SkipJack vs. AES

Oliver Türpe

Institut für Informatik
Freie Universität Berlin

01. Juli 2010

Gliederung

1. Blockchiffren
2. WSNs
3. vorgestellt: AES
4. vorgestellt: SkipJack
5. Beispiele
6. Vergleich
7. Fazit

Blockchiffren

Im Prinzip arbeitet eine (symmetrische) Blockchiffre wie folgt:

- geheimer Schlüssel wird erstellt oder ausgewählt
- zu verschlüsselnden Daten werden im Klartext in mehrere gleich große Blöcke unterteilt
- jeder Block wird mit Schlüssel kombiniert und dann transformiert
- transformierte Blöcke bilden nun den "Geheimtext"
- Länge des Textes wird vom Chiffrieren nicht geändert
- Algorithmus invers (deshalb symmetrisches Verfahren genannt), kann mit dem gleichen Schlüssel der Geheimtext wieder lesbar gemacht werden.

- Netz von verschiedenen Sensorknoten, die per Funk kommunizieren
- infrastruktur-basiert sein oder sich selbst organisieren
- Ziel des Sensornetzes ist es jedoch, immer die Umgebung mit Sensoren zu überwachen, Ereignisse zu entdecken und zu protokollieren
- Besteht aus:
  - Mikroprozessor
  - Speichermedium
  - Funkmodul
  - Batterie

- geringe Rechenleistung
- kleiner Speicher
- problematische Energieversorgung
- hoher Verbrauch bei Übertragung

Als das NIST (U.S. National Institute of Standards and Technology (www.nist.gov)) 1997 offiziell eine Suche nach einer symmetrischen Blockchiffre, die zur Verschlüsselung von persönlichen Daten genutzt werden sollte, ausschrieb, war ein Kandidat der Rijnadael Algorithmus.

- basiert auf Bytes und Wörtern
- symmetrische Blockchiffre
- Blockgröße und Schlüssellänge unabhängig voneinander (Werte von 128, 192 oder 256 Bit)
- besteht aus mehreren Runden

Was ist eine Runde?

1. Rundenschlüssel[1] erstellen
2. Darauf folgt die Schüsseladdition (Bitweise XOR-Verknüpfung zwischen dem Klartextblockund Rundenschlüssel)
3. monoalphabetisch verschlüsselt[2].
4. Zeilen der Tabelle, die die monoalphabetische Verschlüsselung vornimmt, tauschen
5. Spalten mit einer konstanten multipliziert und anschließend per XOR verknüpft

[1] Ein Rundenschlüssel ist ein Teilschlüssel des Originalschlüssels.

[2] In einer Tabelle ist genau vorgeschrieben, welche Werte der Blöcke mit anderen getauscht werden. Vergleiche dazu die Tausch-Tabelle von SkipJack.

- Menge an Runden abhängig von der Schlüssellänge und der Blockgröße (jeweils in Bit)
- Mehr Runden bedeuten hier mehr Abstraktion.

