

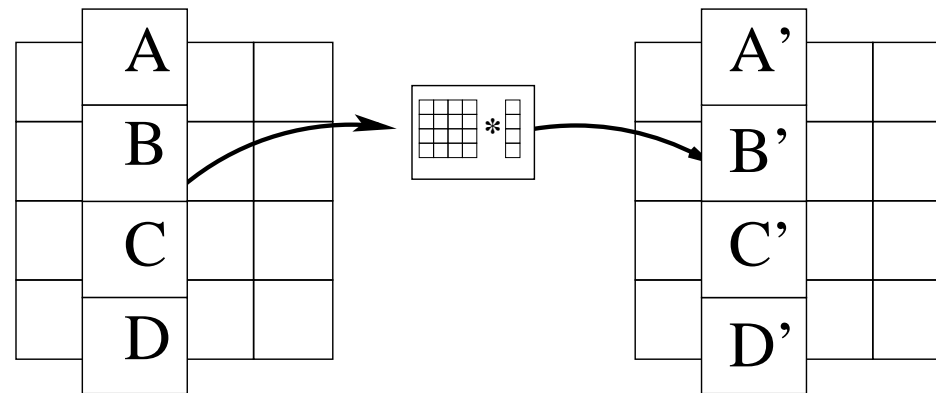
Der SQUARE-Angriff auf Rijndael

1. Eigenschaften von Rijndael

- MDS-Eigenschaft des MixColumn
- Vertauschen von Operationen
- S-Box Schreibweise

2. Der SQUARE-Angriff

MixColumn



- Linearkombination der Bytes einer Spalte
- Maximum Distance Separability - Eigenschaft
Unterscheiden sich 2 Spalten vorher in d Bytes, so unterscheiden sie sich nachher in $5 - d$ Bytes

MDS-Eigenschaft

$$\underbrace{\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}}_{=\mathbf{A} \in \mathbb{F}_{2^8}^{4 \times 4}} \cdot \underbrace{\begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \end{bmatrix}}_M = \begin{bmatrix} 02 \cdot m_0 \oplus 03 \cdot m_1 \oplus 01 \cdot m_2 \oplus 01 \cdot m_3 \\ 01 \cdot m_0 \oplus 02 \cdot m_1 \oplus 03 \cdot m_2 \oplus 01 \cdot m_3 \\ 01 \cdot m_0 \oplus 01 \cdot m_1 \oplus 02 \cdot m_2 \oplus 03 \cdot m_3 \\ 03 \cdot m_0 \oplus 01 \cdot m_1 \oplus 01 \cdot m_2 \oplus 02 \cdot m_3 \end{bmatrix}$$

MDS-Eigenschaft

$$\underbrace{\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}}_{=\mathbf{A} \in \mathbb{F}_{2^8}^{4 \times 4}} \cdot \underbrace{\begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \end{bmatrix}}_M = \begin{bmatrix} 02 \cdot m_0 \oplus 03 \cdot m_1 \oplus 01 \cdot m_2 \oplus 01 \cdot m_3 \\ 01 \cdot m_0 \oplus 02 \cdot m_1 \oplus 03 \cdot m_2 \oplus 01 \cdot m_3 \\ 01 \cdot m_0 \oplus 01 \cdot m_1 \oplus 02 \cdot m_2 \oplus 03 \cdot m_3 \\ 03 \cdot m_0 \oplus 01 \cdot m_1 \oplus 01 \cdot m_2 \oplus 02 \cdot m_3 \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}}_{=\mathbf{A} \in \mathbb{F}_{2^8}^{4 \times 4}} \cdot \underbrace{\begin{bmatrix} m'_0 \\ m_1 \\ m_2 \\ m_3 \end{bmatrix}}_{M'} = \begin{bmatrix} 02 \cdot m'_0 \oplus 03 \cdot m_1 \oplus 01 \cdot m_2 \oplus 01 \cdot m_3 \\ 01 \cdot m'_0 \oplus 02 \cdot m_1 \oplus 03 \cdot m_2 \oplus 01 \cdot m_3 \\ 01 \cdot m'_0 \oplus 01 \cdot m_1 \oplus 02 \cdot m_2 \oplus 03 \cdot m_3 \\ 03 \cdot m'_0 \oplus 01 \cdot m_1 \oplus 01 \cdot m_2 \oplus 02 \cdot m_3 \end{bmatrix}$$

MDS-Eigenschaft

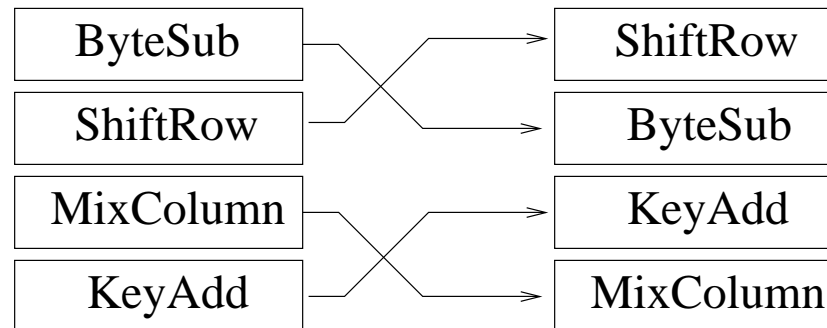
$$\underbrace{\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}}_{=\mathbf{A} \in \mathbb{F}_{2^8}^{4 \times 4}} \cdot \underbrace{\begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \end{bmatrix}}_M = \begin{bmatrix} 02 \cdot m_0 \oplus 03 \cdot m_1 \oplus 01 \cdot m_2 \oplus 01 \cdot m_3 \\ 01 \cdot m_0 \oplus 02 \cdot m_1 \oplus 03 \cdot m_2 \oplus 01 \cdot m_3 \\ 01 \cdot m_0 \oplus 01 \cdot m_1 \oplus 02 \cdot m_2 \oplus 03 \cdot m_3 \\ 03 \cdot m_0 \oplus 01 \cdot m_1 \oplus 01 \cdot m_2 \oplus 02 \cdot m_3 \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}}_{=\mathbf{A} \in \mathbb{F}_{2^8}^{4 \times 4}} \cdot \underbrace{\begin{bmatrix} m'_0 \\ m_1 \\ m_2 \\ m_3 \end{bmatrix}}_{M'} = \begin{bmatrix} 02 \cdot m'_0 \oplus 03 \cdot m_1 \oplus 01 \cdot m_2 \oplus 01 \cdot m_3 \\ 01 \cdot m'_0 \oplus 02 \cdot m_1 \oplus 03 \cdot m_2 \oplus 01 \cdot m_3 \\ 01 \cdot m'_0 \oplus 01 \cdot m_1 \oplus 02 \cdot m_2 \oplus 03 \cdot m_3 \\ 03 \cdot m'_0 \oplus 01 \cdot m_1 \oplus 01 \cdot m_2 \oplus 02 \cdot m_3 \end{bmatrix}$$

Die Differenz lautet

$$\begin{bmatrix} 02 \cdot (m_0 \oplus m'_0) \\ 01 \cdot (m_0 \oplus m'_0) \\ 01 \cdot (m_0 \oplus m'_0) \\ 03 \cdot (m_0 \oplus m'_0) \end{bmatrix} \cdot$$

Vertauschung von Transformationen



- ByteSub und ShiftRow sind vertauschbar, weil sie Wert bzw. Position eines Bytes ändern
- MixColumn und KeyAdd sind vertauschbar, weil beide lineare Transformationen sind:

$$MC(State) \oplus K = MC(State \oplus MC^{-1}(K))$$

Ablauf einer Verschlüsselung

- KeyAdd zu Beginn der Verschlüsselung
- Dann 9 Runden:
 1. ByteSub
 2. ShiftRow
 3. MixColumn
 4. KeyAdd
- Letzte Runde ohne MixColumn

Ablauf einer Entschlüsselung

'normal'	'alternativ'
InvKeyAdd	InvKeyAdd
InvShiftRow	
InvByteSub	InvByteSub
	InvShiftRow
InvKeyAdd	InvMixColumn
InvMixColumn	InvKeyAdd
InvShiftRow	:
InvByteSub	InvByteSub
:	InvShiftRow
InvKeyAdd	InvKeyAdd

Rijndael in S-Box Darstellung

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{S}[a_{0,j}] \\ \mathbf{S}[a_{1,j+C_1}] \\ \mathbf{S}[a_{2,j+C_2}] \\ \mathbf{S}[a_{3,j+C_3}] \end{bmatrix} \oplus \begin{bmatrix} k_{0,j} \\ k_{1,j} \\ k_{2,j} \\ k_{3,j} \end{bmatrix}$$

Rijndael in S-Box Darstellung

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{S}[a_{0,j}] \\ \mathbf{S}[a_{1,j+C_1}] \\ \mathbf{S}[a_{2,j+C_2}] \\ \mathbf{S}[a_{3,j+C_3}] \end{bmatrix} \oplus \begin{bmatrix} k_{0,j} \\ k_{1,j} \\ k_{2,j} \\ k_{3,j} \end{bmatrix}$$

$$\mathbf{S}[a_{0,j}] \begin{bmatrix} 02 \\ 01 \\ 01 \\ 03 \end{bmatrix} \oplus \mathbf{S}[a_{1,j+C_1}] \begin{bmatrix} 03 \\ 02 \\ 01 \\ 01 \end{bmatrix} \oplus \mathbf{S}[a_{2,j+C_2}] \begin{bmatrix} 01 \\ 03 \\ 02 \\ 01 \end{bmatrix} \oplus \mathbf{S}[a_{3,j+C_2}] \begin{bmatrix} 01 \\ 01 \\ 03 \\ 02 \end{bmatrix} \oplus \begin{bmatrix} k_{0,j} \\ k_{1,j} \\ k_{2,j} \\ k_{3,j} \end{bmatrix}$$

Rijndael in S-Box Darstellung (2)

$$\mathbf{T}_0[a] = \begin{bmatrix} \mathbf{S}[a] \cdot 02 \\ \mathbf{S}[a] \\ \mathbf{S}[a] \\ \mathbf{S}[a] \cdot 03 \end{bmatrix}, \mathbf{T}_1[a] = \begin{bmatrix} \mathbf{S}[a] \cdot 03 \\ \mathbf{S}[a] \cdot 02 \\ \mathbf{S}[a] \\ \mathbf{S}[a] \end{bmatrix}$$
$$\mathbf{T}_2[a] = \begin{bmatrix} \mathbf{S}[a] \\ \mathbf{S}[a] \cdot 03 \\ \mathbf{S}[a] \cdot 02 \\ \mathbf{S}[a] \end{bmatrix}, \mathbf{T}_3[a] = \begin{bmatrix} \mathbf{S}[a] \\ \mathbf{S}[a] \\ \mathbf{S}[a] \cdot 03 \\ \mathbf{S}[a] \cdot 02 \end{bmatrix}$$

Rijndael in S-Box Darstellung (2)

$$\mathbf{T}_0[a] = \begin{bmatrix} \mathbf{S}[a] \cdot 02 \\ \mathbf{S}[a] \\ \mathbf{S}[a] \\ \mathbf{S}[a] \cdot 03 \end{bmatrix}, \quad \mathbf{T}_1[a] = \begin{bmatrix} \mathbf{S}[a] \cdot 03 \\ \mathbf{S}[a] \cdot 02 \\ \mathbf{S}[a] \\ \mathbf{S}[a] \end{bmatrix} \\
 \mathbf{T}_2[a] = \begin{bmatrix} \mathbf{S}[a] \\ \mathbf{S}[a] \cdot 03 \\ \mathbf{S}[a] \cdot 02 \\ \mathbf{S}[a] \end{bmatrix}, \quad \mathbf{T}_3[a] = \begin{bmatrix} \mathbf{S}[a] \\ \mathbf{S}[a] \\ \mathbf{S}[a] \cdot 03 \\ \mathbf{S}[a] \cdot 02 \end{bmatrix}$$

$$\mathbf{T}_0[a_{0,j}] \oplus \mathbf{T}_1[a_{1,j+C1}] \oplus \mathbf{T}_2[a_{2,j+C2}] \oplus \mathbf{T}_3[a_{3,j+C3}] \oplus k_j$$

