Ligthweight Cryptography

Marc Beunardeau

April 29, 2015

Table of contents

Introduction

Software Requirements

State of the Art
- Trivium
- PRESENT
- PRINCE

PRIDE
- Presentation
- The Linear Layer

Differential Attack
- Differential Analysis
- Attack

SPECK
- Presentation

Fault Attack
- Bit-Flip Attack
- Random Bit Fault

Itroduction

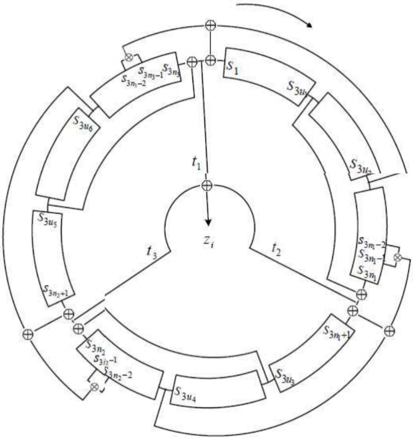
- Developpement of tiny devices (RFID,wireless sensors....)
- Need for new algorithms ($\neq$ AES)
- Pervasive environement (invasive attacks)

- Clock Cycles per encryption
- Memory
- Security
- Consumption

Generalities

- Introduced by Cannière and Preneel in 2005
- Stream cipher
- 1100 cycles for initialisation
- 1 cycle per bits
- 2 faults attack
- optimized for hardware

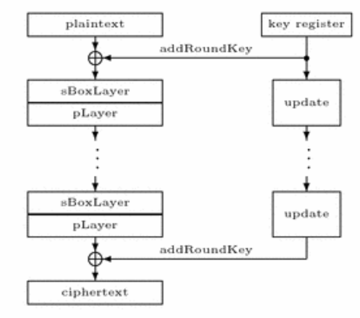
Structure


s_1
s_{3u_1}
s_{3u_2}
s_{3n_1-2}
s_{3n_1-1}
s_{3n_1}
s_{3n_1+1}
s_{3u_3}
s_{3u_4}
s_{3n_2}
s_{3n_2-1}
s_{3n_2-2}
s_{3n_2+1}
s_{3u_5}
s_{3u_6}
s_{3n_3-2}
s_{3n_3-1}
s_{3n_3}
t_1
t_2
t_3
z_i

Generalities

- Bogdanov & Al in 2007
- SP-network
- 32 rounds
- 80, 128 bits keys, 64 bits block
- 32 cycles per block (hardware implementation)
- Cube attack : 2^{15} chosen plain text, 2^{32} encryption
- optimized for hardware

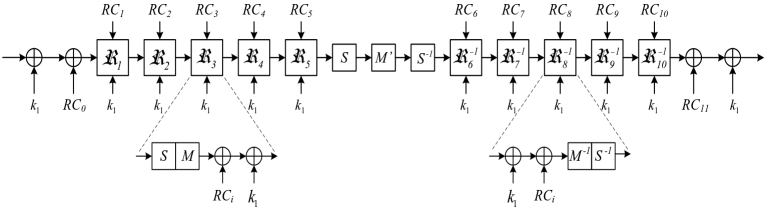
Structure


plaintext
key register
addRoundKey
sBoxLayer
pLayer
update
sBoxLayer
pLayer
update
addRoundKey
ciphertext

Generalities

- Introduced by
- SP-network
- Low latency
- Small aera when fully unrolled
- 128 bits key, 64 bits block
- 1 cycle per block (unrolled hardware implementation)
- α - reflection : $Dec_{(k_0||k'_0||k_1)}(.) = Enc_{(k'_0||k_0||k_1 \oplus \alpha)}(.)$
- 3-4 faults attack

Structure


RC_1
RC_2
RC_3
RC_4
RC_5
RC_6
RC_7
RC_8
RC_9
RC_{10}
$\mathfrak{R}_1$
$\mathfrak{R}_2$
$\mathfrak{R}_3$
$\mathfrak{R}_4$
$\mathfrak{R}_5$
S
M'
S^{-1}
$\mathfrak{R}_6^{-1}$
$\mathfrak{R}_7^{-1}$
$\mathfrak{R}_8^{-1}$
$\mathfrak{R}_9^{-1}$
$\mathfrak{R}_{10}^{-1}$
k_1
RC_0
RC_{11}
S M
RC_i
M^{-1} S^{-1}

