FIMA: Fault Intensity Map Analysis

Keyvan Ramezanpour, Paul Ampadu, William Diehl

Bradley Department of Electrical and Computer Engineering
Virginia Polytechnic Institute and State University
Blacksburg, USA

4 April 2019

Supported by NIST award 70NANB18H219 for Lightweight Cryptography in Hardware and Embedded Systems.

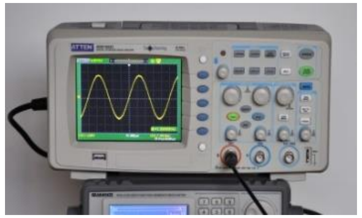
Outline

- Introduction
- Background
- FIMA: Fault Intensity Map Analysis
- Attack on Ascon
- Results
- Conclusions and Future Directions

Introduction

Side-channel Attacks

- Effective information security requires cryptography
- Cryptographic Algorithms mathematically sound
  - ➢Cryptanalysis not much easier than brute-force attacks
- However, cryptography conducted in the physical world
  - ➢Hardware and software
  - ➢Recover secret key
- Side Channel Attack techniques
  - ➢Passive (Power, EM, Timing)
  - ➢Active (Fault injections)

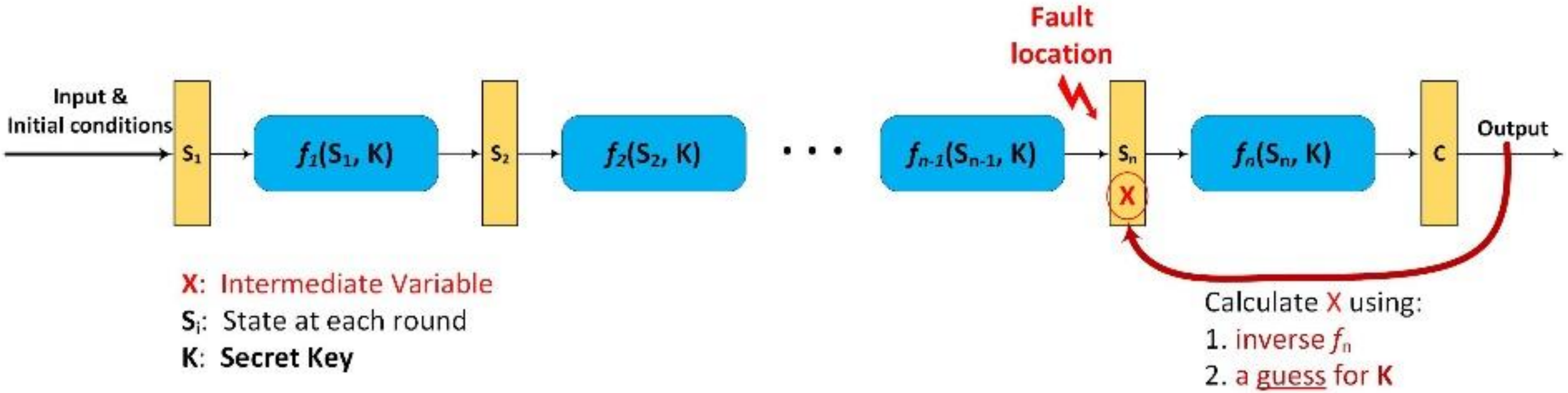


Our research

- Look to improve previous statistical fault analysis attacks
- Introduce FIMA: Fault Intensity Map Analysis
  - Builds on statistical fault attacks
  - Combines observations of fault *bias* and *intensity*
  - Reduces total number of fault experiments to recover secret key
- Demonstrate on Ascon authenticated cipher