Rijndael in S-Box Darstellung (3)

$$\begin{array}{l}
 \mathbf{S}_0[a] = \begin{bmatrix} \mathbf{S}^{-1}[a] \cdot 0E \\ \mathbf{S}^{-1}[a] \cdot 0B \\ \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0E \\ \mathbf{S}^{-1}[a] \cdot 0B \end{bmatrix}, \mathbf{S}_1[a] = \begin{bmatrix} \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0E \\ \mathbf{S}^{-1}[a] \cdot 0B \\ \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 0B \\ \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0E \end{bmatrix}, \\
 \mathbf{S}_2[a] = \begin{bmatrix} \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0E \\ \mathbf{S}^{-1}[a] \cdot 0B \end{bmatrix}, \mathbf{S}_3[a] = \begin{bmatrix} \mathbf{S}^{-1}[a] \cdot 0B \\ \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0E \end{bmatrix}.
 \end{array}$$

Rijndael in S-Box Darstellung (3)

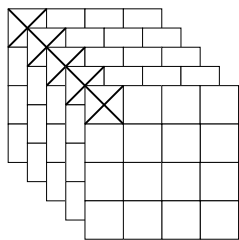
$$\begin{array}{l}
 \mathbf{S}_0[a] = \begin{bmatrix} \mathbf{S}^{-1}[a] \cdot 0E \\ \mathbf{S}^{-1}[a] \cdot 0B \\ \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0E \\ \mathbf{S}^{-1}[a] \cdot 0B \end{bmatrix}, \mathbf{S}_1[a] = \begin{bmatrix} \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0E \\ \mathbf{S}^{-1}[a] \cdot 0B \\ \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 0B \\ \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0E \end{bmatrix}, \\
 \mathbf{S}_2[a] = \begin{bmatrix} \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0E \\ \mathbf{S}^{-1}[a] \cdot 0B \end{bmatrix}, \mathbf{S}_3[a] = \begin{bmatrix} \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0D \\ \mathbf{S}^{-1}[a] \cdot 09 \\ \mathbf{S}^{-1}[a] \cdot 0E \end{bmatrix}.
 \end{array}$$

$$\mathbf{S}_0[a_{0,j}] \oplus \mathbf{S}_1[a_{1,j-C1}] \oplus \mathbf{S}_2[a_{2,j-C2}] \oplus \mathbf{S}_3[a_{3,j-C3}] \oplus k_j$$

Der SQUARE-Angriff

- entwickelt von den Rijndael-Erfindern (siehe z.B. [Daemen & Rijmen(1999)])
- nur für reduzierte Rundenanzahl
- Chosen Plaintext-Angriff auf den letzten Rundenschlüssel
- Begriffsdefinitionen:

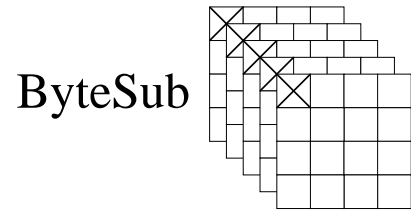
λ -Menge
Menge von 256 Texten



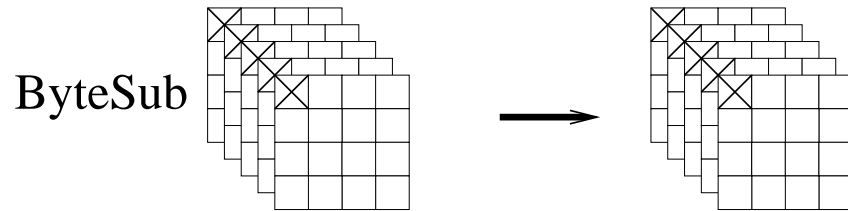
□ passiv
⊗ aktiv

aktives Byte
passives Byte } *ausgegliche Bytes*

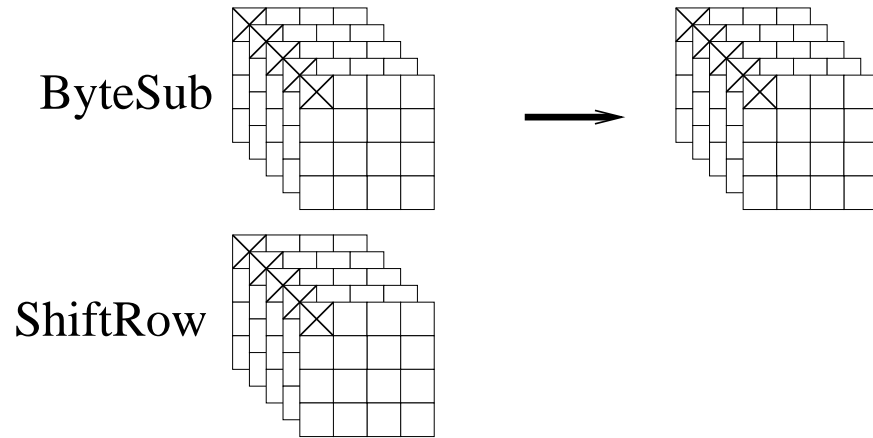
Auswirkungen der Transformationen



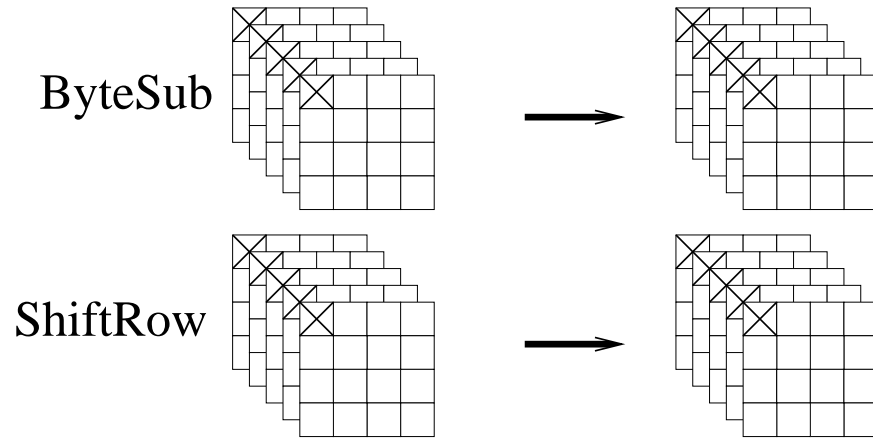
Auswirkungen der Transformationen



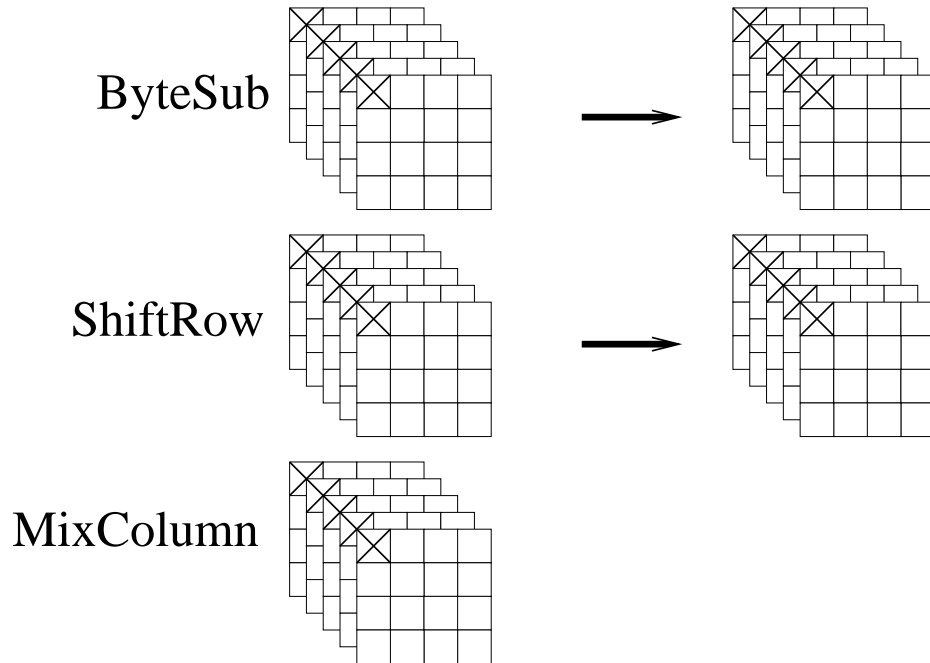
Auswirkungen der Transformationen



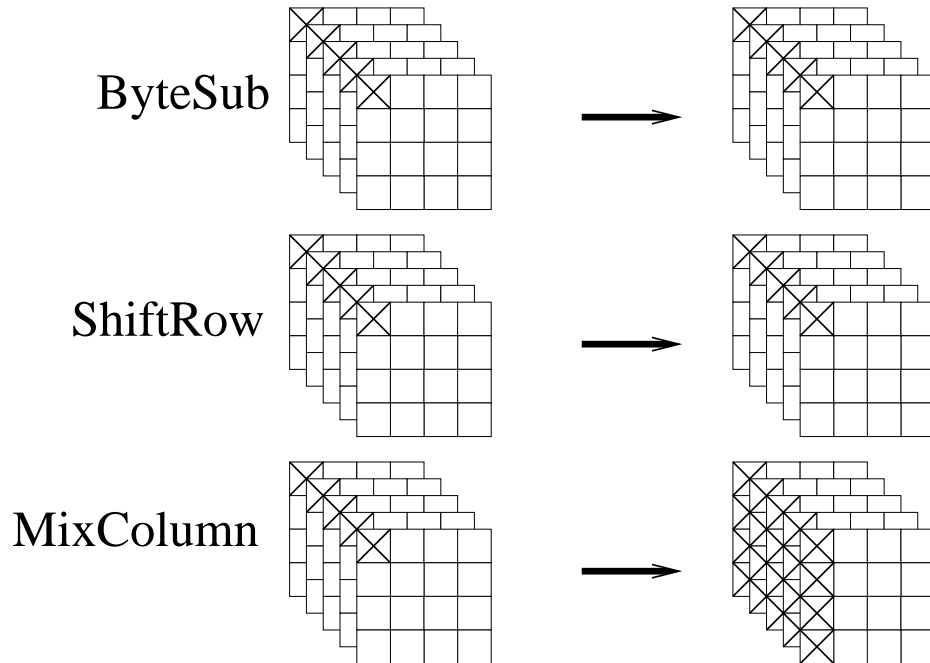
Auswirkungen der Transformationen



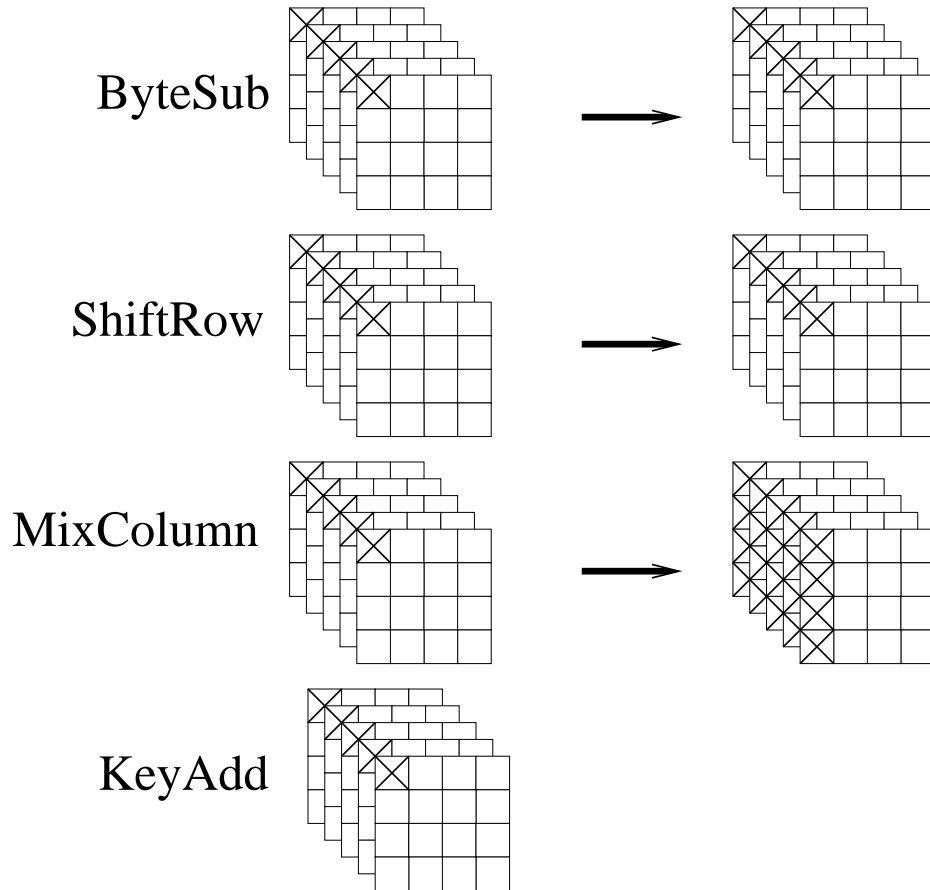
Auswirkungen der Transformationen



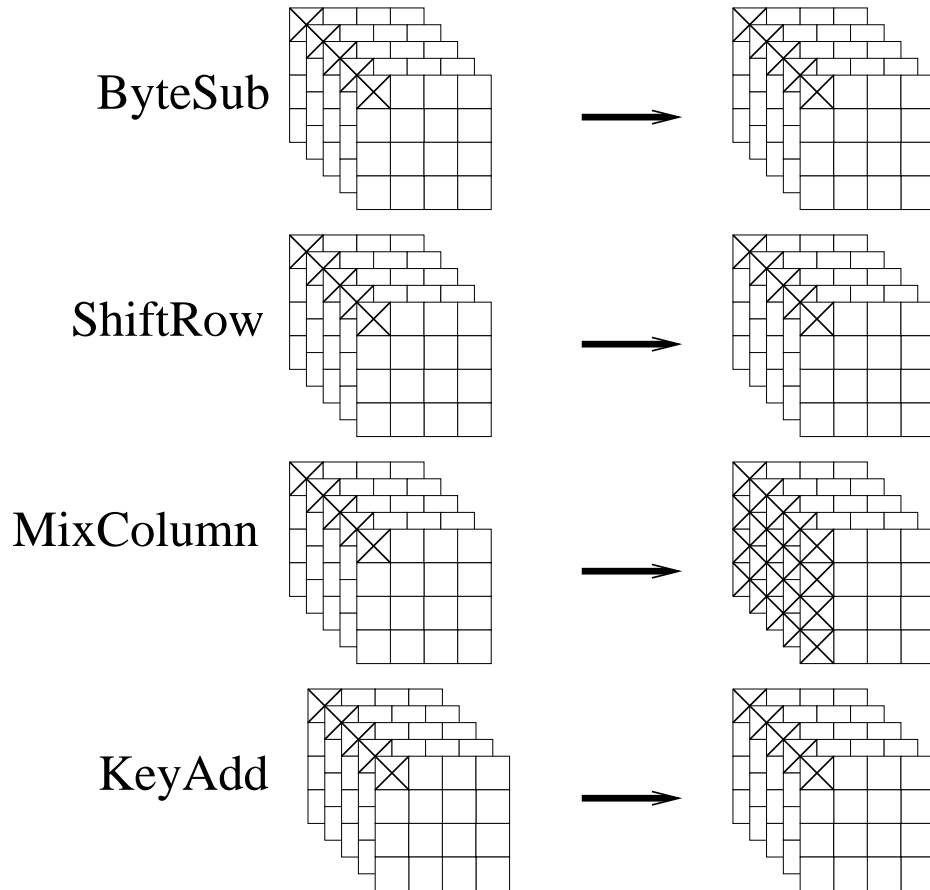
Auswirkungen der Transformationen



Auswirkungen der Transformationen



Auswirkungen der Transformationen

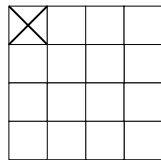


Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd

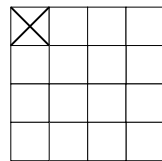
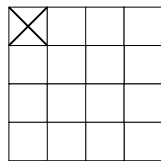
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



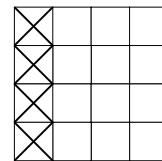
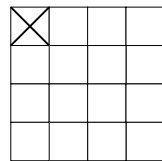
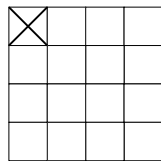
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



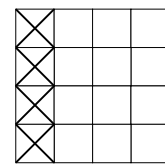
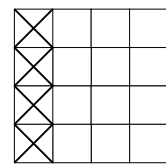
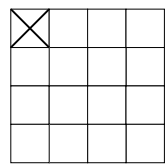
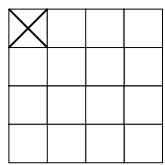
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



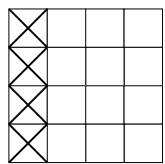
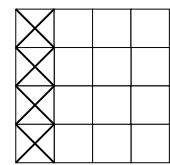
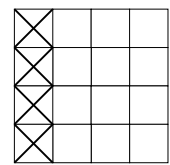
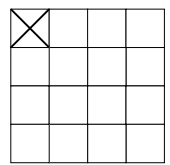
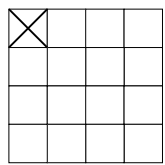
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



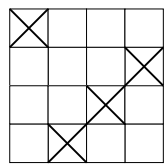
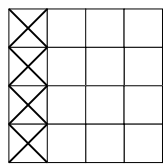
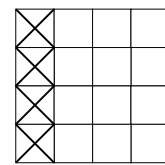
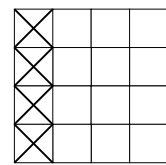
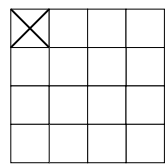
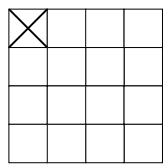
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



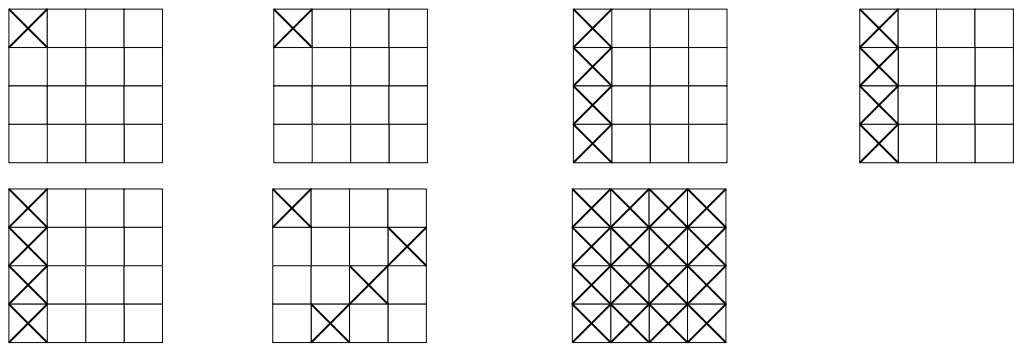
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



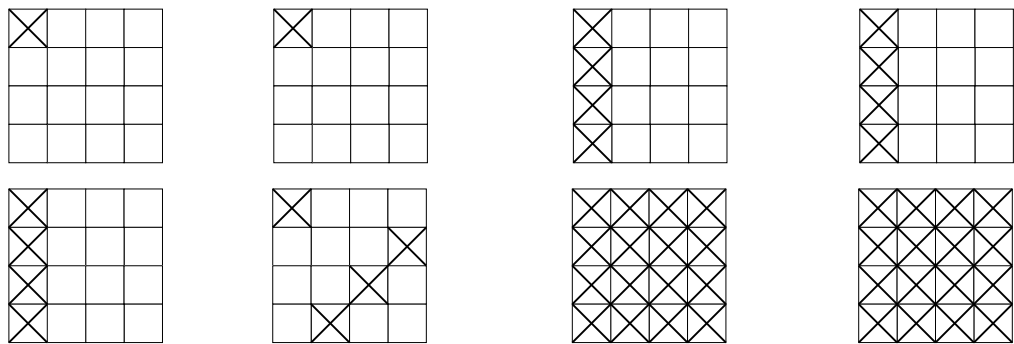
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



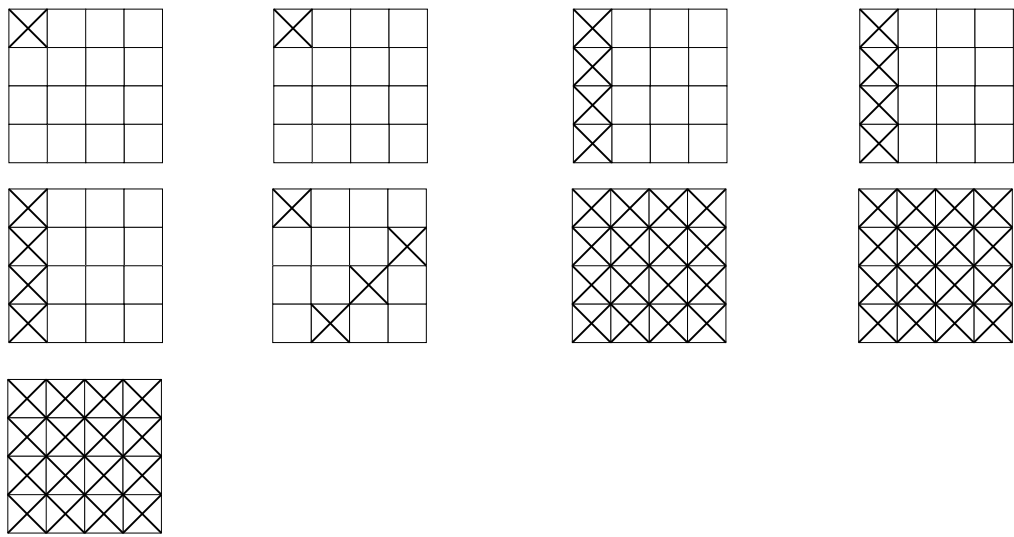
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