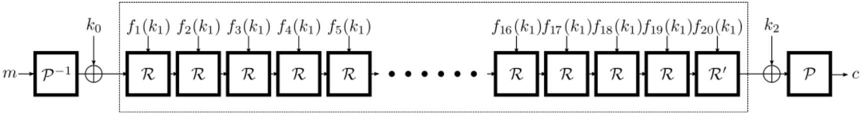
Generalities

- Introduced by Albrecht & Al in 2014
- SPN block cipher with focus on linear layer
- 64 bits blocks
- 128bits key
- 20 rounds

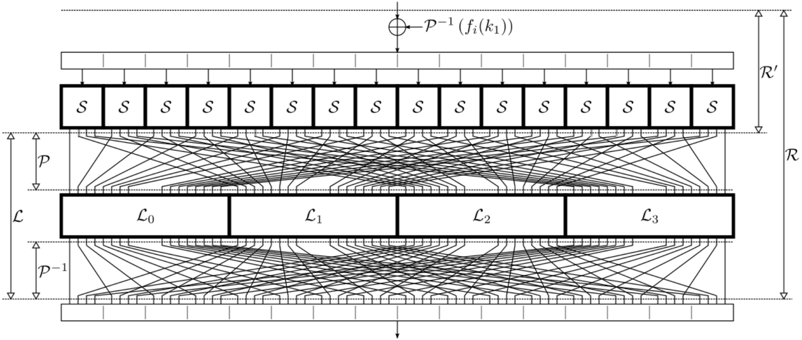
Performances

- 68 cycles per block
- 138 bytes of flash memory (943 flash + 33 S-RAM bytes, and 575 cycles for AES)

Structure


m
$\mathcal{P}^{-1}$
k_0
$f_1(k_1)$
$f_2(k_1)$
$f_3(k_1)$
$f_4(k_1)$
$f_5(k_1)$
$\mathcal{R}$
$f_{16}(k_1)$
$f_{17}(k_1)$
$f_{18}(k_1)$
$f_{19}(k_1)$
$f_{20}(k_1)$
$\mathcal{R}'$
k_2
$\mathcal{P}$
c

A Round of PRIDE



Key Scheduling

- $k = k_0 || k_1$
- $k_1 = k_{1_0} || k_{1_1} || k_{1_2} || k_{1_3} || k_{1_4} || k_{1_5} || k_{1_6} || k_{1_7}$
- $f_i(k_1) = k_{1_0} || g_i^{(0)}(k_{1_1}) || k_{1_2} || g_i^{(1)}(k_{1_3}) || k_{1_4} || g_i^{(2)}(k_{1_5}) || k_{1_6} || g_i^{(3)}(k_{1_7})$
- $g_i^j(x) = x + i \times C_j \mod 256$

S-boxes

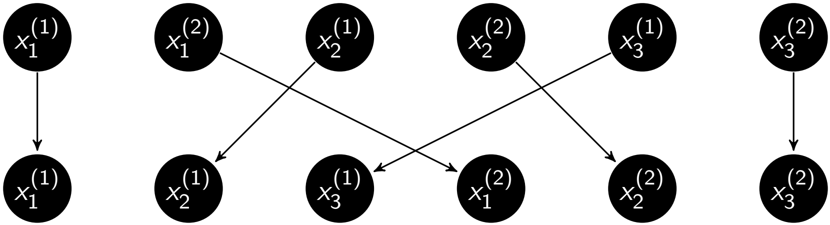
- Involution
- Differential : $1/4$
- Linear : $1/2$
- 20 Clock cycles

Interleaving

$$
\begin{aligned}
P^n_{b_1,\dots b_k} : (\mathbb{F}_2^{b_1} \times \mathbb{F}_2^{b_2} \times \dots \mathbb{F}_2^{b_k})^n &\longrightarrow (\mathbb{F}_2^{b_1})^n \times (\mathbb{F}_2^{b_2})^n \dots \times (\mathbb{F}_2^{b_k})^n \\
(x_1, \dots, x_n) &\longrightarrow ((x_1^{(1)}, \dots, x_n^{(1)}), \dots, (x_1^{(k)}, \dots, x_n^{(k)}))
\end{aligned}
$$

where $x_i = (x_i^{(1)}, \dots, x_i^{(k)})$ with $x_i^{(j)} \in \mathbb{F}_2^{b_j}$

Example $k = 2$, $n = 3$



Interleaving

- $G_i = [I|L_i^T]$ matrix generator of a $(2n, 2^n)$ code of minimal distance d_i over $\mathbb{F}_2$
- $L := P^{-1} \circ (L_1 \times L_2 \times L_3 \times L_4) \circ P$
- $[I|L^T]$ matrix generator of a $(2n, 2^n)$ code of minimal distance $mind_i$ over $\mathbb{F}_2^4$