Background

Fault Analysis



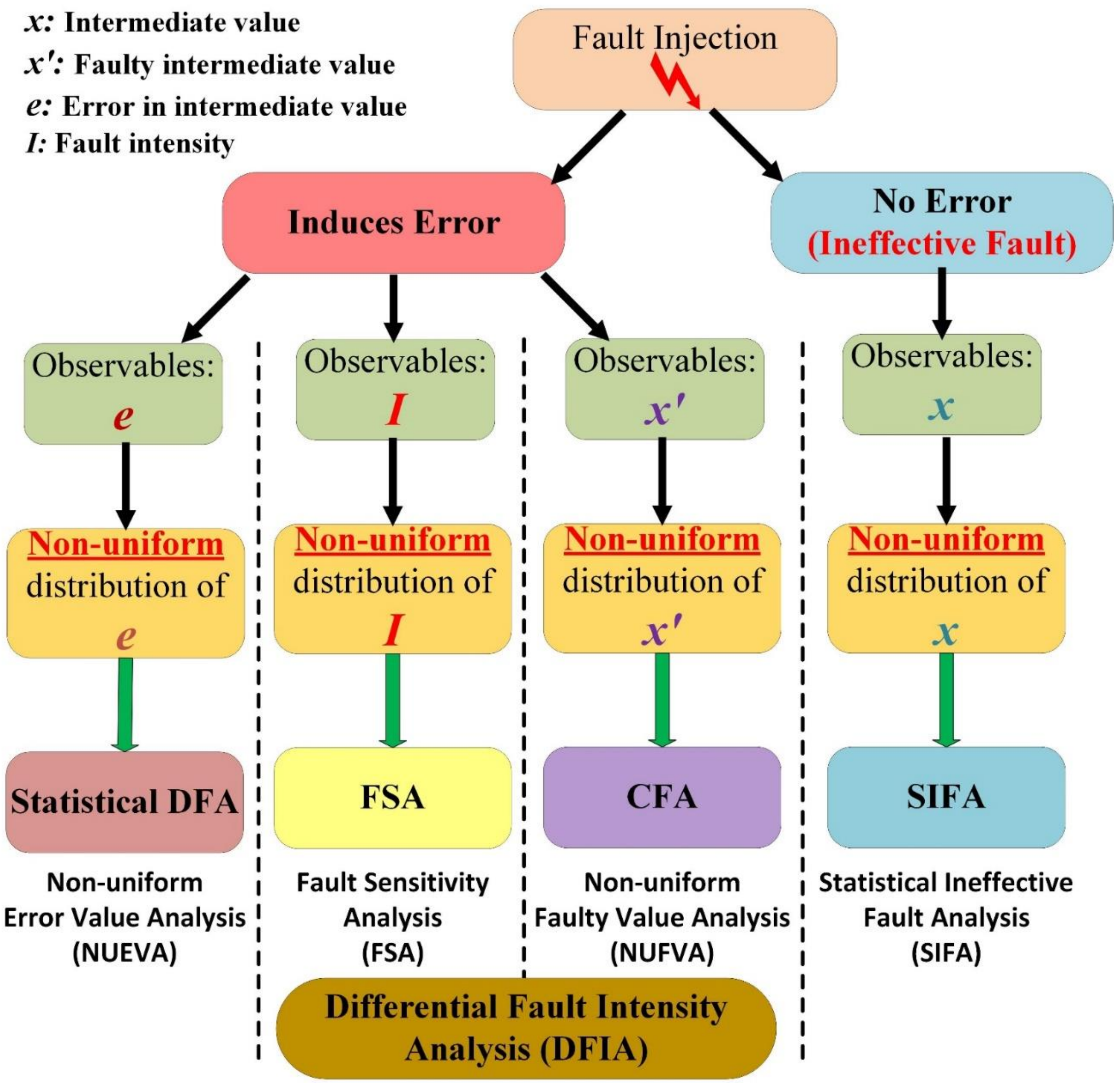
Statistical Fault Analysis

Differential Fault Analysis
- Collect ciphertexts with ***and*** without faults

Advantages	Disadvantages
Fewer experiments	Known state
	Precise faults

Statistical Fault Analysis
- Collect ciphertexts with ***or*** without faults
- Look for properties of output data

Advantages	Disadvantages
Relaxed assumptions on state and fault model	More experiments

