BLIND FAULT ATTACK AGAINST SPN CIPHERS

FDTC 2014

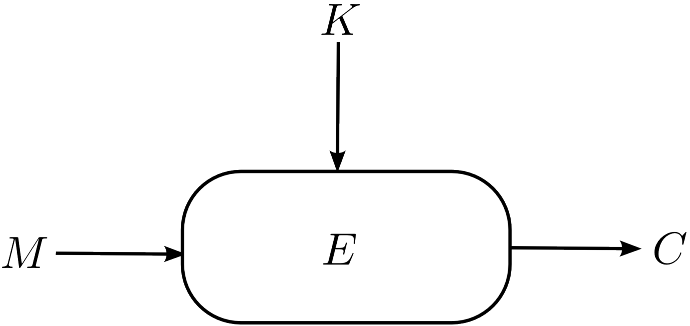
Roman Korkikian, **Sylvain Pelissier**, David Naccache

September 23, 2014

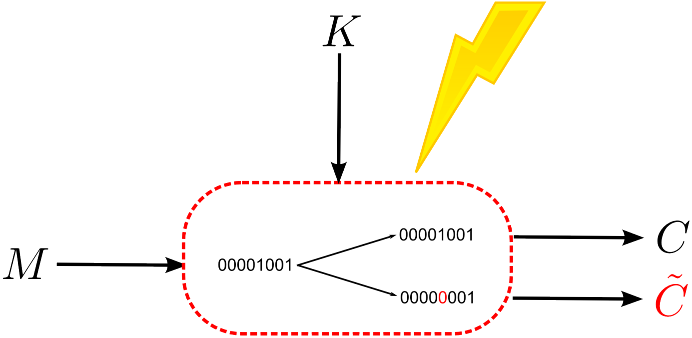
IN BRIEF

- Substitution Permutation Networks (SPN)
- Fault attacks
- Blind fault attack against SPN ciphers
- Results
- Questions

BLOCK CIPHER



FAULT ATTACKS



SUBSTITUTION PERMUTATION NETWORKS

Block cipher construction which uses a sequence of invertible transformations:

- Substitution stage or S-box **S** (Usually 4-bit or 8-bit S-boxes)
- Permutation stage **P**
- Key mixing operation **A**

Structure used by AES, LED, SAFER++, ...

SUBSTITUTION PERMUTATION NETWORKS

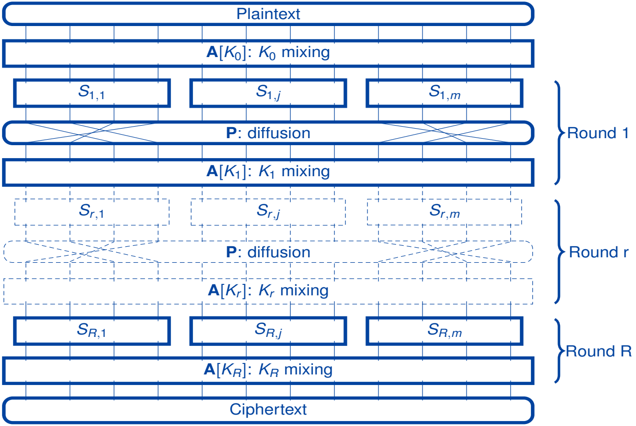


Figure : A typical SPN-based block cipher

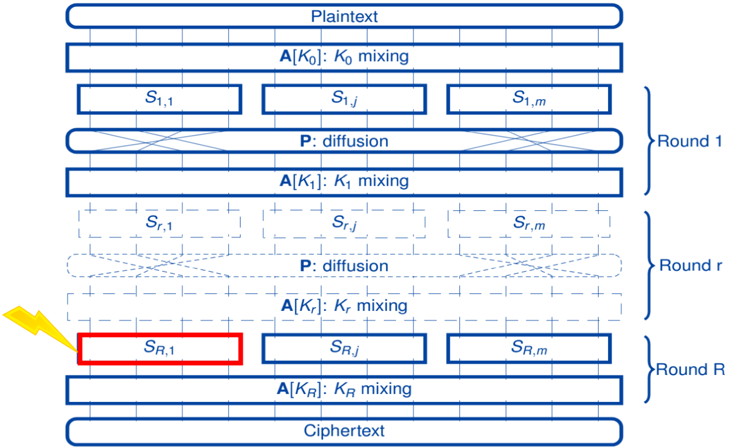
DIFFERENTIAL FAULT ANALYSIS

1. Inject faults in the last rounds of a block cipher
2. Collect pairs $(C_1, \tilde{C}_1), (C_2, \tilde{C}_2), \ldots$
3. Apply a statistical method on these pairs and retrieve the key K.

