A Fault Attack against the FOX Cipher Family

L. Breveglieri[1], I. Koren[2], P. Maistri[3]

[1]Politecnico di Milano, Milano, ITALY

[2]University of Massachusetts, Amherst, MA, USA

[3]TIMA Laboratory, Grenoble, France

Outline

- Introduction
- Symmetric ciphers
- The FOX cipher
- DFA principles
- DFA against FOX
- Results and discussion
- Conclusions

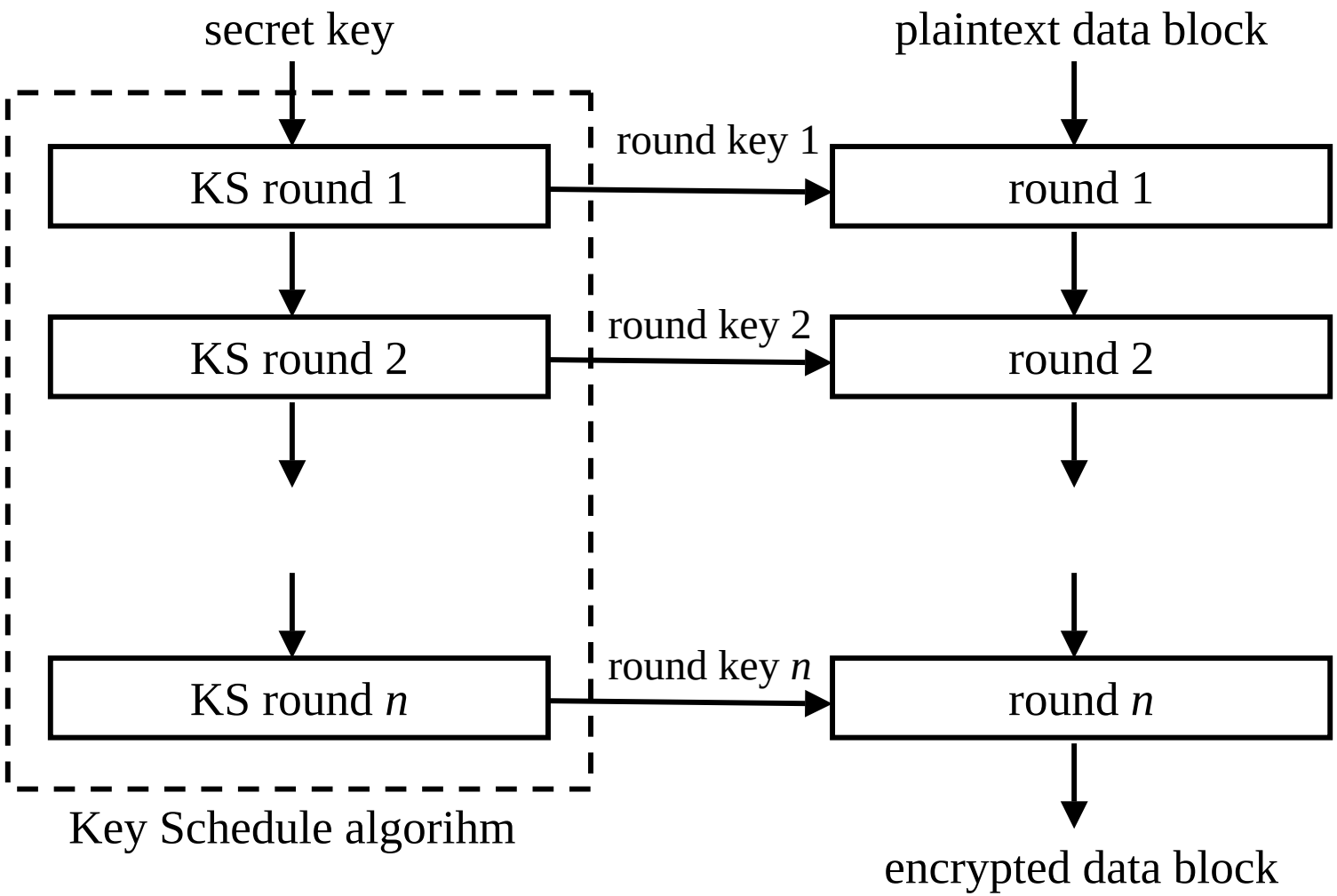
Introduction

- Security analysis of cryptosystems must be considered from specification to implementation.
- Implementations may leak side-channel information:
  - **execution time**
  - **power consumed**
  - **response to fault injections**
- Differential Fault Analysis is very effective and has been successfully applied to DES, AES, RSA, ECC, XTR.