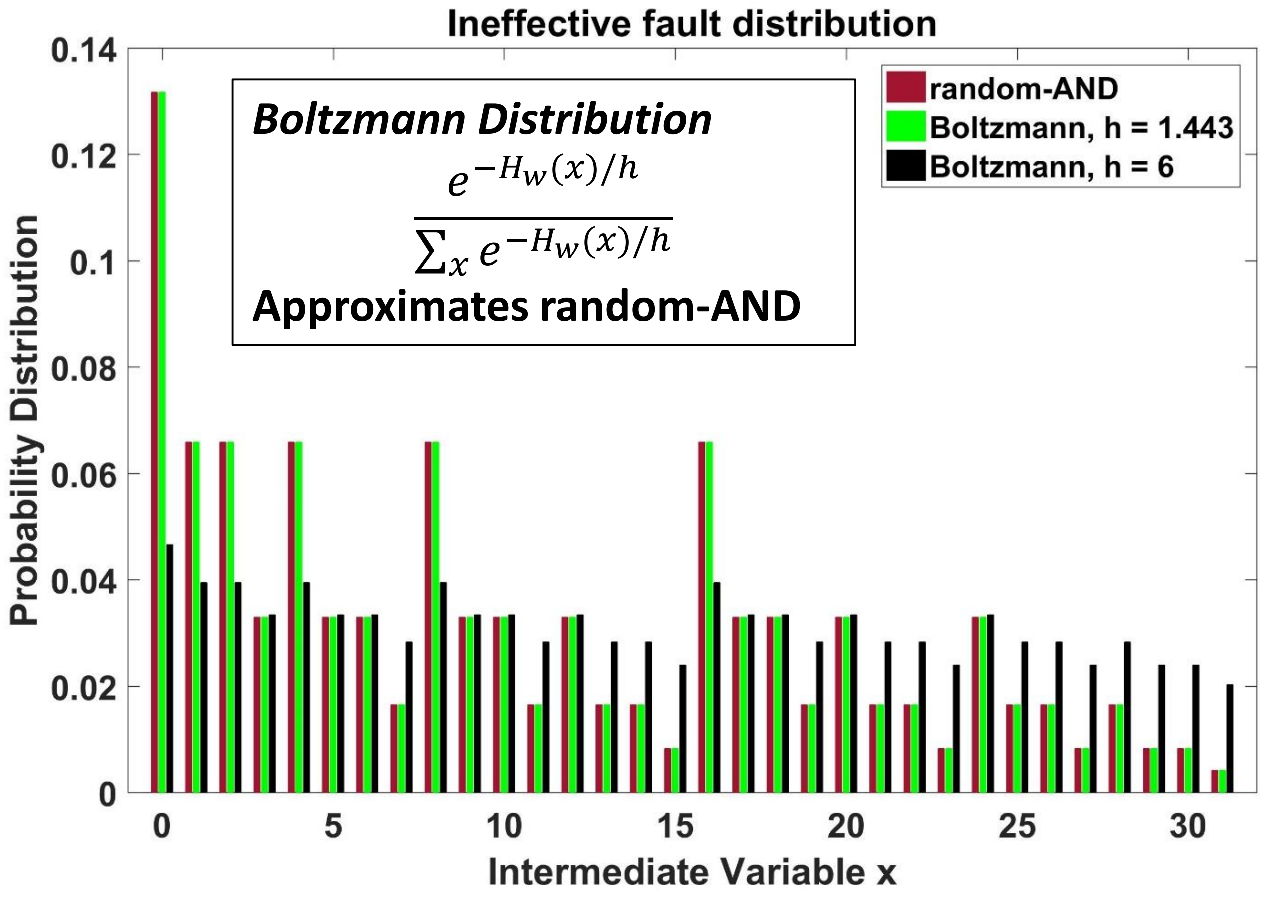


Data-dependent error results in *fault bias*

- Random AND model

  $\boldsymbol{x}' = \boldsymbol{x} \odot \boldsymbol{e}$

- if p_x, $p_e = 0.5$
  - ➢ $\boldsymbol{x}'$ biased to low Hamming weight
  - ➢ Values of $\boldsymbol{x}$ with **low**/**high** HW experience fault **less**/**more** likely;
  - ➢ Correct values biased toward low HW



