Darstellung der von Akkard und Giraud in ”An Implementation of DES and AES, Secure against Some Attacks” entwickelten Maßnahmen zum Schutz vor Side-Channel-Analyse

Samuel Hetterich

Sommerakademie La Colle-Sur-Loup

7. Oktober 2009

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

In dem Paper von Akkard und Giraud werden Gegenmaßnahmen gegen DPA für DES- und AES-Implementierungen dargestellt.

Bemerkung (DPA Differential Power Analyse)

Anhand von Korrelation von Stromverbrauchsberechnung und gemessenem Stromverbrauch wird auf Korrektheit einer Schlüsselhypothese geschlossen.

Funktionsweise der DPA

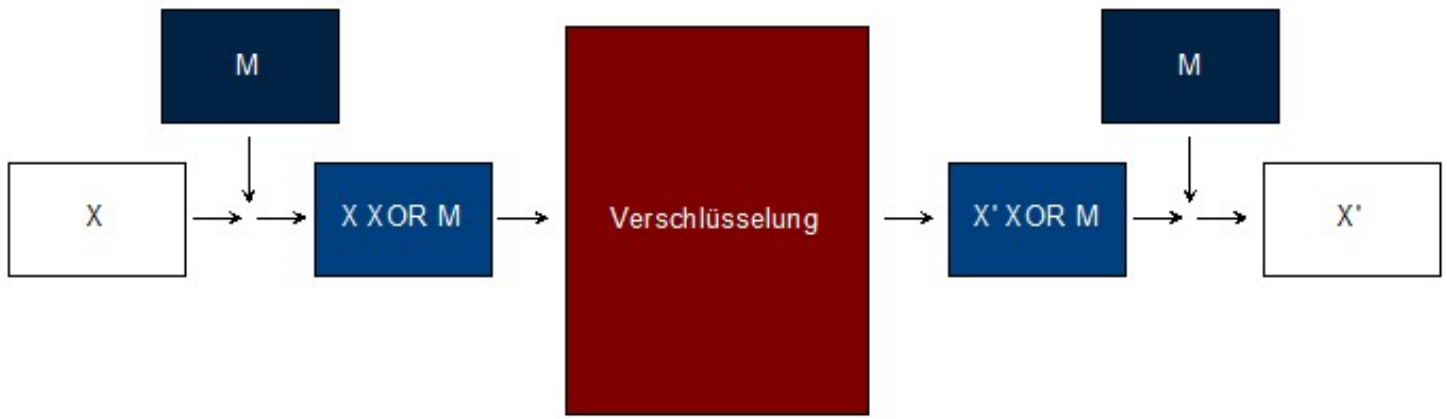
- Mehrfache Verschlüsselung eines Plaintextes X unter einer Schlüsselhypothese.
- Berechnung einer Prognose für Stromverbrauchs der Verschlüsselung berechnet(abhängig vom Plaintext und Schlüsselhypothese).
- Durchführung der Verschlüsselung und Ermittlung des Stromverbrauchs.
- Die Korrelation zwischen Prognose des Stromverbrauchs und gemessenen Stromverbrauchs führt zum Auffinden von Targetbits aus welchen sich der Schlüssel berechnen lässt.

Gegenmaßnahmen gegen DPA

- Einfügen von ”dummy instructions”.
- Randomization von Operationen.
- Transformation der Daten (Duplication Method).
- Maskierung der Daten.
- Kein Rauschen, denn es kann durch eine größere Anzahl von Messungen kompensiert werden!

Transformed Masking Method

Zum Einsatz kommt eine zufällige Maske, welche über die Dauer des gesamten Algorithmus den Plaintext maskiert. An keiner Stelle darf der Plaintext ohne Maske in eine Berechnung eingehen.



Transformed Masking Method

Ist eine einfache Maskierung überhaupt realisierbar?

Grundlegend zwei einfache mögliche Maskierungen:

- Boolean Maske - additiv im $\mathbb{F}_2$ (Bitweise). Anwendung bei linearen Teilen eines Algorithmus:

Linearität

$f(M \oplus X) = f(M) \oplus f(X)$ im $\mathbb{F}_2 \Rightarrow f(X)$ ist leicht und schnell zu berechnen.

- Multiplikative Maske - multiplikativ im $\mathbb{F}_{2^8}$ (Byteweise). Anwendung in nichtlinearen Teilen eines Algorithmus:

Multiplikativität

$f(M \otimes X) = f(M) \otimes f(X)$ im $\mathbb{F}_{2^8} \Rightarrow f(X)$ ist leicht und schnell zu berechnen.