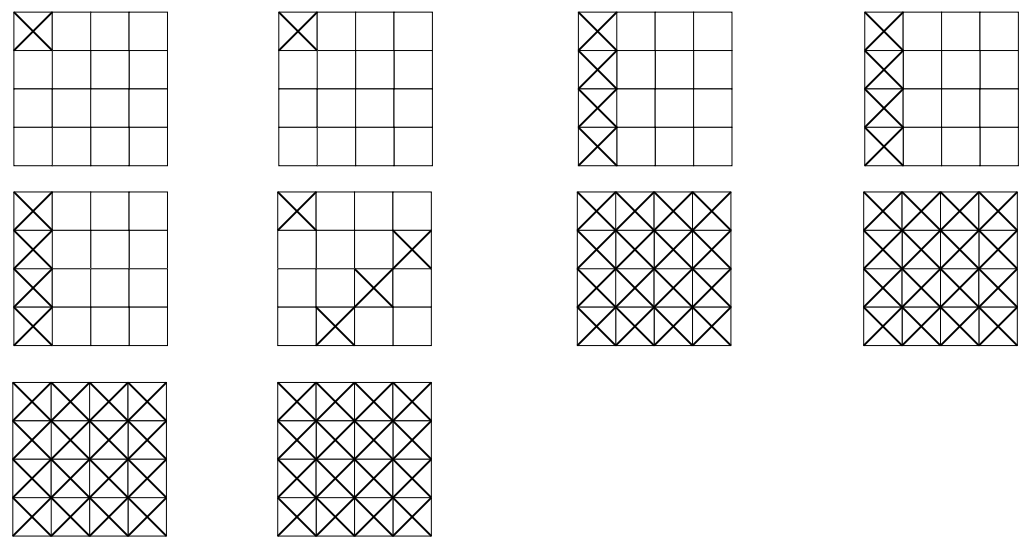
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



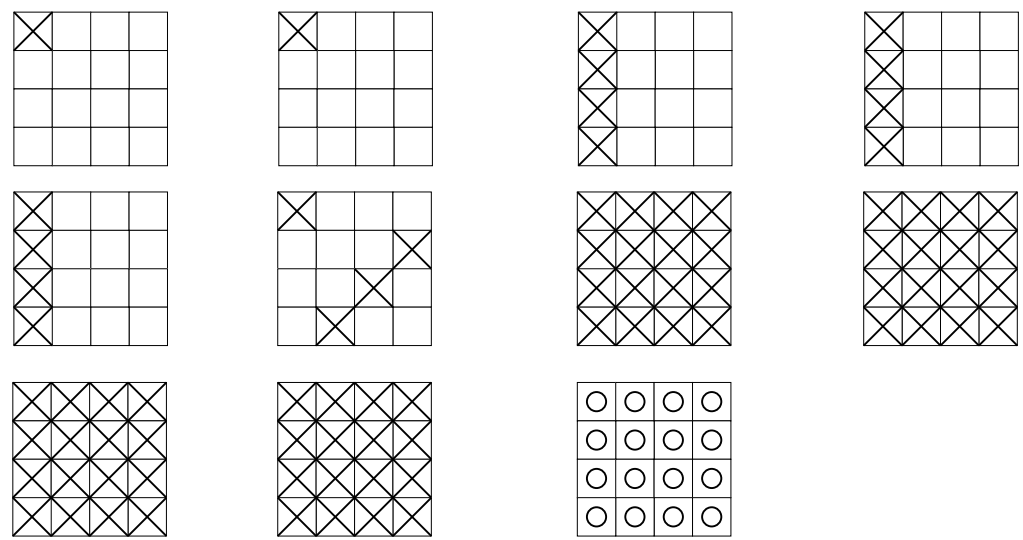
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



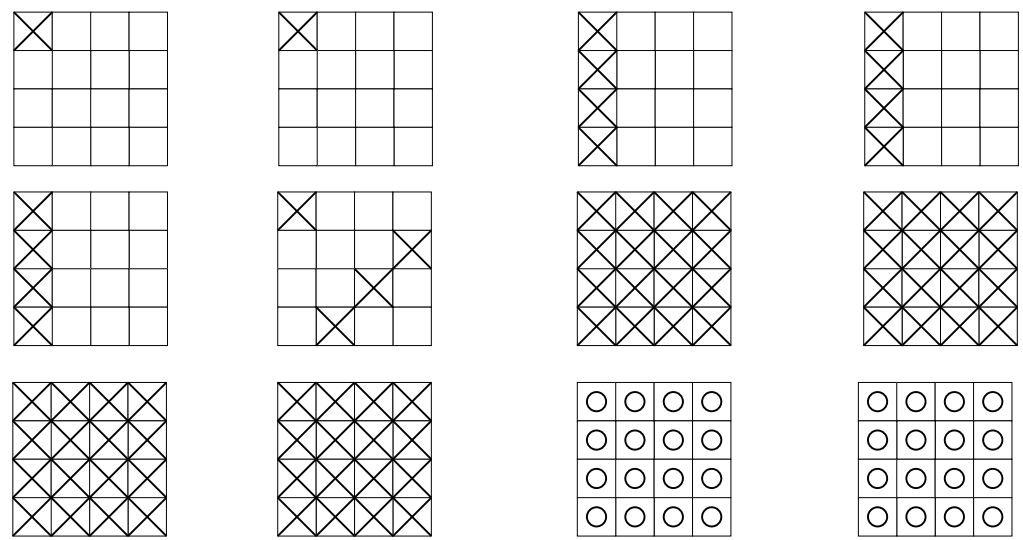
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



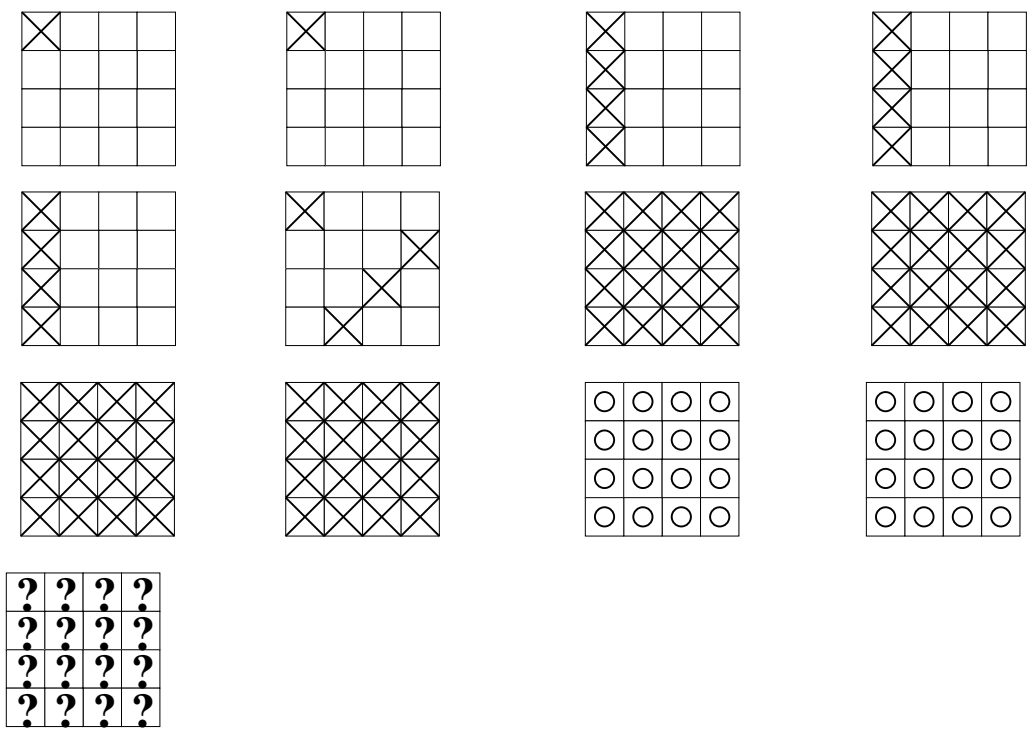
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



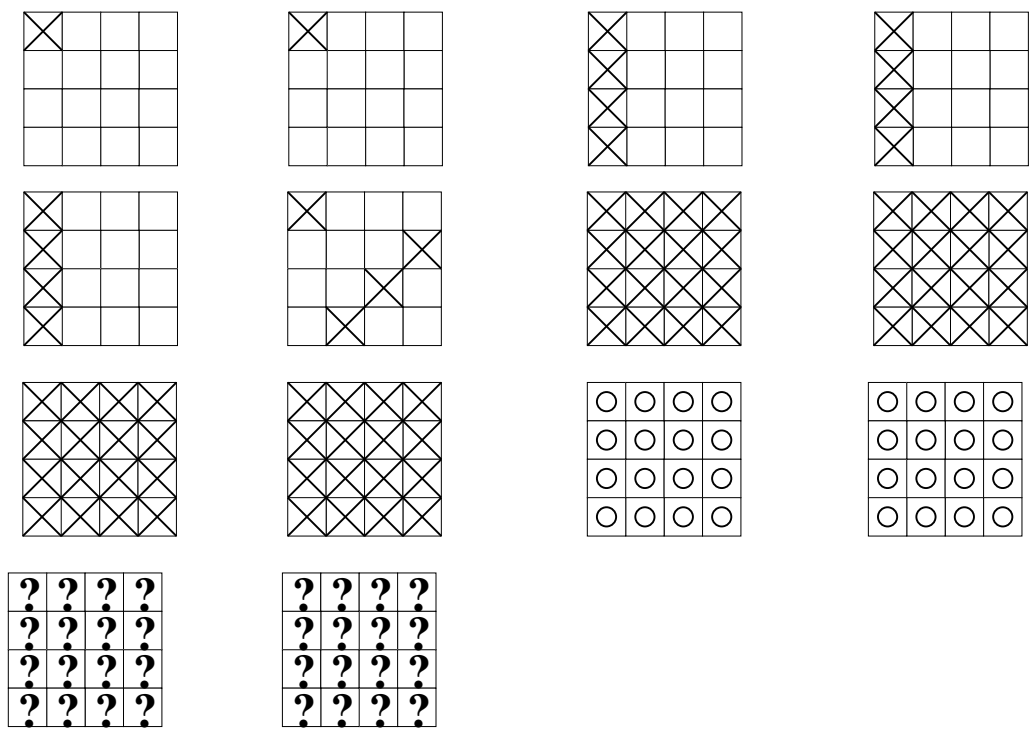
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



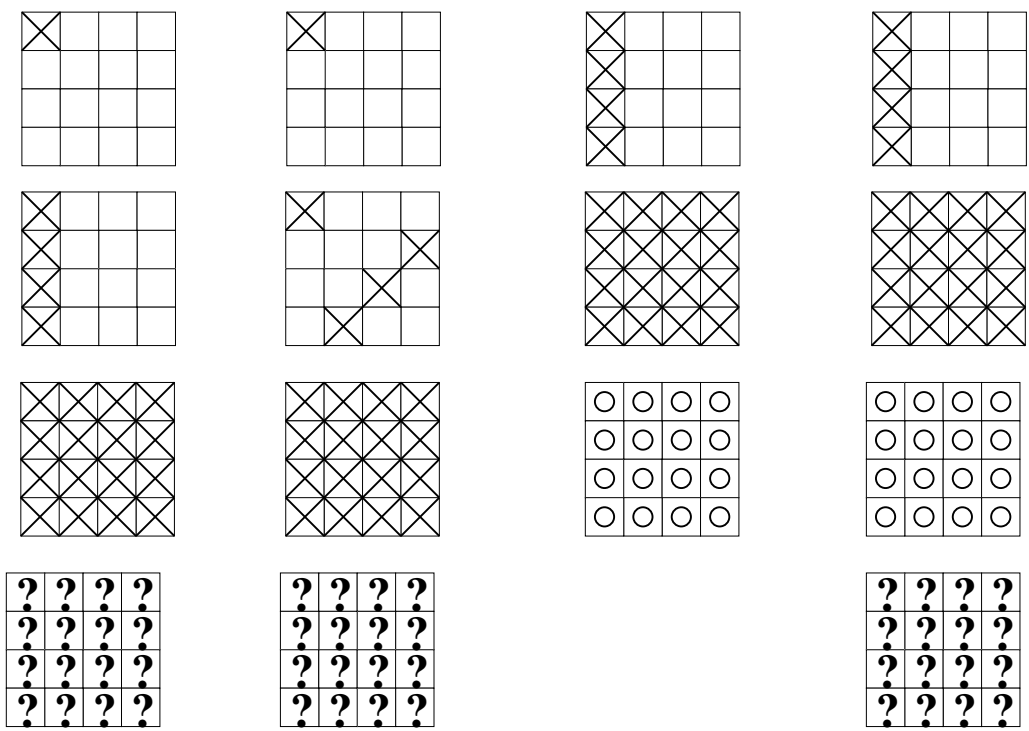
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