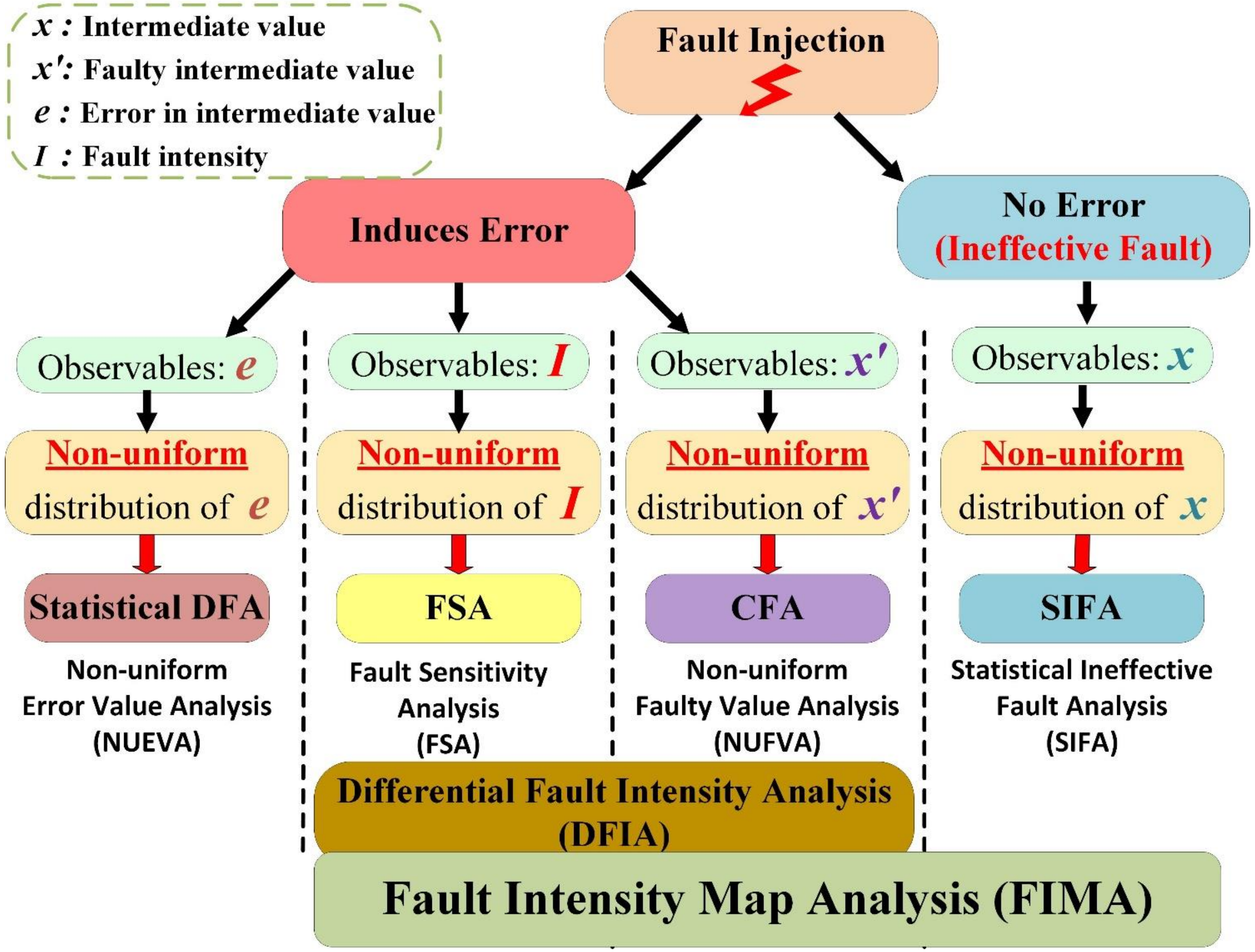
- Suggests analysis of bias to recover $\boldsymbol{x}$
- Foundation of SIFA (Dobraunig et al, 2018)

FIMA: Fault Intensity Map Analysis

FIMA: Fault Intensity Map Analysis

<u>FIMA exploits</u>

- Fault bias
- *Intensity disposition*
  - Correlation between fault distribution and intensity



Correlation of fault distribution with intensity

- Fault intensity is the rate, propensity, or strength at which faults are applied
  - Different meanings for different physical faults (e.g., clock, voltage, optical, etc.)
- With very low intensity, distribution of intermediate variables close to uniform
- With high intensity, probabilities of intermediate variables change
  - some ↑, some ↓
- *Probability distribution correlated with fault intensity*
  - Used in Fault Sensitivity Analysis (FSA) (Li et al, 2010), Differential Fault Intensity Analysis (DFIA) (Ghalaty et al, 2014)
  - *Stable* intensity profile - *correct* key guess
  - *Changing* intensity profile - *incorrect* key guess