Symmetric Ciphers (1)

- Single key for both encryption and decryption.
- Usually of the block type, for instance:
  - input data (64 bits), key (64 bits) → output data (64 bits)
- Most symmetric ciphers are iterative,
  and structured in rounds.
- Each round precesses the input block
  and a specific round key.
- The number of rounds (and hence of round keys)
  may range from 10 to 100.
- Each round key is derived from the (master) secret key
  by means of an additional algorithm, called key scheduling.

Symmetric Ciphers (2)



The FOX Cipher (1)

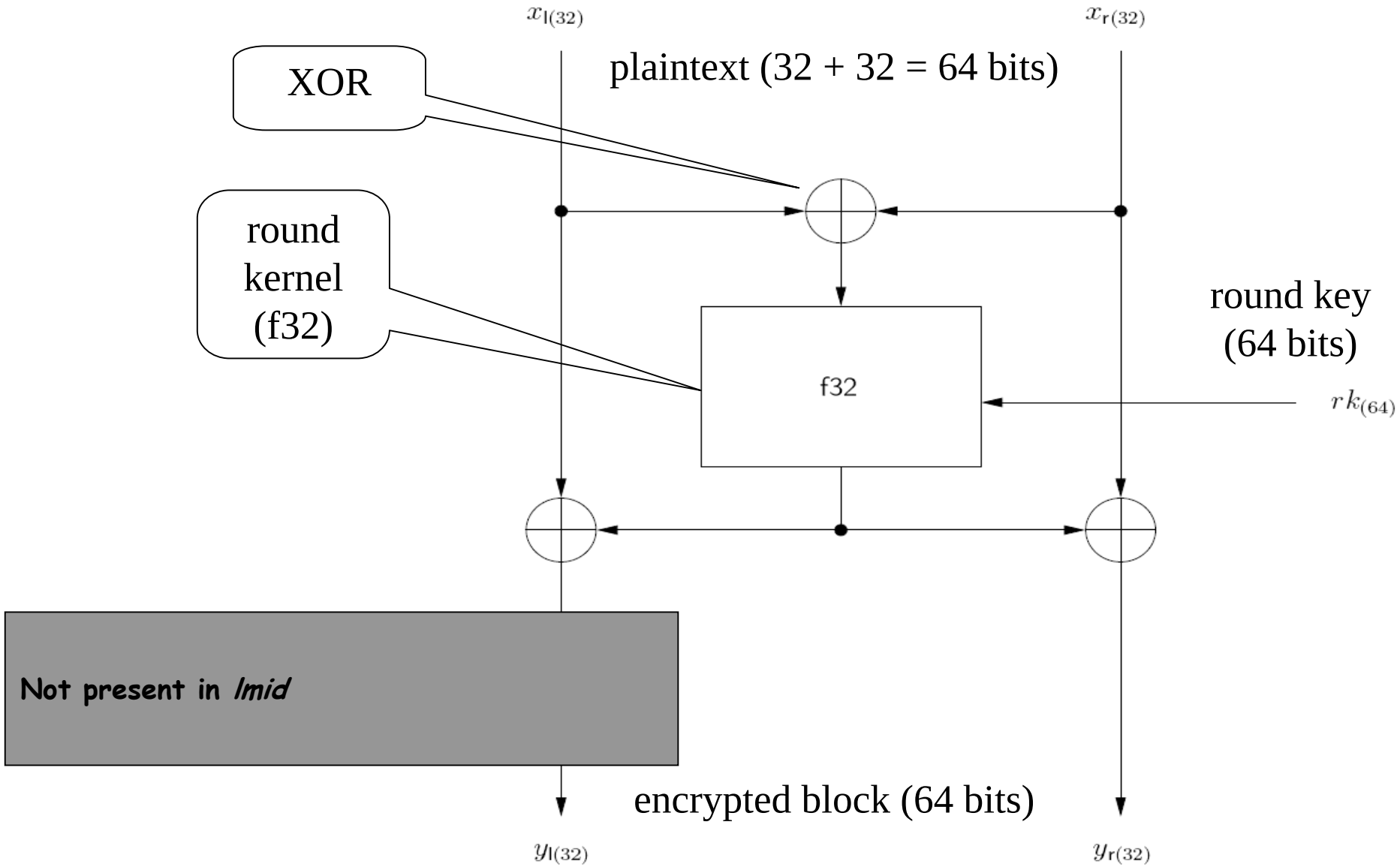
- Is a recent family of symmetric ciphers (2004).
- Conceived as a successor to the cipher IDEA, that was based on a (somewhat complex and twisted) mix of logical, finite field and integer operations.
- FOX is more uniform than IDEA, as it works completely in finite fields (more AES-oriented).
- FOX is proposed mainly for multimedia streaming and data storage applications.