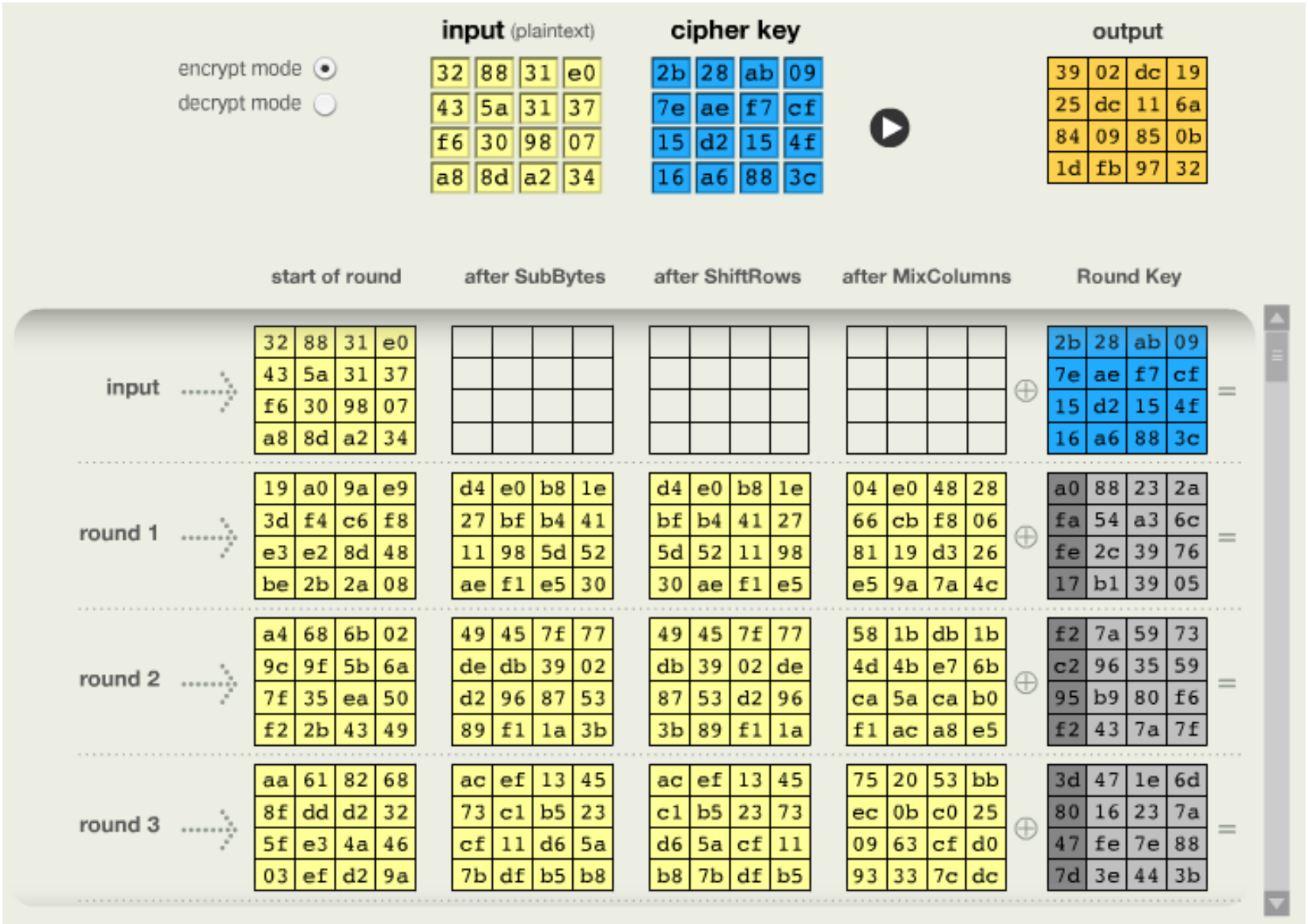
Blockgröße ($\rightarrow$) / Schlüssellänge ($\downarrow$)	128	192	256
128	10	12	14
192	12	12	14
256	14	14	14