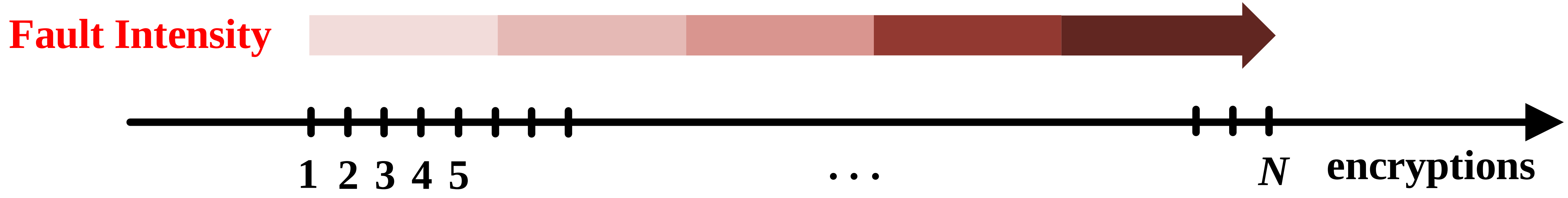
Random AND model modified for intensity

- random-AND model modified to include the effect of *intensity*:

$$x' = \begin{cases} x, & \text{with probability} \quad 1-p \\ x \odot e, & \text{with probability} \quad p \end{cases}$$

  - Where $\boldsymbol{p}$ is a measure of fault intensity;
- With probability $\boldsymbol{p}$:
  - Data distribution follows the random-AND model.
- With probability $\mathbf{1} - \boldsymbol{p}$ fault intensity not enough to induce any error;
  - Differs from ineffective fault induction!
  - Under ineffective fault induction, intensity is enough to induce errors for *some* data.

Fault Intensity Map Analysis

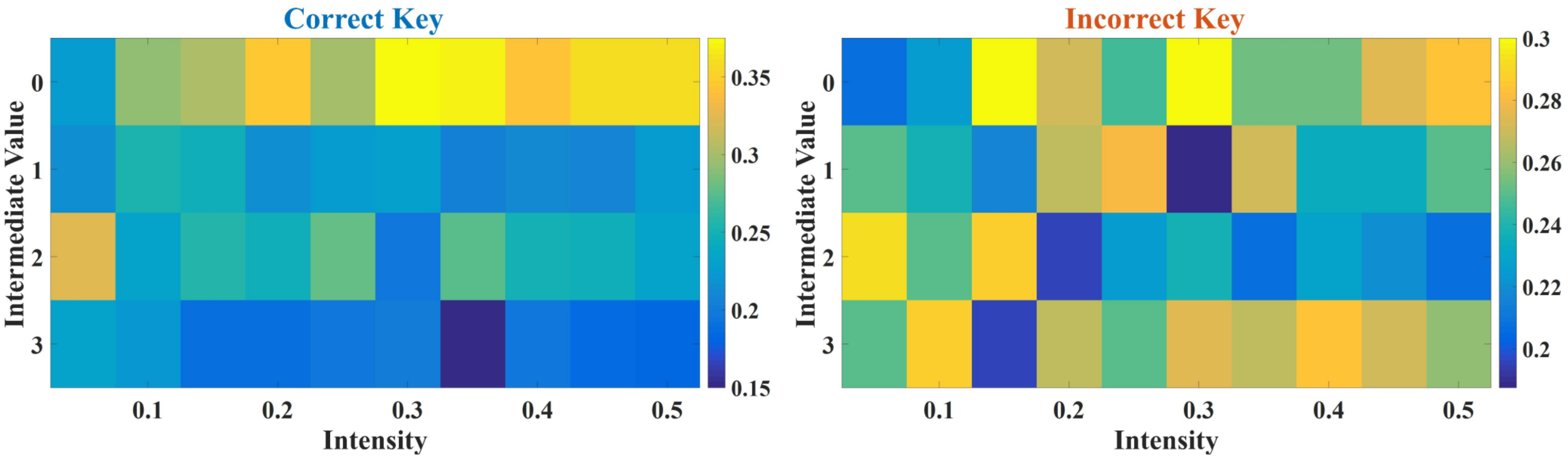


- Collect ciphertexts under fault injection with varying intensities
- Calculate the intermediate variable with a key candidate