The input messages do not need to be known:

$$M_1 \to (C_1, \tilde{C}_1),$$
$$M_2 \to (C_2, \tilde{C}_2),$$
$$\ldots$$

DIFFERENTIAL FAULT ANALYSIS



COLLISION FAULT ANALYSIS

The same idea can be applied for inputs:

1. Inject faults in the first rounds
2. Find colliding input pairs $(M_1, \tilde{M}_1), (M_2, \tilde{M}_2), \ldots$
3. Apply a statistical method on these pairs and retrieve the key K.

The output messages do not have to be known:

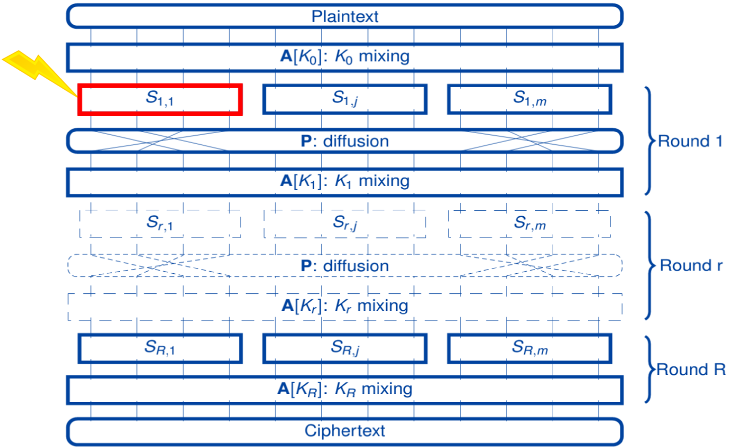
$$(M_1, \tilde{M}_1) \rightarrow C_1$$

$$(M_2, \tilde{M}_2) \rightarrow C_2$$

$$\ldots$$

But they have to be somehow compared (equality check $\mathcal{O}(C_1 = C_2)$).

COLLISION FAULT ANALYSIS



CURRENT ATTACKS FOR AES

Attack	Rounds	Plaintexts	Ciphertexts
DFA	6-10	Unknown	Known
CFA	1-2	Known	Unknown*

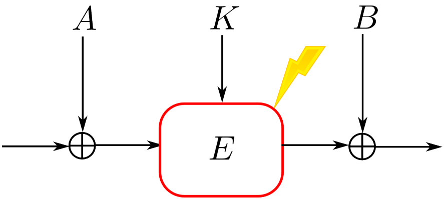
* equality test check ($\mathcal{O}(C_1 = C_2)$)

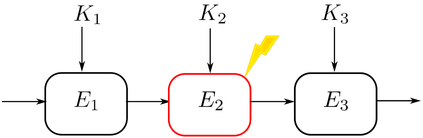
BLIND FAULT ATTACK

What if input **and** output values are not directly accessible ?

EXAMPLES

- Input and output whitening
- Cascade encryption
- Hardware security module





OUR CONTRIBUTION

Attack	Rounds	Plaintexts	Ciphertexts
DFA	6-10	Unknown	Known
CFA	1-2	Known	Unknown*
BFA	Any	Unknown	Unknown*

* equality test check ($\mathcal{O}(C_1 = C_2)$)

ASSUMPTIONS

1. A multi-bit set or reset fault can be injected to an internal byte/nibble X of a SPN block cipher.
2. Unknown plaintexts can be encrypted several times.
3. The different faulted or correct outputs can be compared pairwise without revealing their values (pairwise equality check $\mathcal{O}(C_1 = C_2)$).

BLIND FAULT ATTACK OVERVIEW

1. For each plaintext:
   1.1 Introduce faults during a round execution and compare the different outputs.
   1.2 From the number of different faulted outputs determine the Hamming weights of an algorithm's internal state.
2. For each possible key candidate:
   2.1 Perform a key search to recover a key byte/nibble.

FAULT MODEL

- Multi-bit reset fault

11001001 →
00001000
00000000
11000001
...

- Multi-bit set fault

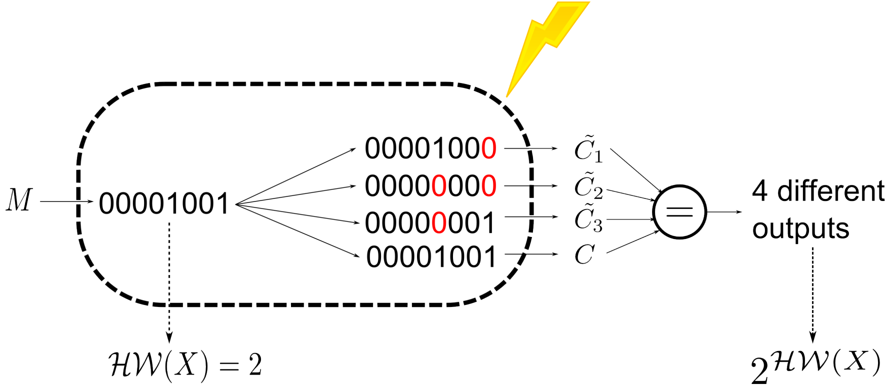
11001001 →
11001101
11011011
11111011
...