Tabelle: Anzahl der Runden

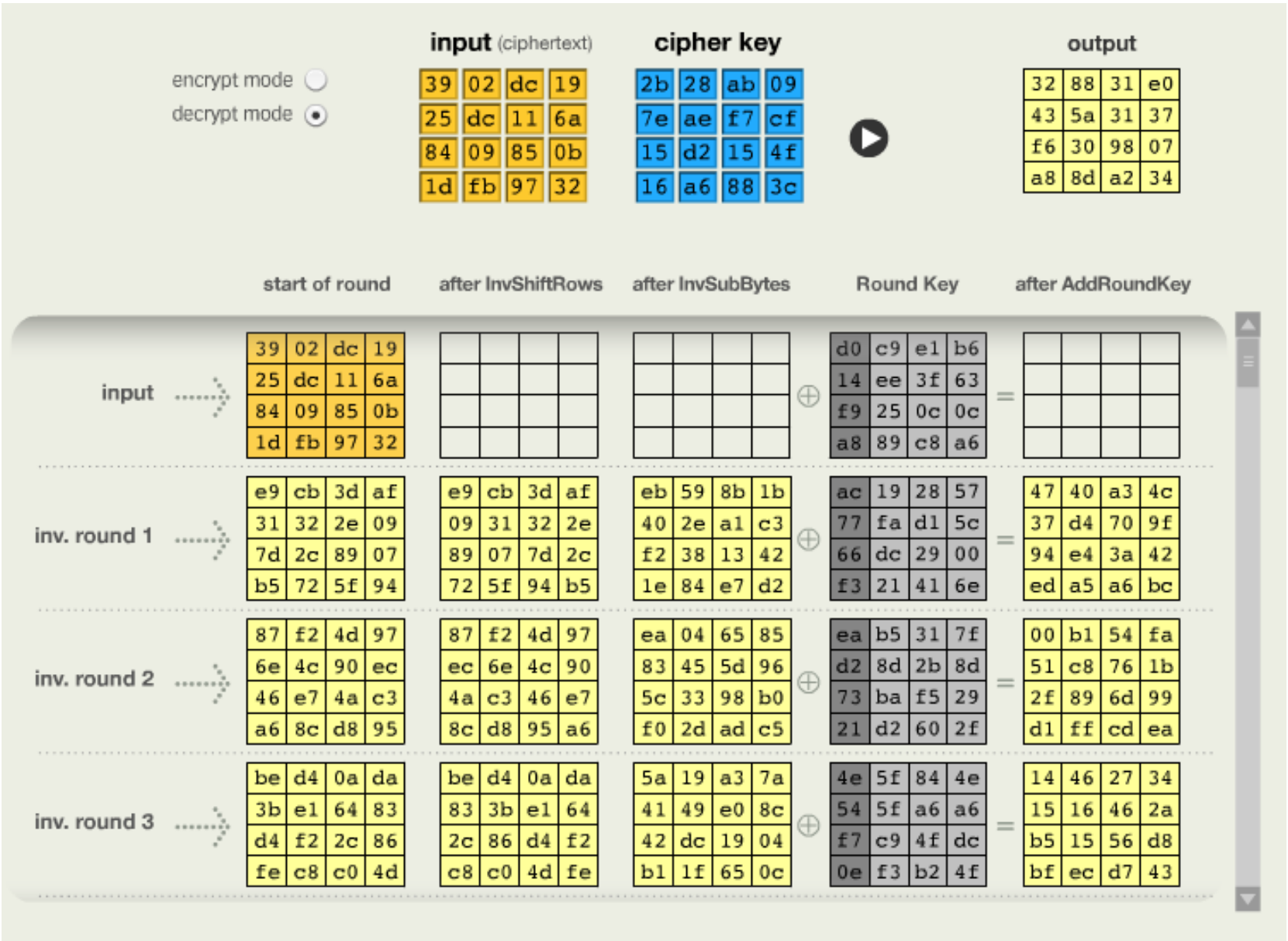
- Klartext mit 10 Runden verschlüsselt
- Angreifer muss:
  - $10*2^{32}$ Klartextblöcke sammeln
  - auf diese Blöcke dann 2^{44} Operationen anwenden
  - jetzt ist der Schlüssel ermittelt[3]
- nur einer Runde mehr, bedeutet Rechenaufwand von 2^{120}

[3] Das wären 17592186044416 Operationen, man bräuchte also mit einem Core i7 ungefähr 148082374 Stunden (ca. 16950 Jahre), da er 33 GigaFLOPS schafft.

AES
Encrypt


encrypt mode
decrypt mode
input (plaintext)
32 88 31 e0
43 5a 31 37
f6 30 98 07
a8 8d a2 34
cipher key
2b 28 ab 09
7e ae f7 cf
15 d2 15 4f
16 a6 88 3c
output
39 02 dc 19
25 dc 11 6a
84 09 85 0b
1d fb 97 32
start of round
after SubBytes
after ShiftRows
after MixColumns
Round Key
input
32 88 31 e0
43 5a 31 37
f6 30 98 07
a8 8d a2 34
2b 28 ab 09
7e ae f7 cf
15 d2 15 4f
16 a6 88 3c
round 1
19 a0 9a e9
3d f4 c6 f8
e3 e2 8d 48
be 2b 2a 08
d4 e0 b8 1e
27 bf b4 41
11 98 5d 52
ae f1 e5 30
d4 e0 b8 1e
bf b4 41 27
5d 52 11 98
30 ae f1 e5
04 e0 48 28
66 cb f8 06
81 19 d3 26
e5 9a 7a 4c
a0 88 23 2a
fa 54 a3 6c
fe 2c 39 76
17 b1 39 05
round 2
a4 68 6b 02
9c 9f 5b 6a
7f 35 ea 50
f2 2b 43 49
49 45 7f 77
de db 39 02
d2 96 87 53
89 f1 1a 3b
49 45 7f 77
db 39 02 de
87 53 d2 96
3b 89 f1 1a
58 1b db 1b
4d 4b e7 6b
ca 5a ca b0
f1 ac a8 e5
f2 7a 59 73
c2 96 35 59
95 b9 80 f6
f2 43 7a 7f
round 3
aa 61 82 68
8f dd d2 32
5f e3 4a 46
03 ef d2 9a
ac ef 13 45
73 c1 b5 23
cf 11 d6 5a
7b df b5 b8
ac ef 13 45
c1 b5 23 73
d6 5a cf 11
b8 7b df b5
75 20 53 bb
ec 0b c0 25
09 63 cf d0
93 33 7c dc
3d 47 1e 6d
80 16 23 7a
47 fe 7e 88
7d 3e 44 3b