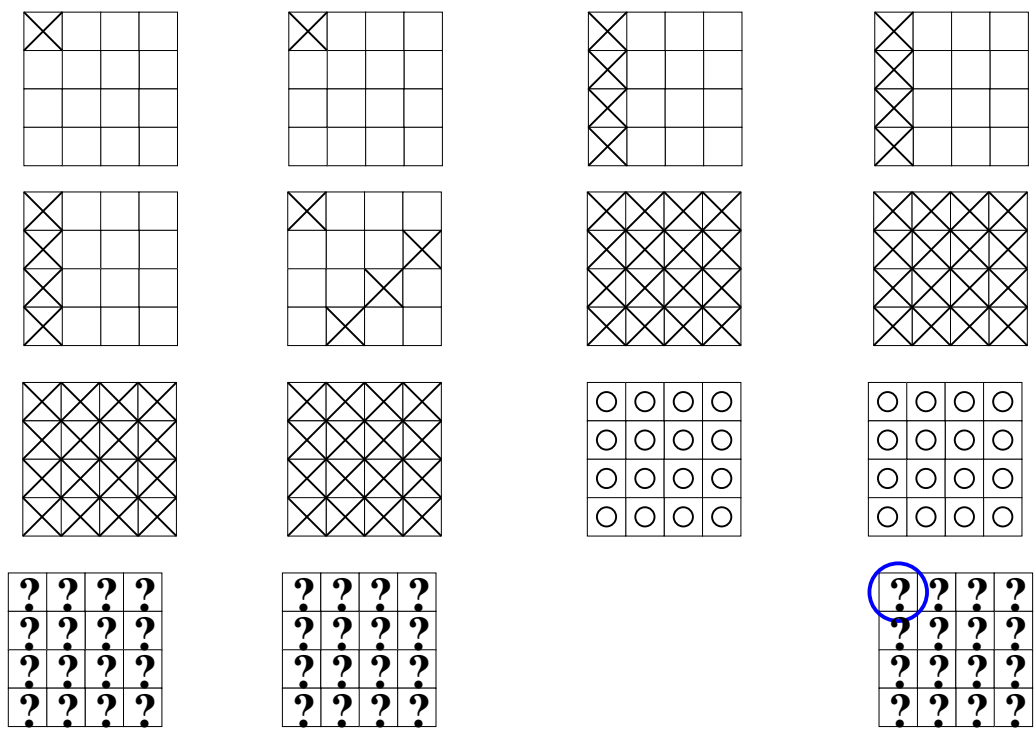
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



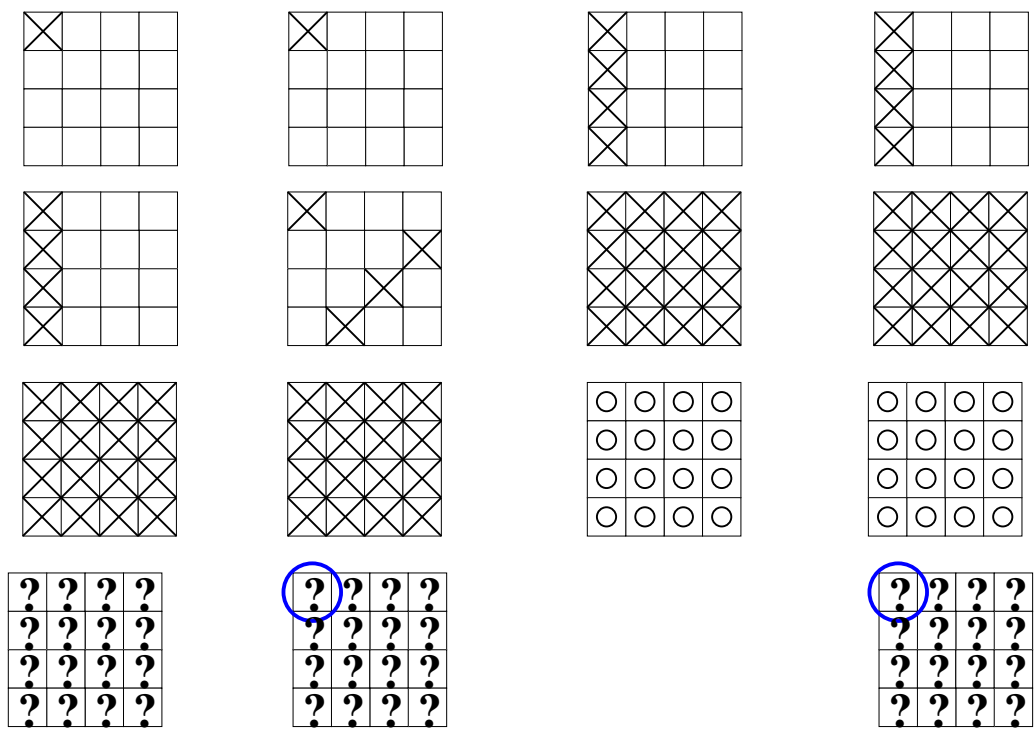
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



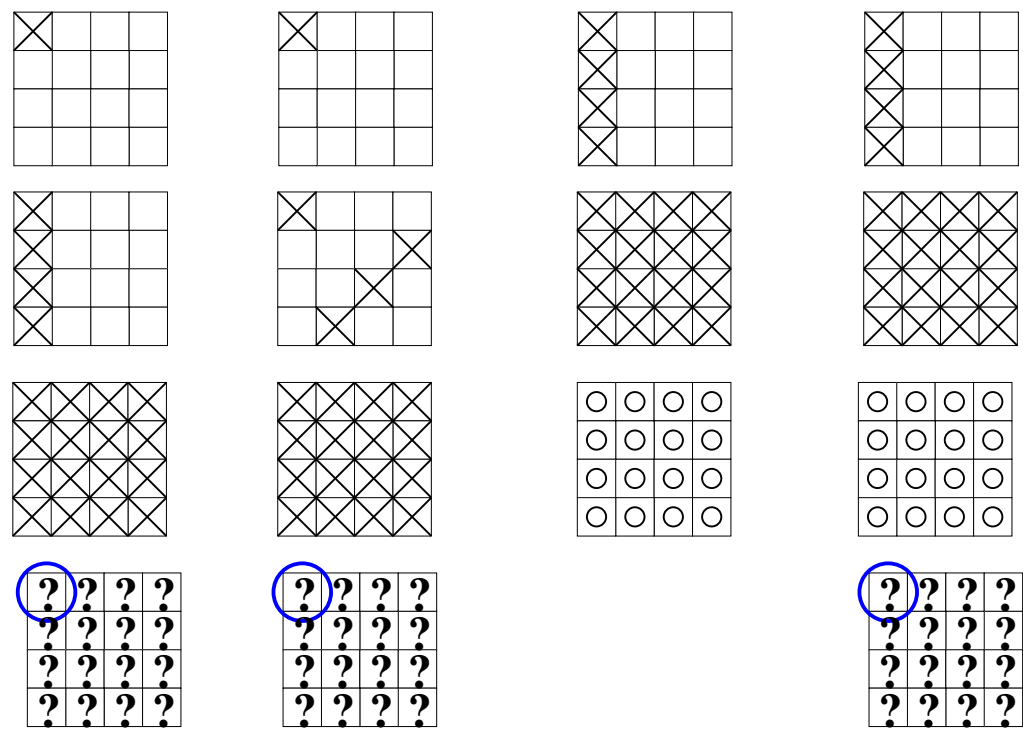
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



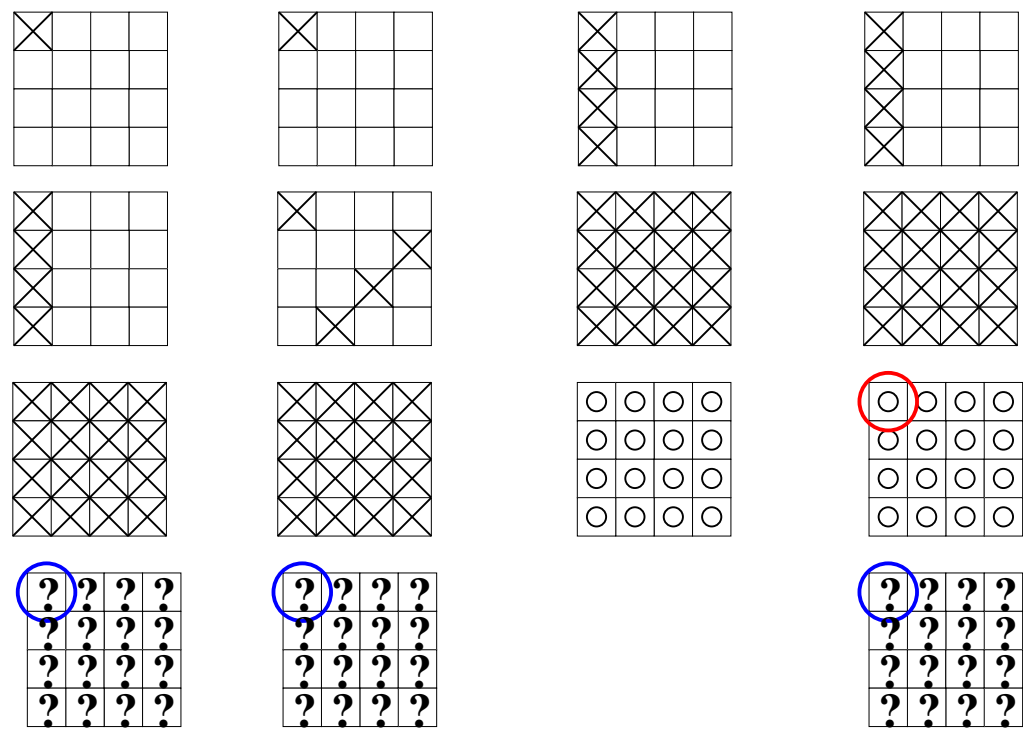
Angriff auf 4 Runden

ByteSub ShiftRow MixColumn KeyAdd



Angriff auf 4 Runden

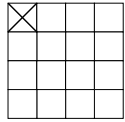
ByteSub ShiftRow MixColumn KeyAdd



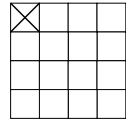
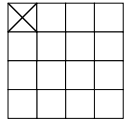
Erweiterung am Ende

ByteSub ShiftRow MixColumn KeyAdd

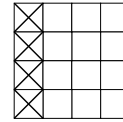
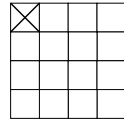
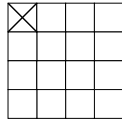
ByteSub ShiftRow MixColumn KeyAdd



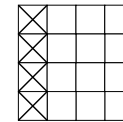
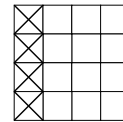
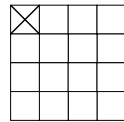
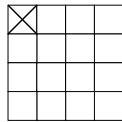
ByteSub ShiftRow MixColumn KeyAdd