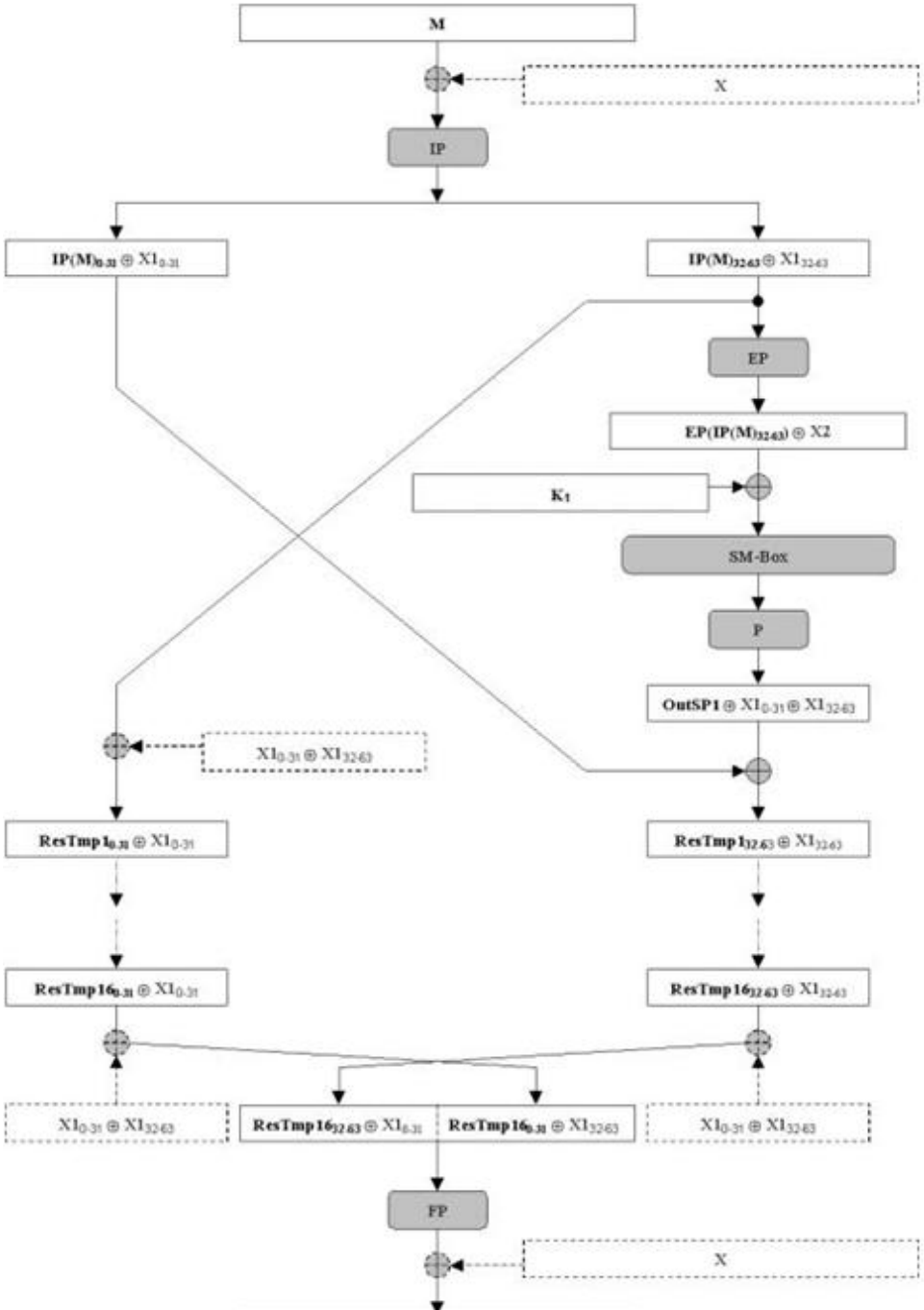
Transformed Masking Method

In der Praxis wird der Plaintext X mit einer zufälligen Boolean Maske M XOR verknüpft. Diese wird für die nichtlinearen Teile in eine multiplikative Maske transformiert (AES) oder die nichtlinearen Teile werden modifiziert (DES).

- Der nichtlineare Teil des DES ist die S-Box.
- Der nichtlineare Teil des AES ist die ByteSub-Funktion.

Implementierung des DES mit TMM

Auf der nächsten Folie ist ein modifizierter DES zu sehen. Er unterscheidet sich einzig durch eine Modifizierte S-Box (genannt SM-Box) vom herkömmlichen DES.