AES
Decrypt


encrypt mode
decrypt mode
input (ciphertext)
39 02 dc 19
25 dc 11 6a
84 09 85 0b
1d fb 97 32
cipher key
2b 28 ab 09
7e ae f7 cf
15 d2 15 4f
16 a6 88 3c
output
32 88 31 e0
43 5a 31 37
f6 30 98 07
a8 8d a2 34
start of round
after InvShiftRows
after InvSubBytes
Round Key
after AddRoundKey
input
39 02 dc 19
25 dc 11 6a
84 09 85 0b
1d fb 97 32
d0 c9 e1 b6
14 ee 3f 63
f9 25 0c 0c
a8 89 c8 a6
inv. round 1
e9 cb 3d af
31 32 2e 09
7d 2c 89 07
b5 72 5f 94
e9 cb 3d af
09 31 32 2e
89 07 7d 2c
72 5f 94 b5
eb 59 8b 1b
40 2e a1 c3
f2 38 13 42
1e 84 e7 d2
ac 19 28 57
77 fa d1 5c
66 dc 29 00
f3 21 41 6e
47 40 a3 4c
37 d4 70 9f
94 e4 3a 42
ed a5 a6 bc
inv. round 2
87 f2 4d 97
6e 4c 90 ec
46 e7 4a c3
a6 8c d8 95
87 f2 4d 97
ec 6e 4c 90
4a c3 46 e7
8c d8 95 a6
ea 04 65 85
83 45 5d 96
5c 33 98 b0
f0 2d ad c5
ea b5 31 7f
d2 8d 2b 8d
73 ba f5 29
21 d2 60 2f
00 b1 54 fa
51 c8 76 1b
2f 89 6d 99
d1 ff cd ea
inv. round 3
be d4 0a da
3b e1 64 83
d4 f2 2c 86
fe c8 c0 4d
be d4 0a da
83 3b e1 64
2c 86 d4 f2
c8 c0 4d fe
5a 19 a3 7a
41 49 e0 8c
42 dc 19 04
b1 1f 65 0c
4e 5f 84 4e
54 5f a6 a6
f7 c9 4f dc
0e f3 b2 4f
14 46 27 34
15 16 46 2a
b5 15 56 d8
bf ec d7 43

- kompletter Algorithmus in Formel mit 2^{50} Termen
- zeigt Nachvollziehbarkeit und Einfachheit des Algorithmus
- Laufzeit?
  - Vertauschen der Spalten in einer Matrix durch Schleifen in zwei-dimensionalen Array
  - dauert länger oder speicherintensiv
- Vertreter: IEEE 802.11i, FileVault und Microsoft .NET

Das symmetrische Verschlüsselungsverfahren SkipJack wurde von der National Security Agency (NSA) entwickelt. Nach einer langen Phase der Geheimhaltung wurde am 24. Juni 1998 die Spezifikation des Algorithmus veröffentlicht.