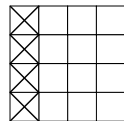
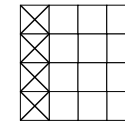
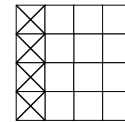
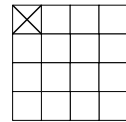
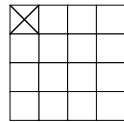
ByteSub ShiftRow MixColumn KeyAdd



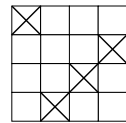
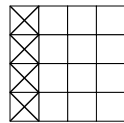
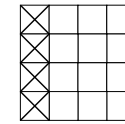
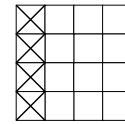
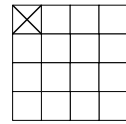
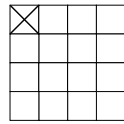
ByteSub ShiftRow MixColumn KeyAdd



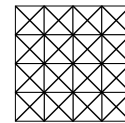
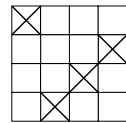
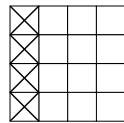
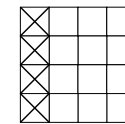
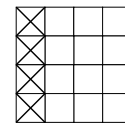
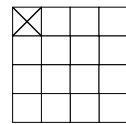
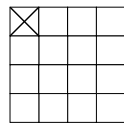
ByteSub ShiftRow MixColumn KeyAdd



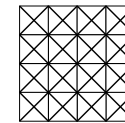
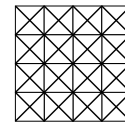
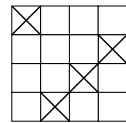
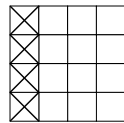
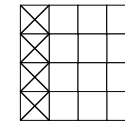
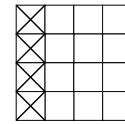
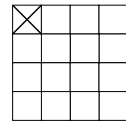
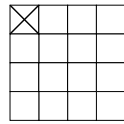
ByteSub ShiftRow MixColumn KeyAdd



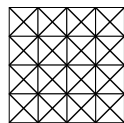
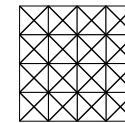
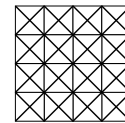
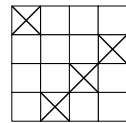
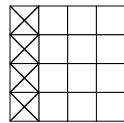
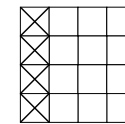
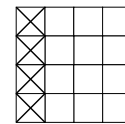
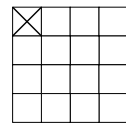
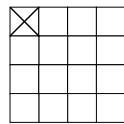
ByteSub ShiftRow MixColumn KeyAdd



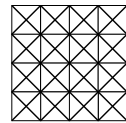
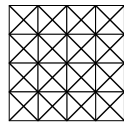
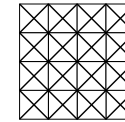
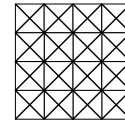
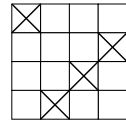
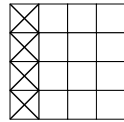
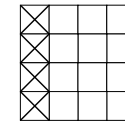
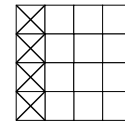
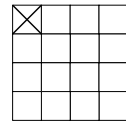
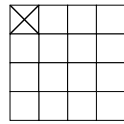
ByteSub ShiftRow MixColumn KeyAdd



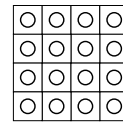
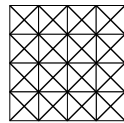
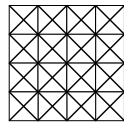
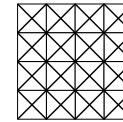
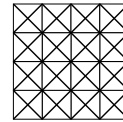
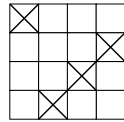
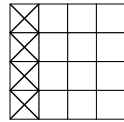
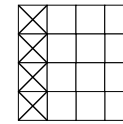
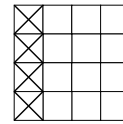
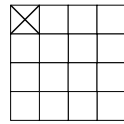
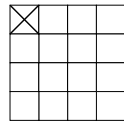
ByteSub ShiftRow MixColumn KeyAdd



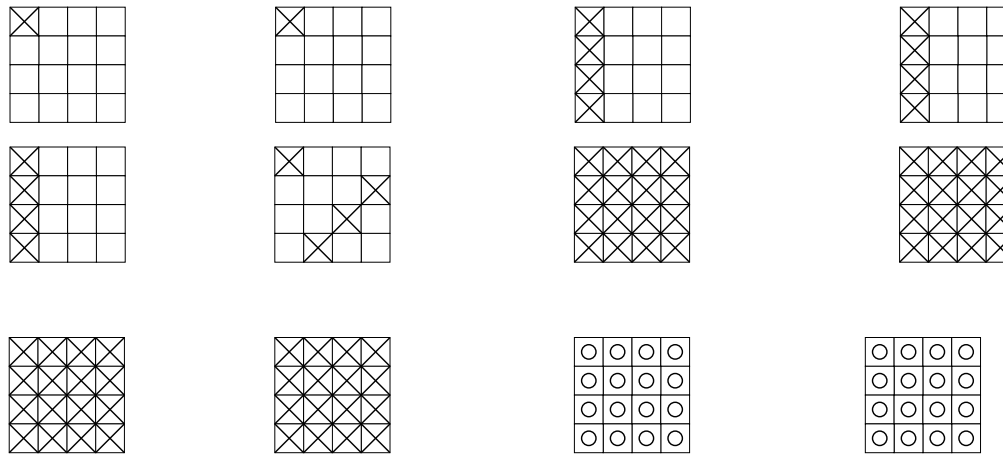
ByteSub ShiftRow MixColumn KeyAdd



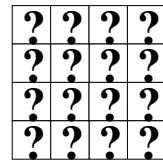
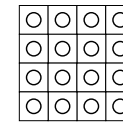
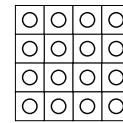
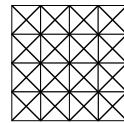
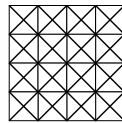
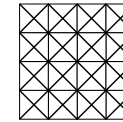
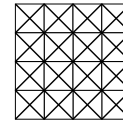
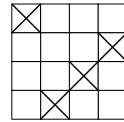
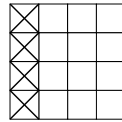
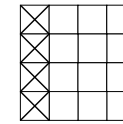
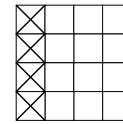
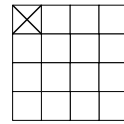
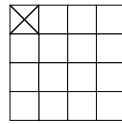
ByteSub ShiftRow MixColumn KeyAdd



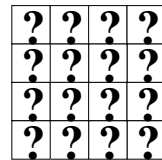
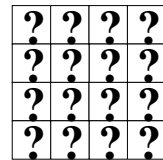
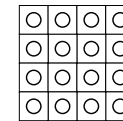
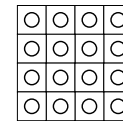
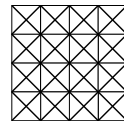
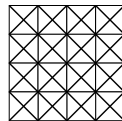
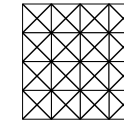
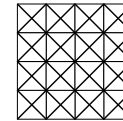
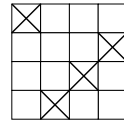
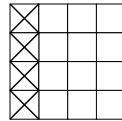
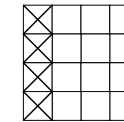
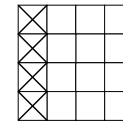
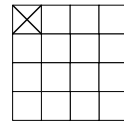
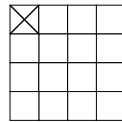
ByteSub ShiftRow MixColumn KeyAdd



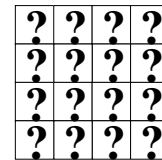
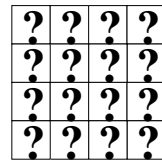
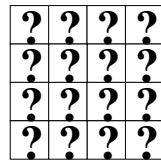
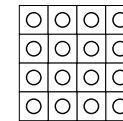
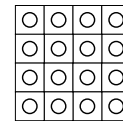
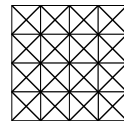
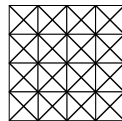
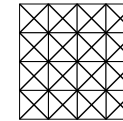
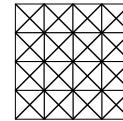
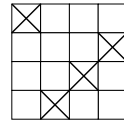
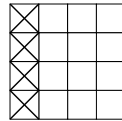
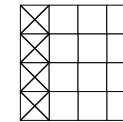
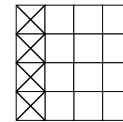
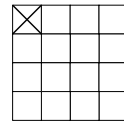
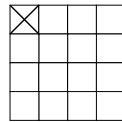
ByteSub ShiftRow MixColumn KeyAdd



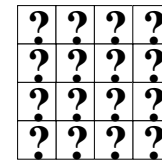
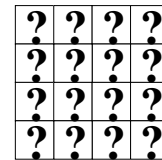
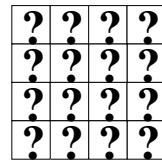
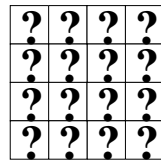
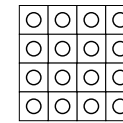
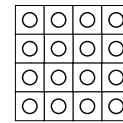
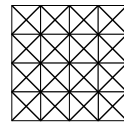
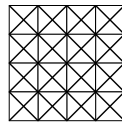
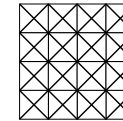
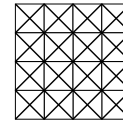
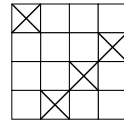
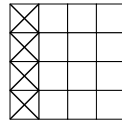
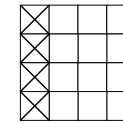
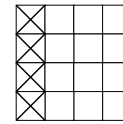
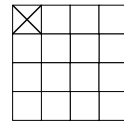
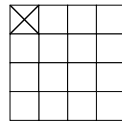
ByteSub ShiftRow MixColumn KeyAdd



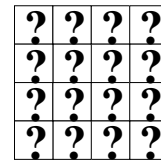
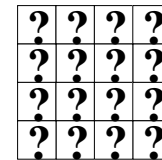
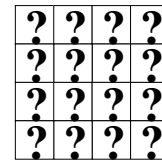
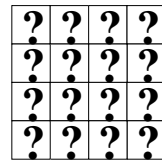
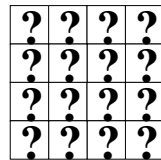
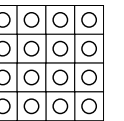
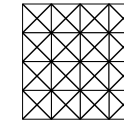
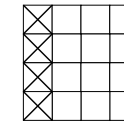
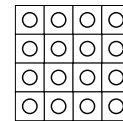
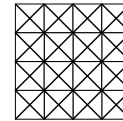
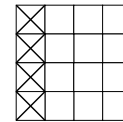
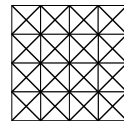
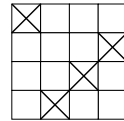
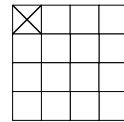
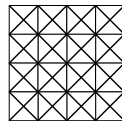
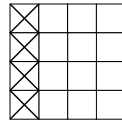
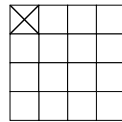
ByteSub ShiftRow MixColumn KeyAdd