M
X
IP
IP(M)0-31 ⊕ X10-31
IP(M)32-63 ⊕ X132-63
EP
EP(IP(M)32-63) ⊕ X2
K1
SM-Box
P
OutSP1 ⊕ X10-31 ⊕ X132-63
X10-31 ⊕ X132-63
ResTmp10-31 ⊕ X10-31
ResTmp132-63 ⊕ X132-63
ResTmp160-31 ⊕ X10-31
ResTmp1632-63 ⊕ X132-63
X10-31 ⊕ X132-63
ResTmp1632-63 ⊕ X10-31
ResTmp160-31 ⊕ X132-63
X10-31 ⊕ X132-63
FP
X

Implementierung des DES mit TMM

SM-Box als Modifizierung der S-Box

$$SM - Box(A) = S - Box(A \oplus X2) \oplus P^{-1}(X1_{0-31} \oplus X1_{32-63})$$

Bemerkung (SM-Box)

- *Die SM-Box ändert den DES nicht. Ihr liegt die S-Box zugrunde.*
- *Die SM-Box wird für jede Verschlüsselung neu berechnet.*

Bemerkung (Implementierung des DES mit TMM)

Die Zwischenwerte X, X1 und X2 müssen sehr sicher berechnet werden! (randomized bit-per-bit calculation) Dies ist teuer, aber nur einmal zu berechnen und kann dann für jede Runde des DES verwendet werden.

Praxis DES

Messung mit 32-bit risc CPU in C

Type of DES	Timing at 5 Mhz	ROM (bytes)	RAM (bytes)
Normal DES	9.4 ms	1540	42
DES (TMM)	21.2 ms	2656	452

Messung mit 32-bit risc CPU, assembly code

Type of DES	Timing at 5 Mhz	ROM (bytes)	RAM (bytes)
Normal DES	46.2 μs	596	16
DES (TMM)	237.6 μs	2017	272

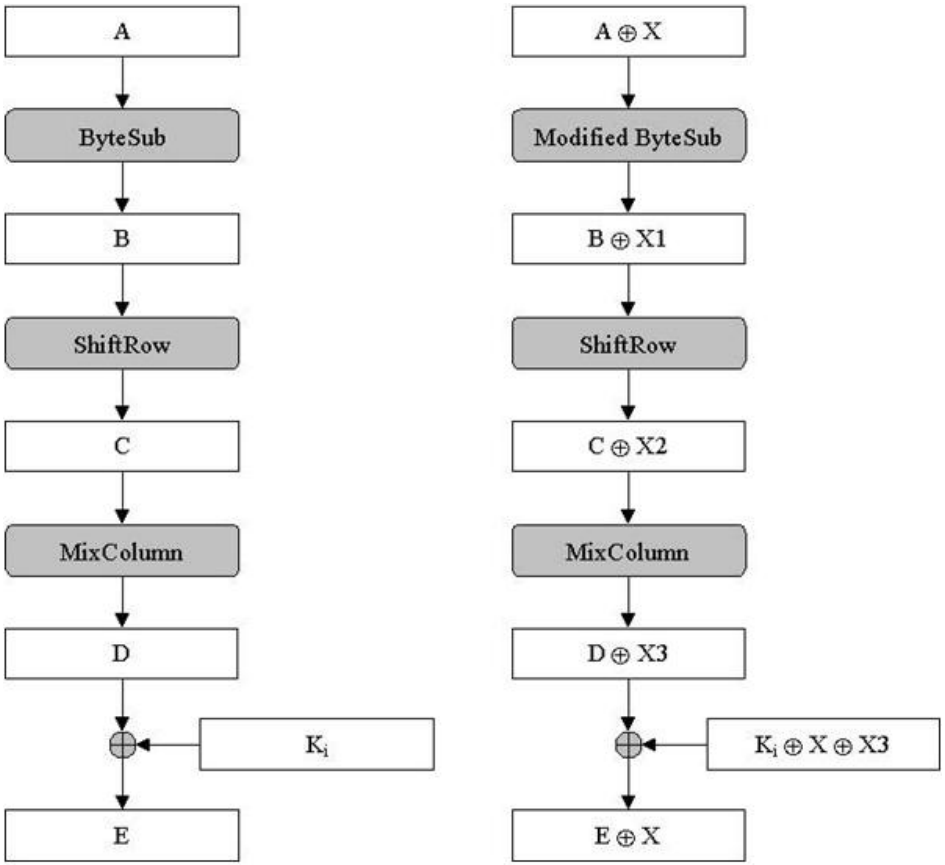
Implementierung des AES mit TMM

Bei der AES-Implementierung wird die Transformed Masking Method besonders deutlich.