- Länge des Schlüssels beträgt genau 80 Bit
- Blockgröße 64 Bit
- immer 8 Byte an Datenblöcken verschlüsselt
- abwechselnd zwei Regeln eingesetzt

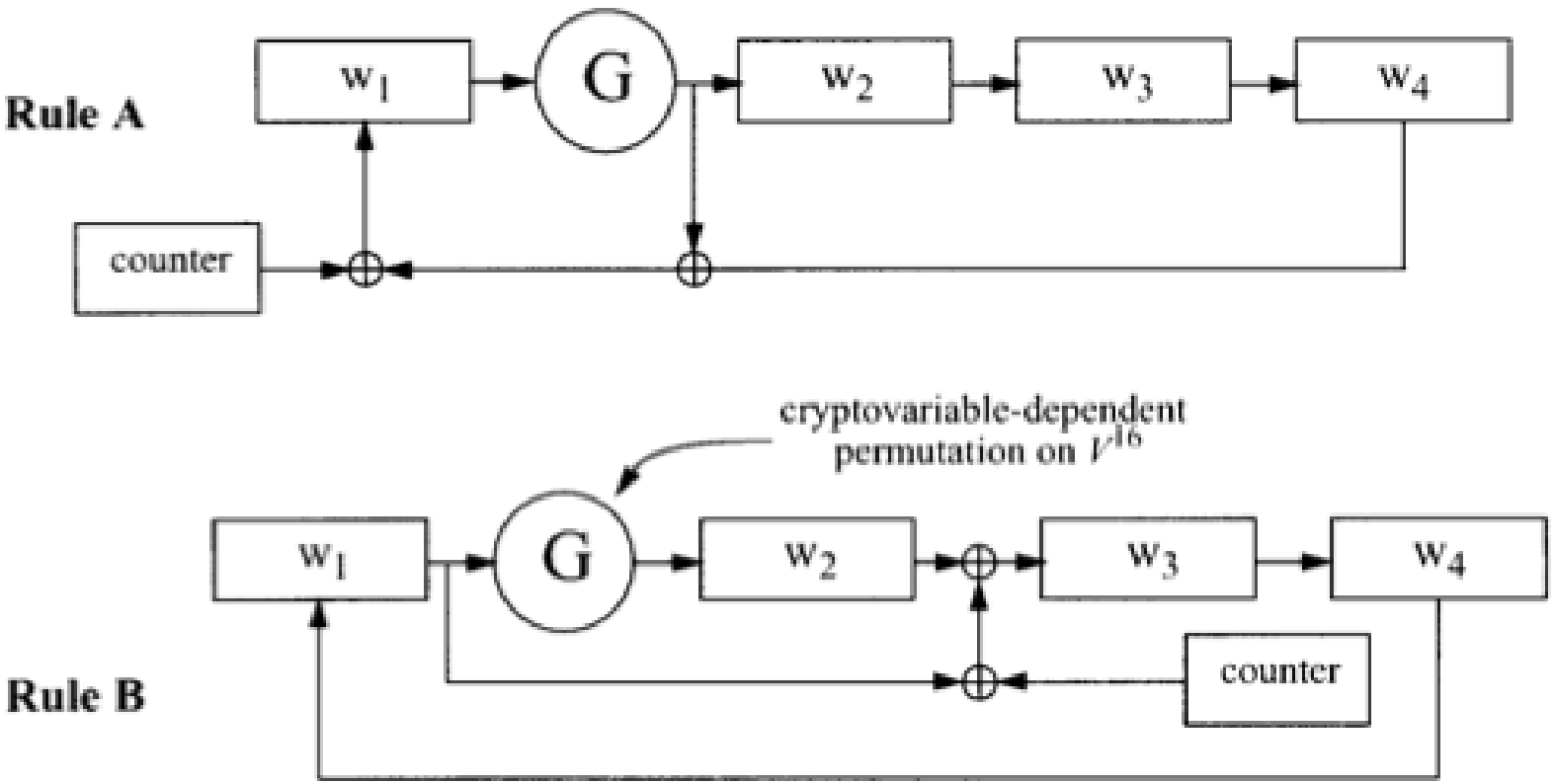


Abbildung: Visualisierung der Regeln

Der Algorithmus benötigt insgesamt 32 Schritte.