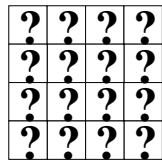
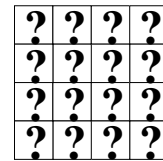
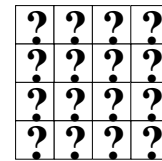
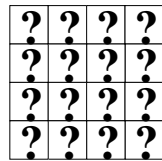
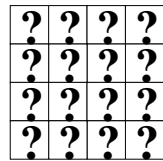
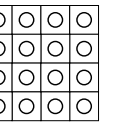
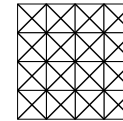
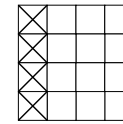
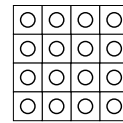
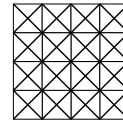
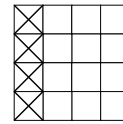
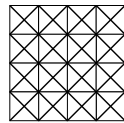
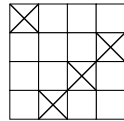
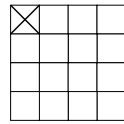
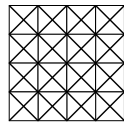
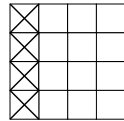
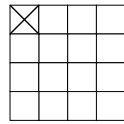
ByteSub ShiftRow MixColumn KeyAdd



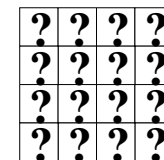
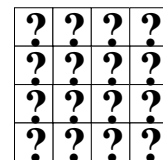
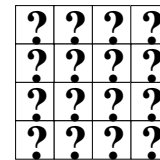
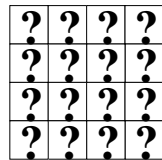
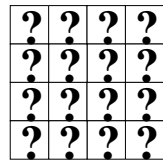
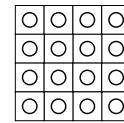
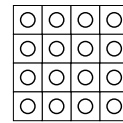
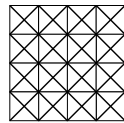
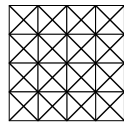
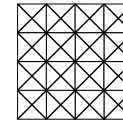
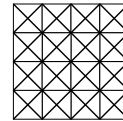
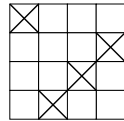
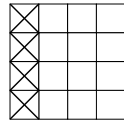
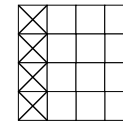
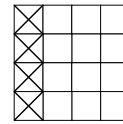
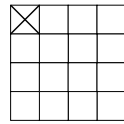
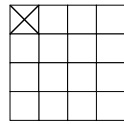
ByteSub ShiftRow MixColumn KeyAdd



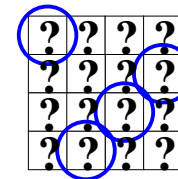
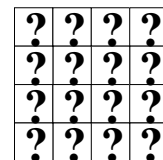
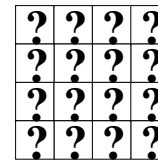
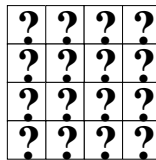
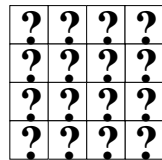
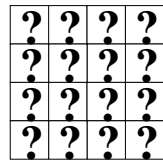
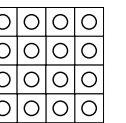
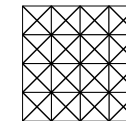
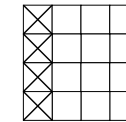
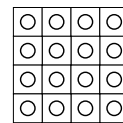
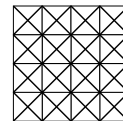
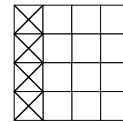
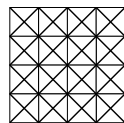
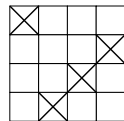
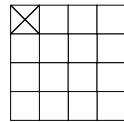
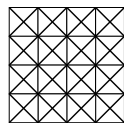
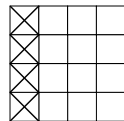
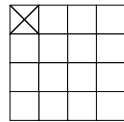
ByteSub ShiftRow MixColumn KeyAdd



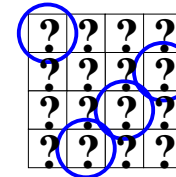
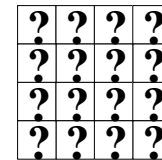
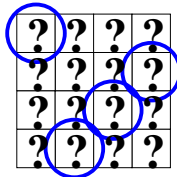
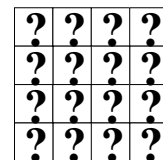
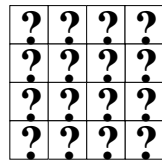
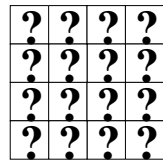
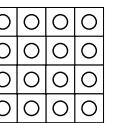
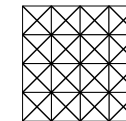
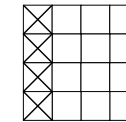
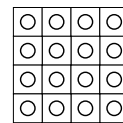
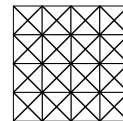
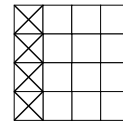
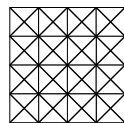
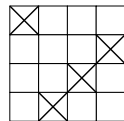
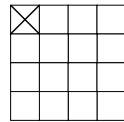
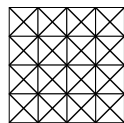
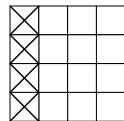
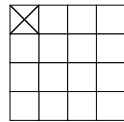
ByteSub ShiftRow MixColumn KeyAdd



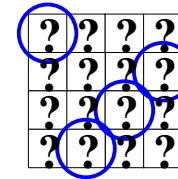
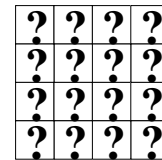
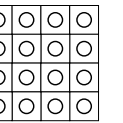
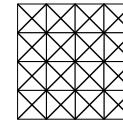
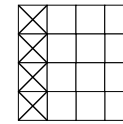
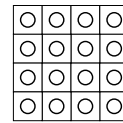
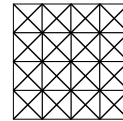
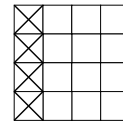
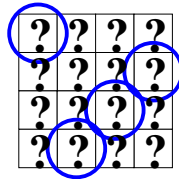
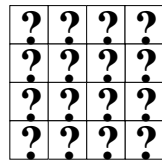
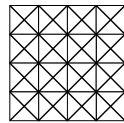
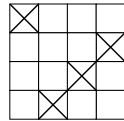
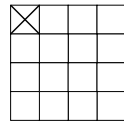
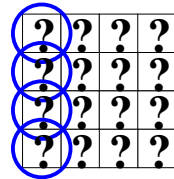
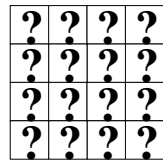
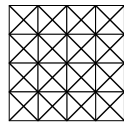
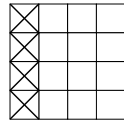
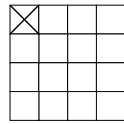
ByteSub ShiftRow MixColumn KeyAdd



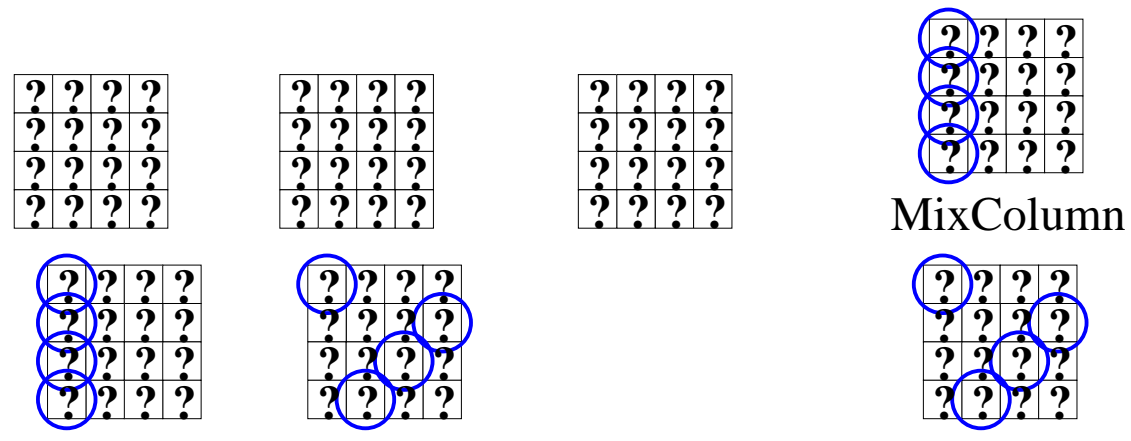
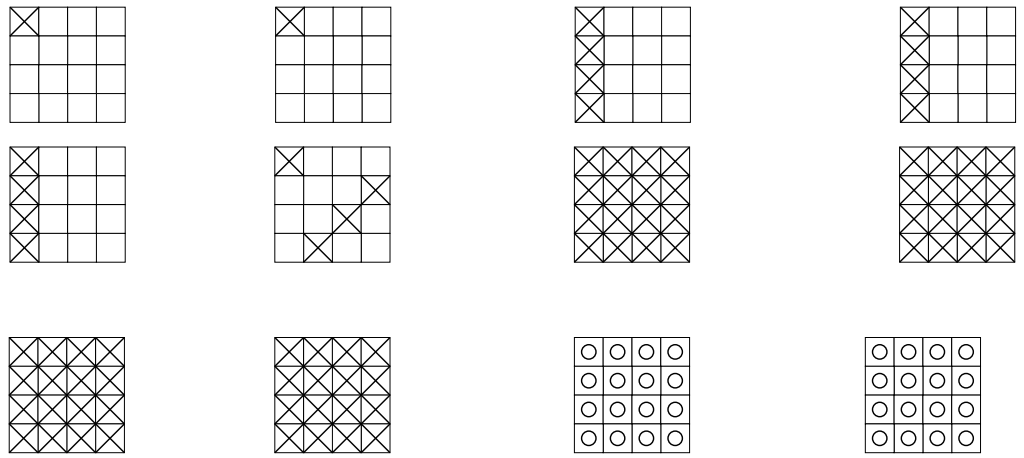
ByteSub ShiftRow MixColumn KeyAdd



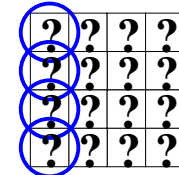
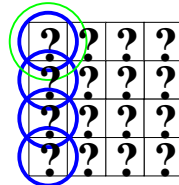
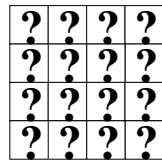
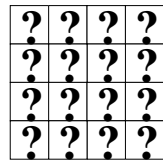
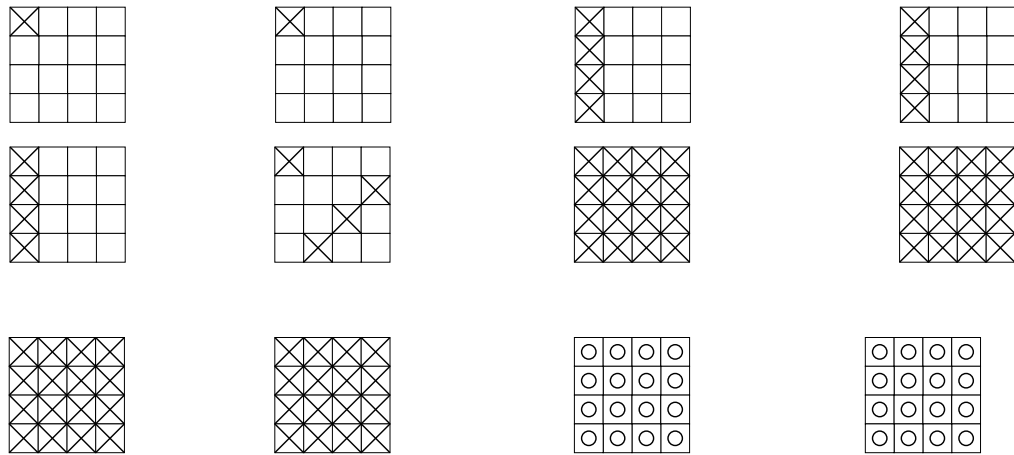
ByteSub ShiftRow MixColumn KeyAdd