Finding the Linear Layer

- Set $n = 64$, $k = 4$, $b_i = 1$
- Look for $L_0 ... L_3 \in \mathcal{M}_{16}(\mathbb{F}_2)$ with branch number 4 and achieving high depdencie.
- Set a set of assembly instruction
- Check after N instruction if the matrix fulffil our criteria for L_0 ($N = 7$ achieved)
- Derive $L_i = PL_{i-1}Q$ with P, Q permutation (found with Constraint Integer Programing) and the density of $L_i \vee L_{i-1}$ is maximum

Principle

- Find differential characteristics :
- $\Delta X = X_1 \oplus X_2$ a constant
- $\Delta Y = Encr(X_1) \oplus Encr(X_2) = cst$ for a high number($>> 1/2^{|K|}$) of pair (X_1, X_2)
- Retrieve information on the key

Notations

- I_r input of the r-th round
- X_r after the round key additional of the r-th round
- Y_r after the S-box layer of the r-th round
- Z_r after the permutation of the r-th round
- W_r after the matrix of the r-th round
- O_r the output of the r-th round
- $X[n_1, n_2...]$ the $n_1, n_2..$ nibbles of state X

S-Boxes

	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xa	0xb	0xc	0xd	0xe	0xf
0x0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1	0	0	0	0	4	4	4	4	0	0	0	0	0	0	0	0
0x2	0	0	0	0	0	0	0	0	4	0	0	4	2	2	2	2
0x3	0	0	0	0	0	0	0	0	4	0	0	4	2	2	2	2
0x4	0	4	0	0	0	0	4	0	0	2	2	0	2	0	0	2
0x5	0	4	0	0	0	4	0	0	0	2	2	0	2	0	0	2
0x6	0	4	0	0	4	0	0	0	0	2	2	0	0	2	2	0
0x7	0	4	0	0	0	0	0	4	0	2	2	0	0	2	2	0
0x8	0	0	4	4	0	0	0	0	4	0	4	0	0	0	0	0
0x9	0	0	0	0	2	2	2	2	0	0	0	0	2	2	2	2
0xa	0	0	0	0	2	2	2	2	4	0	4	0	0	0	0	0
0xb	0	0	4	4	0	0	0	0	0	0	0	0	2	2	2	2
0xc	0	0	2	2	2	2	0	0	0	2	0	2	2	0	2	0
0xd	0	0	2	2	0	0	2	2	0	2	0	2	0	2	0	2
0xe	0	0	2	2	0	0	2	2	0	2	0	2	2	0	2	0
0xf	0	0	2	2	2	2	0	0	0	2	0	2	0	2	0	2

2 Rounds Characterisitcs

ΔI_r	0x0	0x8	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
ΔX_r	0x0	0x8	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
ΔY_r	0x0	0x8	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
ΔZ_r	0x4	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
ΔW_r	0x0	0x4	0x4	0x4	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
ΔI_{r+1}	0x0	0x0	0x0	0x0	0x0	0x8	0x0	0x0	0x0	0x8	0x0	0x0	0x0	0x8	0x0	0x0
ΔX_{r+1}	0x0	0x0	0x0	0x0	0x0	0x8	0x0	0x0	0x0	0x8	0x0	0x0	0x0	0x8	0x0	0x0
ΔY_{r+1}	0x0	0x0	0x0	0x0	0x0	0x8	0x0	0x0	0x0	0x8	0x0	0x0	0x0	0x8	0x0	0x0
ΔZ_{r+1}	0x0	0x4	0x4	0x4	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
ΔW_{r+1}	0x4	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
ΔI_{r+2}	0x0	0x8	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0

Differential Analysis

ΔI_1	0000 0000 0000 0000 0000 ???? 0000 0000 0000 ???? 0000 0000 0000 ???? 0000 0000
ΔX_1	0000 0000 0000 0000 0000 ???? 0000 0000 0000 ???? 0000 0000 0000 ???? 0000 0000
ΔY_1	0000 0000 0000 0000 0000 1000 0000 0000 0000 1000 0000 0000 0000 1000 0000 0000
ΔZ_1	0000 0100 0100 0100 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
ΔW_1	0100 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
ΔI_2	0000 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