MULTI-BIT SET/RESET FAULT MODEL

These fault models have been observed in practice:

- Laser fault injection in SRAM:
  [**Roscian, Sarafianos, Dutertre, Tria**] FDTC 2013
- Electromagnetic glitch fault injections:
  [**Moro, Dehbaoui, Heydemann, Robisson, Encrenaz**] FDTC 2013

HAMMING WEIGHT GUESS



OCCUPANCY PROBLEM

The number of faults injections can be minimized when considered as an "occupancy problem".

- The probability that after ℓ fault injections, Y_ℓ different possible ciphertexts (among $\lambda = 2^{\mathcal{HW}(X)}$) are received can be considered as the probability that Y_ℓ out of λ bins are occupied after throwing randomly ℓ balls.

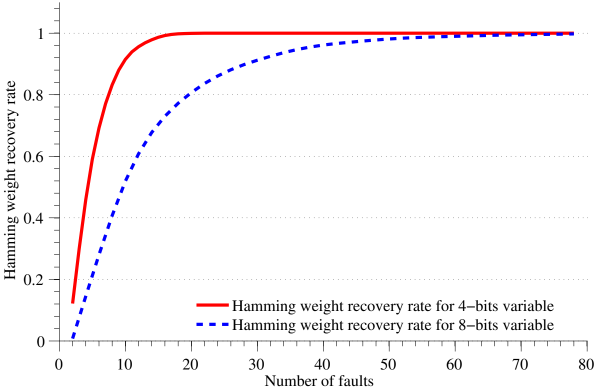
OCCUPANCY PROBLEM

$$\mathbf{Pr}(Y_\ell = \kappa) = \begin{cases} \frac{\lambda! \alpha_{\kappa,\ell}}{(\lambda-\kappa)! \lambda^\ell} & \kappa \in \{1, ..., \min(\lambda, \ell)\} \\ 0 & \text{else} \end{cases}$$

$\hat{\lambda}$ with maximum likelihood is assumed as correct:

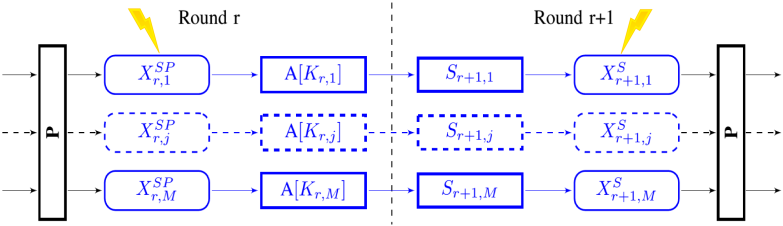
$$\hat{\lambda} = \arg\max_{\lambda_i} \mathbf{Pr}(Y_\ell = \kappa | \lambda_i)$$

SIMULATION

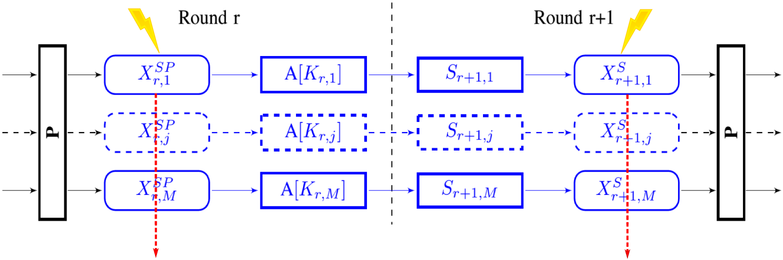


- 15 faults give a 99% success probability for a 4-bit variable.
- 62 faults give a 99% success probability for a 8-bit variable.