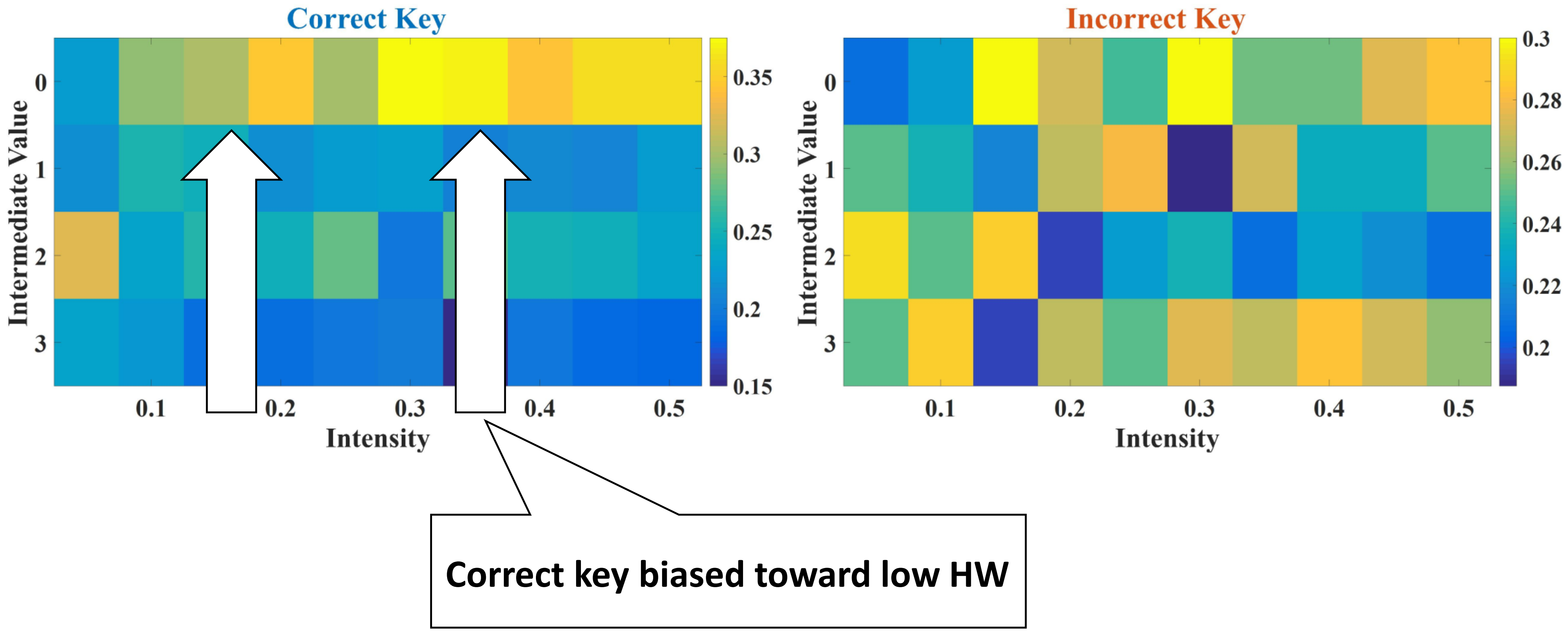
- Define *fault images*:
  - 2D map of the distribution of intermediate values at every fault intensity
  - One for each target subkey

- Distinct features of the fault image reveals the correct key.