The FOX Cipher (2)

Name	**Inp./Outp. Size**	**Key Size**	**# Rounds**
FOX 64	64 bits	up to 128 bits	16
FOX 128	128 bits	up to 256 bits	16

- The FOX round consists of the *lmor* function, the last round is slightly different (*lmid* function).
- Each round contains two non-linear layers and three (round)key-mixing phases.

The complete round: *lmor*



The round kernel: *f32*

Addition with left round key

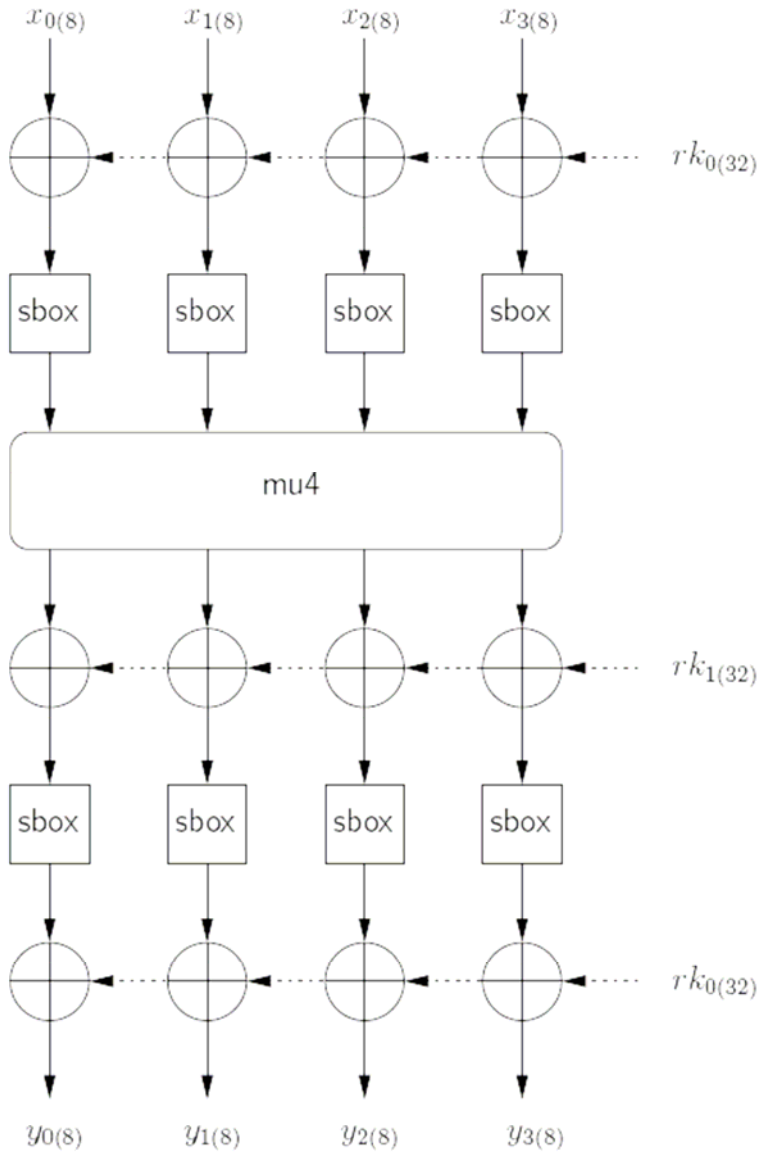
Non-linear subst. layer (Sbox)

Linear diffusion layer (addition and scaling)

Addition with right round key

Non-linear subst. layer (Sbox)

Addition with left round key



Operations

- Underlying finite field GF(2^8), with fixed generating polynomial (not the same as AES).
- Non-linear substitution layer:
  - a series of Sboxes – input 8 bit, output 8 bit
  - does not have a simple algebraic description
- Linear diffusion layer:
  - a series of additions and scalings in GF(2^8)
  - has an algebraic description
- All is rooted in the finite field GF(2^8).