TARGETED STATES



TARGETED STATES



$$h_r = \mathcal{HW}\left(X^{SP}_{r,1}\right)$$

$$h_{r+1} = \mathcal{HW}\left(\mathbf{S}_{r+1,1} \circ \mathbf{A} \mid_K \left(X^{SP}_{r,1}\right)\right)$$

KEY SIFTING

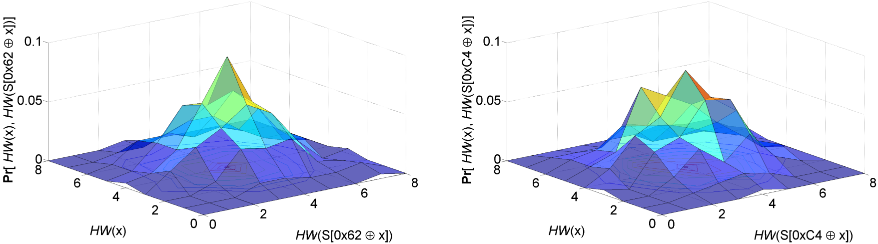
- For each key byte/nibble candidate K_i:
  - if $\nexists X \ \mathcal{HW}(X) = h_r$ and $\mathcal{HW}\left(\mathbf{S}_{r+1,j} \circ \mathbf{A}\,|_{K_i}(X)\right) = h_{r+1}$: K_i is discarded from the candidate list.

$\Rightarrow$ A lot of Hamming weight pairs needed to reduce the candidate list

$\Rightarrow$ Can be improved with key likelihood estimation.

KEY LIKELIHOOD ESTIMATION

It was determined that Hamming weight distribution of key mixing and S-box is unique for the tested ciphers:



KUDELSKI SECURITY

KEY LIKELIHOOD ESTIMATION

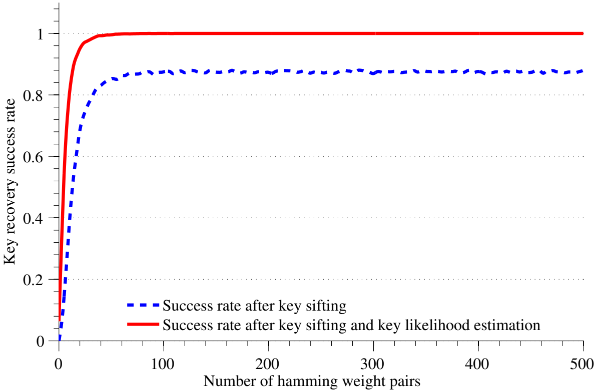
1. Before the attack, the Hamming weight distributions are precomputed for each key candidate.
2. The Euclidean distance between the distribution of the recovered Hamming weight pairs and the precomputed distributions is computed for all remaining key candidates.
3. The key candidate with the minimal distance is assumed to be correct.

SIMULATIONS

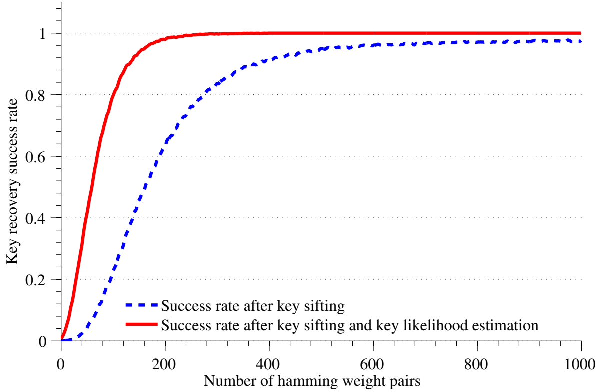
Table : Specification of operation for different ciphers

Cipher	Exact operation	Size
LED	$X^{S}_{r+1,j} = \mathbf{S}\left[K_{r,j} \oplus X^{SP}_{r,j}\right]$	4-bit
AES	$X^{S}_{r+1,j} = \mathbf{S}\left[K_{r,j} \oplus X^{SP}_{r,j}\right]$	8-bit
SAFER++	$X^{S}_{r+1,j} = \mathbf{S}\left[K_{r,j} + X^{SP}_{r,j}\right]$	8-bit

LED SIMULATION



AES SIMULATION



RESULTS

Number of faults used to recover a key byte/nibble:

Cipher	# plaintexts	# faults per plaintext	Total # faults
LED	50	40	2,000
AES	250	120	30,000
SAFER++	200	120	24,000

RESULTS

- Fault attacks are feasible even when input and output messages are not known but ciphertext equality check is available.
- Fault attacks can be applied against any SPN round.
- Fault model is generic and has been observed in practice.
- The total number of faults to recover a key is the price to pay for blindness (480,000 for a complete AES key).

FUTURE DEVELOPEMENTS

- New methods to reduce the number of required fault injections.
- Hamming weight distribution theory.
- Results and problems when applied in practice.

QUESTIONS

KUDELSKI SECURITY

THANK YOU!

www.kudelskisecurity.com
cyber security unit of Kudelski Group

HAMMING WEIGHT PROBABILITY DISTRIBUTION

$$\mathbf{Pr}_k \left[\mathcal{HW}(x), \mathcal{HW}\left(\mathbf{S}_{r+1,j} \circ \mathbf{A} \mid_k (x)\right)\right]$$