Die linearen Teile des Algorithmus werden mit einer Boolean Maske maskiert. Für die ByteSub-Funktion muss diese in eine multiplikative Maske transformiert werden.

Auf der nächsten Folie ist eine Gegenüberstellung einer AES-Runde in herkömmlicher und modifizierter Form gegeben.

Implementierung des AES mit TMM

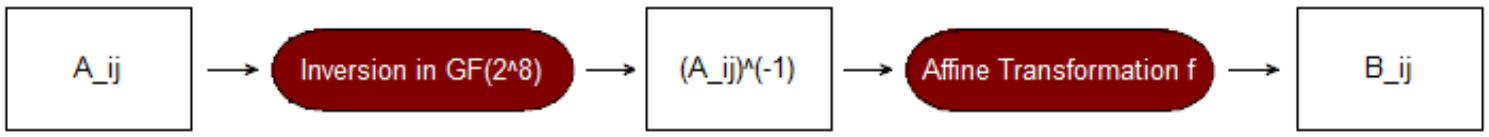


Bemerkung (Notation)

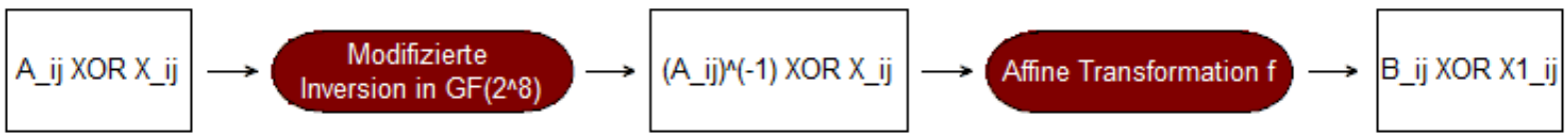
- *X steht für die verwendete Maske;*
- *X1 = f1(X), mit f1 als dem linearen Teil der affinen Transformation f der ByteSub-Funktion;*
- *X2 = ShiftRow(X1);*
- *X3 = MixColumn(X2);*
- *K_i ist der Rundenschlüssel der Runde i.*

Implementierung des AES mit TMM

Die Allgemeine Form der ByteSub-Funktion:

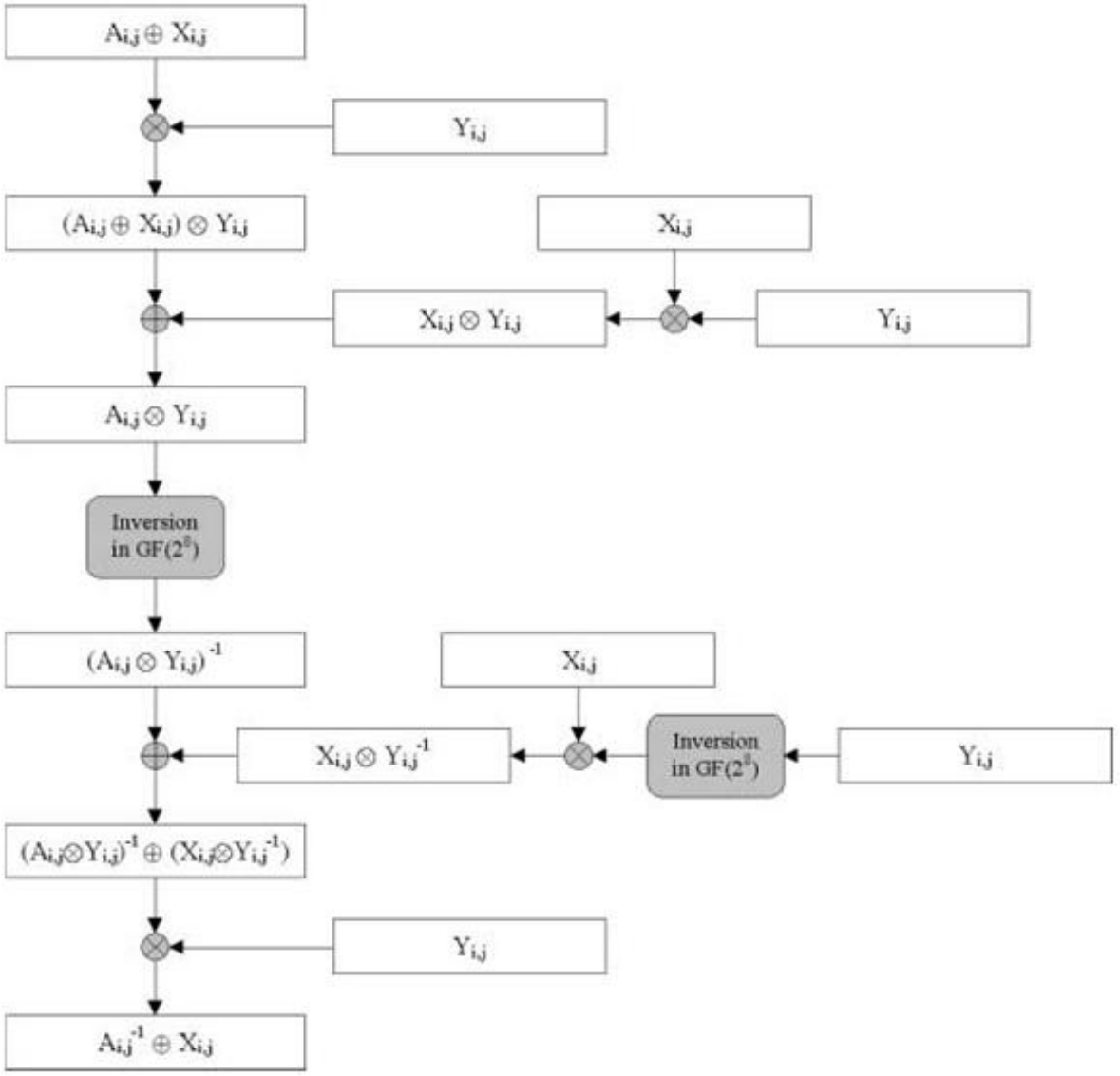


Die modifizierte ByteSub-Funktion:



Implementierung des AES mit TMM

Modifizierte Inversion:



Implementierung des AES mit TMM

Die in der Grafik Dargestellte modifizierte Inversion ist im wesentlichen eine Berechnung im $GF(2^8)$ bezüglich des irreduziblen Polynoms $m(x) = x^8 + x^4 + x^3 + x + 1$ der Form:

$$((A_{ij} \oplus X_{ij}) \otimes Y_{ij}) \oplus (X_{ij} \otimes Y_{ij}) = A_{ij} \otimes Y_{ij} \tag{1}$$

und

$$((A_{ij} \otimes Y_{ij})^{-1} \oplus (X_{ij} \otimes Y_{ij}^{-1})) \otimes Y_{ij} \tag{2}$$

$$= ((A_{ij}^{-1} \otimes Y_{ij}^{-1}) \oplus (X_{ij} \otimes Y_{ij}^{-1})) \otimes Y_{ij} \tag{3}$$

$$= A_{ij}^{-1} \oplus X_{ij} \tag{4}$$

Bemerkung (Berechnung)

- *An keiner Stelle der Maskentransformation geht der Plaintext ohne Maske in eine Berechnung ein.*

Implementierung des AES mit TMM

Bemerkung (AES-Implementierung)

- *Man könnte wie beim DES eine modifizierte S-Box berechnen, jedoch beinhaltet die S-Box des AES 256 Bytes, welche den zu Verfügung stehenden Ram in Smartcards übersteigt.*
- *Das die Null bei der Inversion im $GF(2^8)$ auf die Null abgebildet wird bietet die Möglichkeit eines Angriffes.*

Praxis AES

Verschlüsselung einer 128-bit Nachricht mit einem 128-bit Schlüssel (assembly code, 8-bit CPU)

Type of AES	Timing at 5 Mhz	ROM (bytes)	RAM (bytes)
Normal AES	18.1 ms	730	41
AES (TMM)	58.7 ms	1752	121

Schlussfolgerung

- Die Transformed Masking Method bietet ausreichned Schutz gegen DPA, nicht jedoch gegen HODPA.
- In dem 2003 von Elena Trichina, Domenico De Seta und Lucia Germani veröffentlichten Paper „Simplified Adaptive Multiplicative Masking for AES“ wird eine einfachere Maskentransformation für den AES vorgestellt.

Quellen

- Mehdi-Laurent Akkar und Christophe Giraud. An Implementation of DES and AES, Secure against Some Attacks. 2001
- Andreas Hoheisel. Side-Channel Analysis Resistant Implementation of AES on Automotive Processors. Master Thesis, Ruhr-University Bochum. 2009