Differential Fault Analysis (DFA) Principles

- DFA exploits the responses to error injection.
- It allows to apply linear and differential analysis to individual steps of the cipher (e.g., the last single round).
- Significant results are obtained by injecting the error before the last non-linear layer:
  - injection may occur earlier: analysis is more complex, but fewer attempts may be required
- Most round-based symmetric ciphers can be attacked in this way (e.g. AES has been).

General DFA Attack Procedure

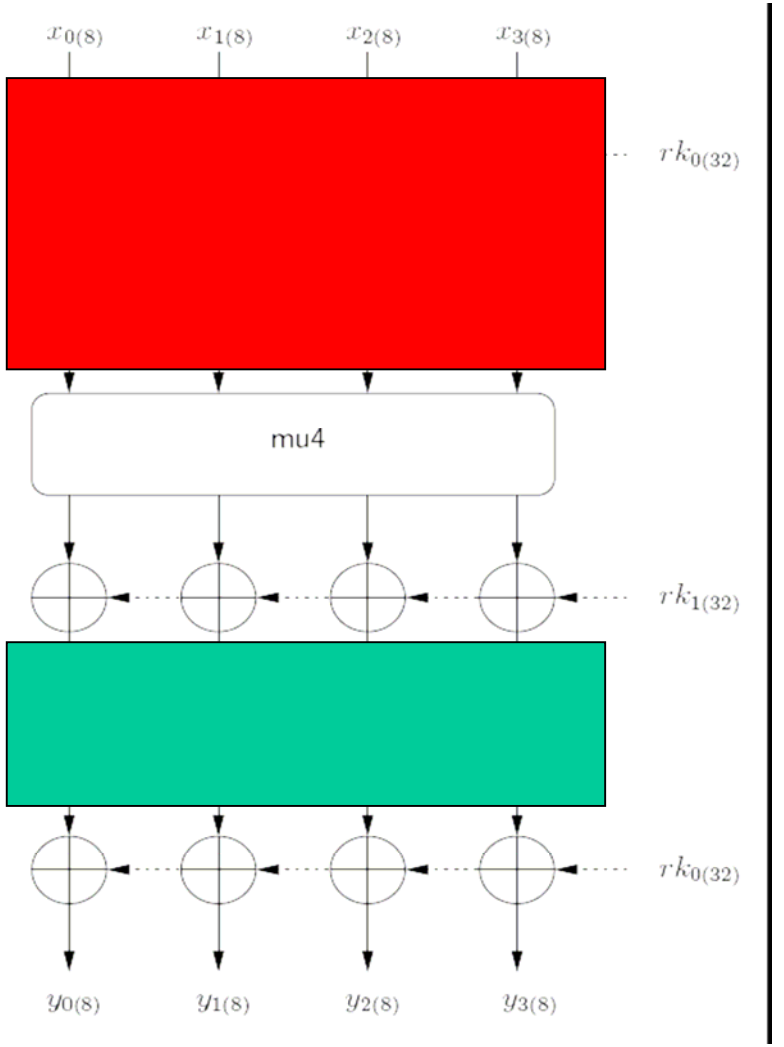
- Inject faults before the last round and collect some differential pairs.
- Reduce progressively and finally guess the values processed in the last round.
- Recover the last round key from these values.
- Invert the key schedule, if possible, and retrieve the master secret key.
- Otherwise, try to attack the last second round, etc.

DFA against FOX (1)

- Attack conducted to the last round in order to retrieve the last round key.
- Assume to be able to inject one error byte between two layers.
- The error is modeled as an additive element in the field GF(2^8) (i.e. one byte XORed to the correct value).
- Note: KS of FOX is not invertible.