ΔX_{17}	0000 0000 0000 0000 0000 1000 0000 0000 0000 1000 0000 0000 0000 1000 0000 0000
ΔY_{17}	0000 0000 0000 0000 0000 ???? 0000 0000 0000 ???? 0000 0000 0000 ???? 0000 0000
ΔZ_{17}	0000 0?00 0?00 0?00 0000 0?00 0?00 0?00 0000 0?00 0?00 0?00 0000 0?00 0?00 0?00
ΔW_{17}	0?00 0?00 0?00 0?00 00?0 ???0 0??0 0??0 ???0 00?0 0??0 0??0 0?00 0?00 0?00 0?00
ΔI_{18}	00?0 ?0?? 0??0 0000 0?00 ??0? 0??0 0000 0000 ???? 0??? 0000 0000 ???? 0?00 0000
ΔX_{18}	00?0 ?0?? 0??0 0000 0?00 ??0? 0??0 0000 0000 ???? 0??? 0000 0000 ???? 0?00 0000
ΔY_{18}	???? ???? ???? 0000 ???? ???? ???? 0000 0000 ???? ???? 0000 0000 ???? ???? 0000
ΔO_{18}	???? ???? ???? 0000 ???? ???? ???? 0000 0000 ???? ???? 0000 0000 ???? ???? 0000

Data Collection

- Choose 2^48 stuctures fix in nibbles 1,2,3,4,5,7,8,9,11,12,13,15,16 (2^{23} pairs)
- Verifiy $\Delta C[4, 8, 9, 12, , 13, 16] = 0$ (2^{-1} pairs left)

Key Recovery(1)

- Guess $(k_0 \oplus \mathcal{P}^{-1}(f_1(k_1)))[6]$
- Look for 2^4 pairs st. $\Delta Y_1[6] = 8$
- 2^{-5} pairs left
- same with $(k_0 \oplus \mathcal{P}^{-1}(f_1(k_1)))[10]$ and $(k_0 \oplus \mathcal{P}^{-1}(f_1(k_1)))[14]$
- 2^{-13} pairs left

Key Recovery(2)

- Guess $k_0[i]$, $i \in \{1, 2, 3, 5, 7, 10, 11, 14\}$

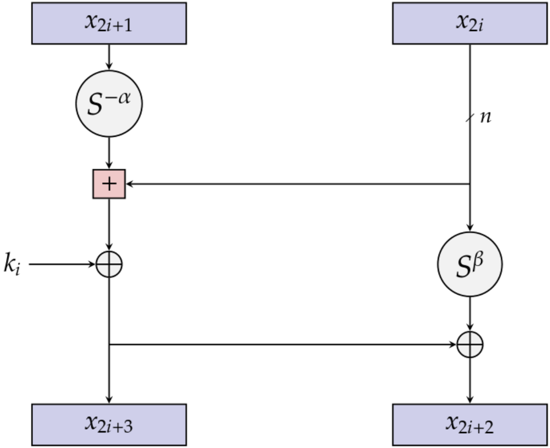
Generalities

- Introduced by Beaulieu & Al (NSA) in 2013
- ARX network
- 48-128 bits blocks
- 96-256 bits key
- 22-34 rounds

Performances (64 bits block/128 bits key)

- 186 bytes of memory
- 150 cycles per block

Structure



Key Scheduling

- $K = (l_{m-2}||l_{m-1}||...||l_0||k_0)$
- $l_{i+m-1} = k_{i-1} + S^{-\alpha}(l_{i-1}) \oplus i$
- $k_i = S^{\beta}(k_{i-1}) \oplus l_{i+m-1}$
- $l_i, k_0 \in \mathbb{F}_2^n$
- $m \in \{2, 3, 4\}$

Principle

- Inject a fault in a chosen state of the computation
- Compare C and C^* the correct and faulty cipher texts
- Retrive information on the key

We control the position of the error (unrealistic)

- $x^T = (S^{-\alpha}(x^{T-1}) + y^{T-1}) \oplus k^{T-1}$
- $y^T = S^{\beta}(y^{T-1}) \oplus x^T$
- $k_j^T = (x_{j+\alpha \mod n}^{T-1} \oplus (y^T + x^T)_{j+\beta \mod n} \oplus c_j) \oplus x_j^T$
- $c_j = (x_{j-1-\alpha \mod n} \& y_{j-1}) | (c_{j-1} \& (x_{j-1-\alpha \mod n} | y_{j-1}))$

- $c_0 = 0$ is known
- Inject a fault in y_0^{T-1}
- Deduce x_α^{T-1} then k_0^{T-1}
- Inject a fault in higher bits of y^{T-1}

We don't control the position of the error

Locate the error

- $e = S^{-\beta}(x^{T^*} \oplus x^{T} \oplus y^{T^*} \oplus y^{T})$