung als Tabelle in Software üblich.

- **Schlüsselfahrplan für die Entschlüsselung**:
  - Unterschlüssel werden im Vergleich zur Verschlüsselung in umgekehrter Reihenfolge benötigt
  - In der Praxis wird derselbe Schlüsselfahrplan für die Ver- und Entschlüsselung verwendet, d.h. alle Unterschlüssel werden **vor** der Durchführung der ersten Runde berechnet.

Übersicht

- Einführung
- Übersicht über den Algorithmus
- Interne Struktur
  - Byte Substitution Schicht
  - Diffusion Schicht
  - Key Addition Schicht
  - Schlüsselfahrplan
- Entschlüsselung
- **Praktische Aspekte**

AES
Implementierung in Software

- Eine Anforderung an den AES war die effiziente Realisierung in Software
- Eine einfache Umsetzung des AES auf 8-Bit-Prozessoren (z.B., Chipkarten) ist sehr effizient, jedoch ineffizient auf 32-Bit oder 64-Bit Architekturen
- Besserer Ansatz: Zusammenführung aller Rundenfunktionen (außer der Schlüsseladdition) in eine Tabelle
  - Resultat: Vier Tabellen mit 256 Einträgen, je 32 Bit breit
  - Eine Runde kann mit 16 Table Look-ups durchgeführt werden
- Typischer Datendurchsatz in SW > 1.6 Gbit/s auf modernen 64-Bit-Prozessoren

AES
Sicherheit

- **Brute-Force-Angriffe:** Unmöglich aufgrund der Schlüssellänge von 128, 192 oder 256 Bit

- **Analytische Angriffe:** Derzeit ist kein praktikabler analytischer Angriff bekannt, welcher besser als Brute-Force ist

- **Seitenkanalangriffe:**
  - Zahlreiche Seitenkanalangriffe auf den AES wurden veröffentlicht
  - Anmerkung: Seitenkanalangriffe greifen die jeweilige Implementierung des Algorithmus an

AES
Lessons Learned

- AES ist eine moderne Blockchiffre mit drei möglichen Schlüssellängen (128, 192 und 256 Bit) und bietet damit Langzeitsicherheit gegenüber Brute-Force-Angriffen.
- AES wird seit Ende der 1990er Jahre intensiv untersucht und bisher wurden keine nennenswerten Angriffe gefunden.
- AES basiert nicht auf Feistelnetzwerken. Die grundlegenden AES Operationen haben eine starke Konfusions- und Diffusionseigenschaft und verwenden Arithmetik in endlichen Körpern.
- AES ist Teil aller wesentlichen Standards und wir z.B. von IPSec oder TLS verwendet und ist der Standard für hoheitliche Verschlüsselung
- AES ist effizient in Software und Hardware