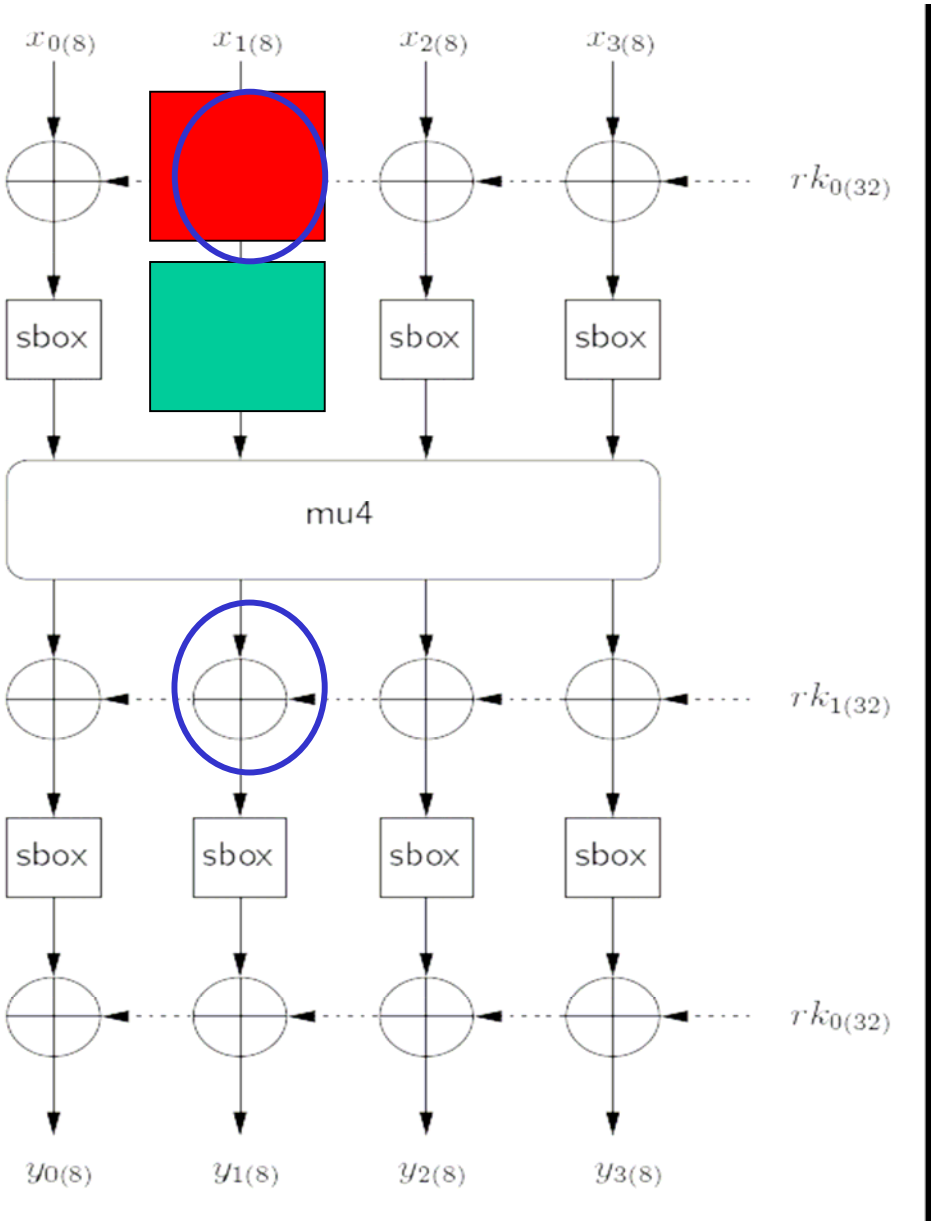
DFA against FOX (2)

- The attack requires a 2-phase setup.
- First phase:
  - inject the fault before the linear layer
  - one byte error only (located with precision)
  - analyze the error propagation in the second sbox layer
  - cross-compare different experiments
  - find the input to 2nd sbox layer