ByteSub ShiftRow MixColumn KeyAdd

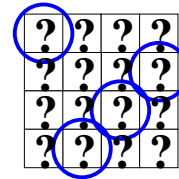
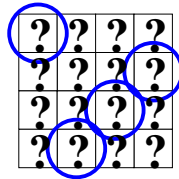
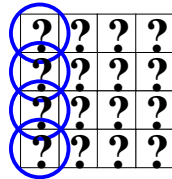


ByteSub ShiftRow MixColumn KeyAdd

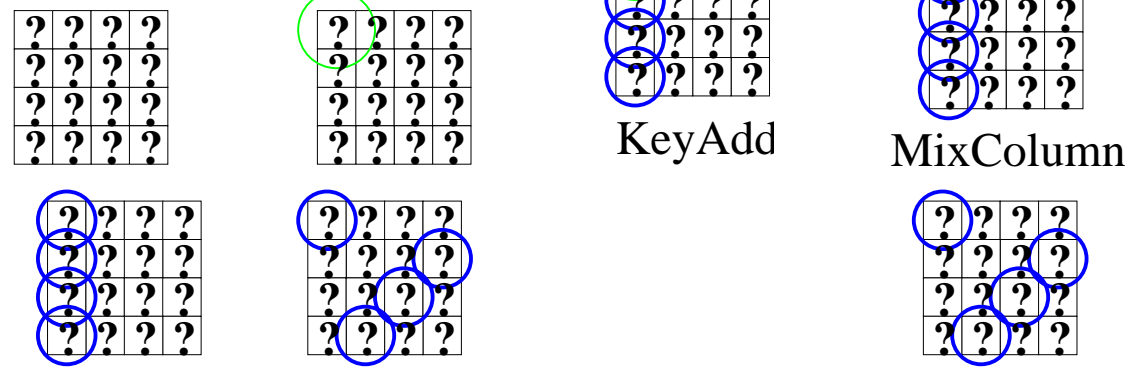
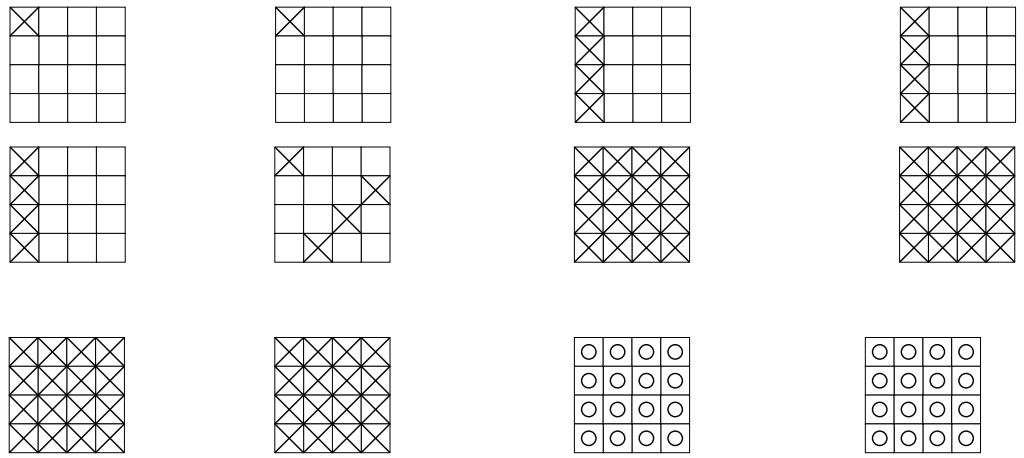


KeyAdd

MixColumn



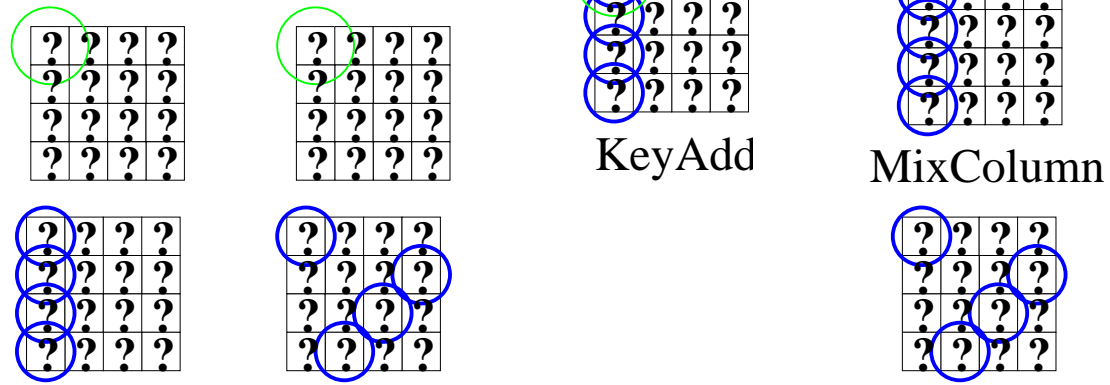
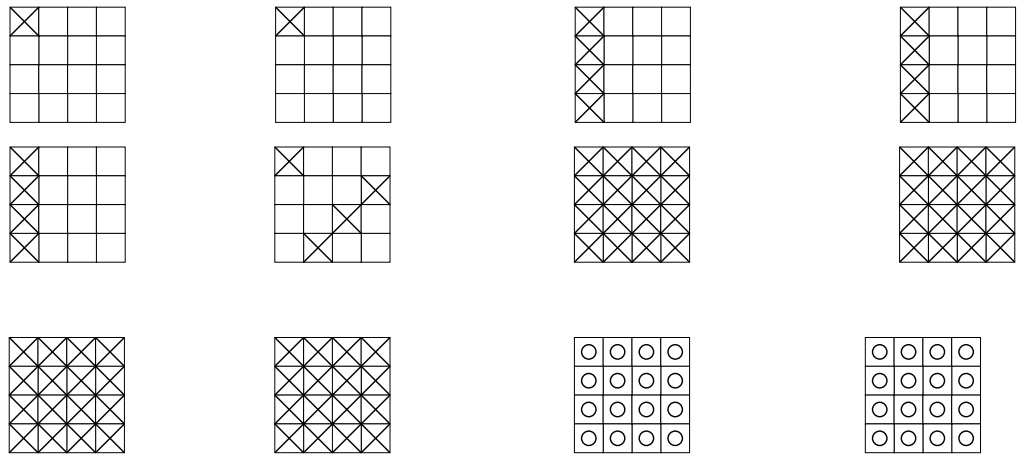
ByteSub ShiftRow MixColumn KeyAdd



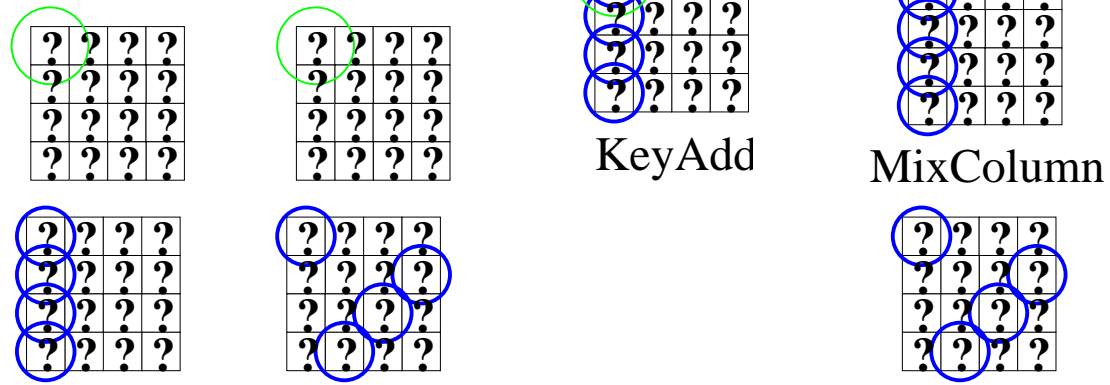
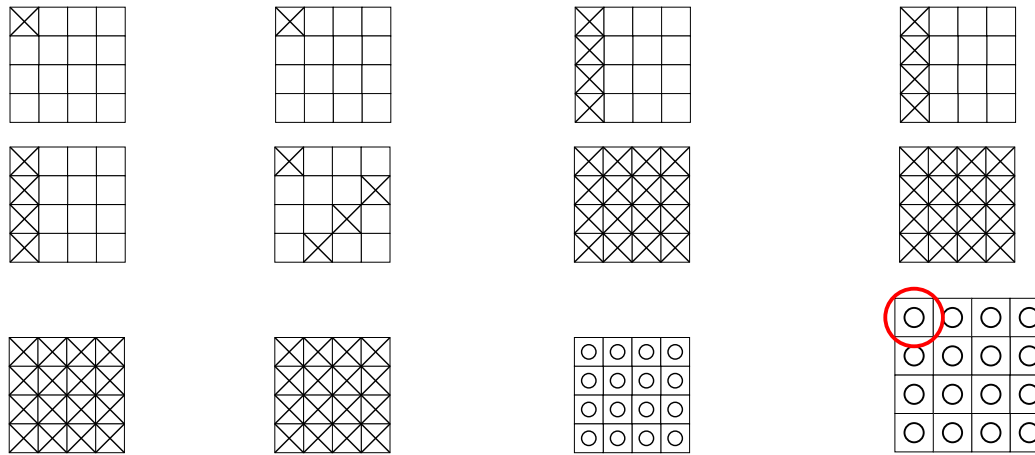
KeyAdd

MixColumn

ByteSub ShiftRow MixColumn KeyAdd



ByteSub ShiftRow MixColumn KeyAdd



KeyAdd

MixColumn

Erweiterung am Anfang

- Wähle Menge von 256 Texten, die nach einer Runde Verschlüsselung eine λ -Menge ergibt
 1. Wähle λ -Menge, rate 4 Schlüsselbytes und 'entschlüssele' sie eine Runde
 2. Wähle 2^{32} Texte, die in einer Spalte alle Werte annehmen, rate 4 Schlüsselbytes und wähle 256 Texte aus
- Verschiebung des 'Ankerpunktes' um eine Runde nach hinten

Anmerkungen zum SQUARE-Angriff

- Verwendung beider Erweiterungen \Rightarrow Angriff auf 6 Runden

# Runden	# Schlüsselbytes	# Texte	# Verschlüsselungen
4	1	$\sim 2^8$	$\sim 2^8$
5	5	$\sim 2^{32}$	$\sim 2^{40}$
6	9	$\sim 2^{32}$	$\sim 2^{72}$
6	5	$\sim 2^{32}$	$\sim 2^{45}$

Teilsummenvariante von

[Ferguson *et al.*(2000)Ferguson, Kelsey, Lucks, Schneier, Stay, Wagner & Whiting]

Ende

Danke für die Aufmerksamkeit

Literatur

[Daemen & Rijmen(1999)] J. DAEMEN & V. RIJMEN (1999). AES proposal: Rijndael.

[Ferguson *et al.*(2000)Ferguson, Kelsey, Lucks, Schneier, Stay, Wagner & Whiting]

N. FERGUSON, J. KELSEY, S. LUCKS, B. SCHNEIER, M. STAY, D. WAGNER & D. WHITING (2000). Improved Cryptanalysis of Rijndael. In *Seventh Fast Software Encryption Workshop*, 19. Springer-Verlag, Berlin, Germany / Heidelberg, Germany / London, UK / etc.