- **Verschlüsseln:**
  - Die Eingabe ist: w_i^0, $1 \leq i \leq 4$.
  - Starte nun den Zähler (counter) mit 1.
  - Führe 8 mal Regel A durch, danach 8 mal Regel B.
  - Nun wieder 8 mal Regel A, gefolgt von 8 mal Regel B.
  - Die Verschlüsselung ist nun abgeschlossen.
  - Nach jedem Schritt wird der Zähler dabei um eins erhöht.
  - Die Ausgabe ist dann: w_i^{32}, $1 \leq i \leq 4$.
- **Entschlüsseln:**
  - Die Eingabe ist: w_i^{32}, $1 \leq i \leq 4$.
  - Wir starten mit einem Zähler von 32.
  - Nun wird 8 mal Regel B^{-1} ausgeführt gefolgt von 8 mal Regel A^{-1}.
  - Dieser Schritt wiederholt sich dann.
  - Der Zähler wird hierbei jedes Mal um eins vermindert.
  - Als Ausgabe erhalten wir dann: w_i^0, $1 \leq i \leq 4$.

Skipjack
Tausch-Tabelle

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	a3	d7	09	83	f8	48	f6	f4	b3	21	15	78	99	b1	af	f9
1x	e7	2d	4d	8a	ce	4c	ca	2e	52	95	d9	1e	4e	38	44	28
2x	0a	df	02	a0	17	f1	60	68	12	b7	7a	c3	e9	fa	3d	53
3x	96	84	6b	ba	f2	63	9a	19	7c	ae	e5	f5	f7	16	6a	a2
4x	39	b6	7b	0f	c1	93	81	1b	ee	b4	1a	ea	d0	91	2f	b8
5x	55	b9	da	85	3f	41	bf	e0	5a	58	80	5f	66	0b	d8	90
6x	35	d5	c0	a7	33	06	65	69	45	00	94	56	6d	98	9b	76
7x	97	fc	b2	c2	b0	fe	db	20	e1	eb	d6	e4	dd	47	4a	1d
8x	42	ed	9e	6e	49	3c	cd	43	27	d2	07	d4	de	c7	67	18
9x	89	cb	30	1f	8d	c6	8f	aa	c8	74	dc	c9	5d	5c	31	a4
Ax	70	88	61	2c	9f	0d	2b	87	50	82	54	64	26	7d	03	40
Bx	34	4b	1c	73	d1	c4	fd	3b	cc	fb	7f	ab	e6	3e	5b	a5
Cx	ad	04	23	9c	14	51	22	f0	29	79	71	7e	ff	8c	0e	e2
Dx	0c	ef	bc	72	75	6f	37	a1	ec	d3	8e	62	8b	86	10	e8
Ex	08	77	11	be	92	4f	24	c5	32	36	9d	cf	f3	a6	bb	ac
Fx	5e	6c	a9	13	57	25	b5	e3	bd	a8	3a	01	05	59	2a	46

Abbildung: Die Tausch Tabelle **F**

I n f o r m a t i k[4] $\rightarrow^{\alpha}$ 49 6E 66 6F 72 6D 61 74 69 $6B_{16}$ $\rightarrow^{F}$ B4 9B 65 76 B2 98 D5 B0 00 56 $\rightarrow^{F}$ D1 C9 06 DB 1C C8 6F 34 A3 BF ...

[4] α ist hier die Funktion, die jedes Zeichen eines Wortes in seinen hexadezimalen Wert laut der ASCII Tabelle umwandelt.

- SkipJack:
  - von NSA erfunden
  - funktional schneller, da nur XOR
  - geringer Speicherverbrauch
  - kurzer Schlüssel, deswegen häufig zu wechseln
  - mehr Traffic
- AES:
  - Professoren
  - Schleifen dadurch längere Laufzeit
  - variiert je nach Implementierung
  - Schlüssel muss nicht gewechselt werden, wenn hinreichend groß
  - weniger Traffic durch entfallenden Schlüsselwechsel

Kriterium	SkipJack	AES
Laufzeit der Berechnung	+	-
Speicherverbrauch	+	-
Betriebszeit	-	+
Energieverbrauch	-	+
Traffic	-	+
"Ruf"	0	+

Tabelle: Kriterien im Vergleich

Was ist besser geeignet?

- AES
  - da variable Schlüssel und Blockgröße
  - durch zwei Tabellen schnelles Tauschen zu realisieren
  - kurze Funkübertragungen