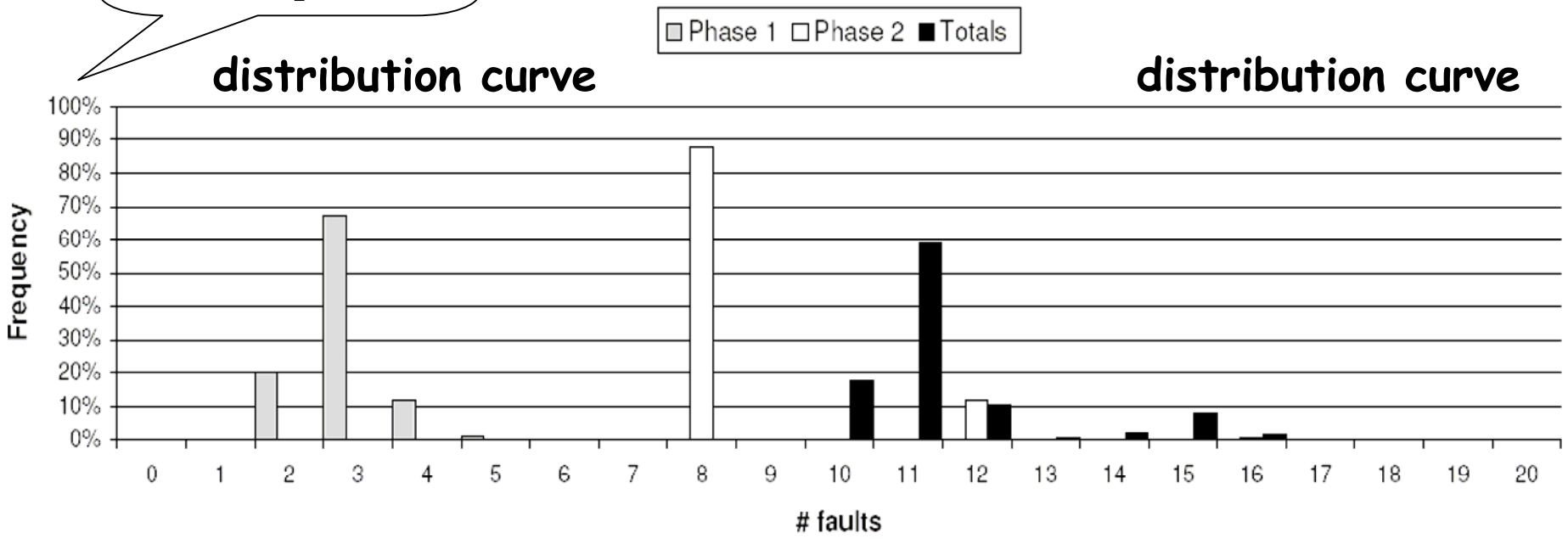
DFA against FOX (3)

- Second phase:
  - inject the fault before the 1*st* sbox layer
  - one byte at a time
  - analyze the error propagation in the 1st sbox layer
  - cross-compare different experiments
  - find the input to the 1st sbox layer
  - **compute the right round key**
  - **compute the left round key**
  - repeat for each byte



Results

% over of whole test space



Minimum, average and maximum number of faults required to recover the entire round key

	Min	**Avg**	*Max*
Phase 1	*2*	**2.94**	*8*
Phase 2	*8*	**8.51**	*28*
Total	*8*	**11.45**	*31*

Discussion (1)

- The last round key can be found with an average of 11.45 faulty encryptions.
- The process can be iterated on each single round, to obtain all the round keys, one at a time.
- Attacking the key schedule directly is not so obvious:
  - the key schedule is not invertible (opposite of AES)
  - however, different types of attacks may perhaps be used together with fault injection to get the secret master key

Discussion (2)

- Presently the procedure relies on the exact location of the error injected in phase 1
- This constraint might be relaxed, by making a guess on the location.
- In most cases only 2 or 3 faults suffice, if the location is known.
- Upper bound of 4^f attemps (f number of possible errors), if the location is not known.
- Actual experiments not carried out.

Conclusions

- Fault attacks to FOX are still a very effective weapon for attackers.
- The round key of FOX cipher can be obtained after 11.45 (on average) faulty encryptions per round.
- The non-invertible key schedule does not seem to help much.

Future

- Improve the attack to FOX and relax the error location constraints or move to the second last round, etc.
- Explore protection methods:
  - e.g. codes, as alredy done for AES
  - uniform cipher, much more than IDEA
  - more suited to cheap code implementation