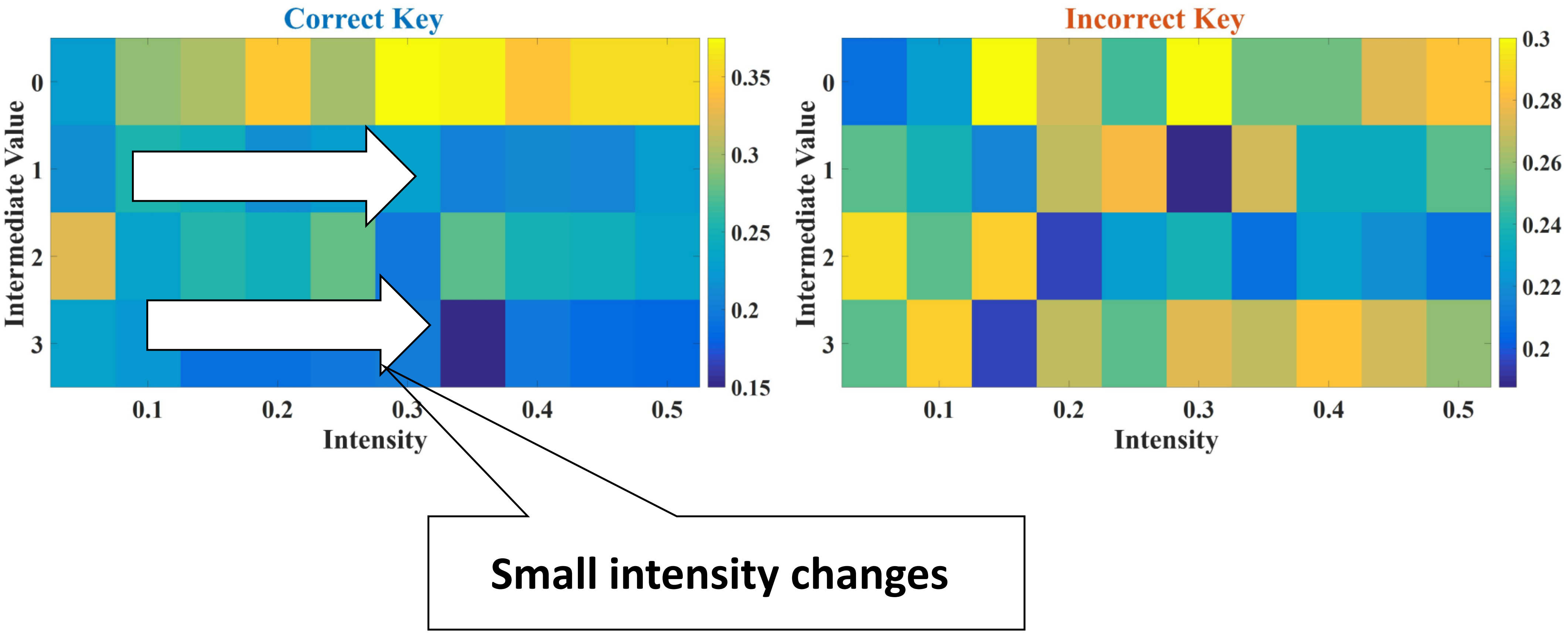
Fault Images



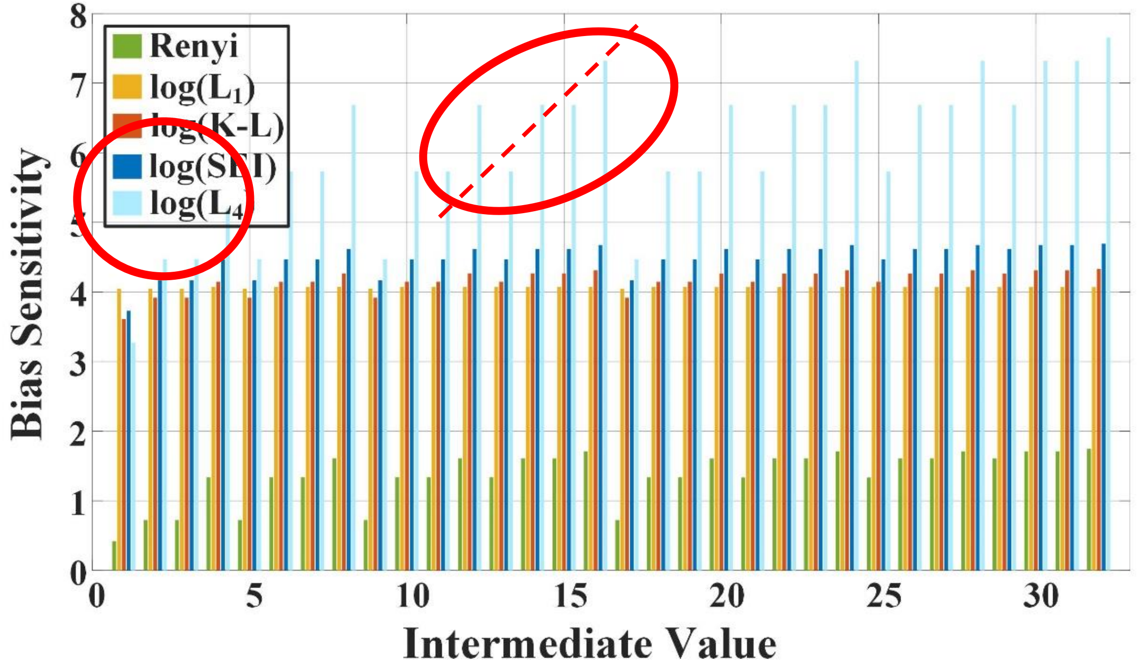
Fault Images

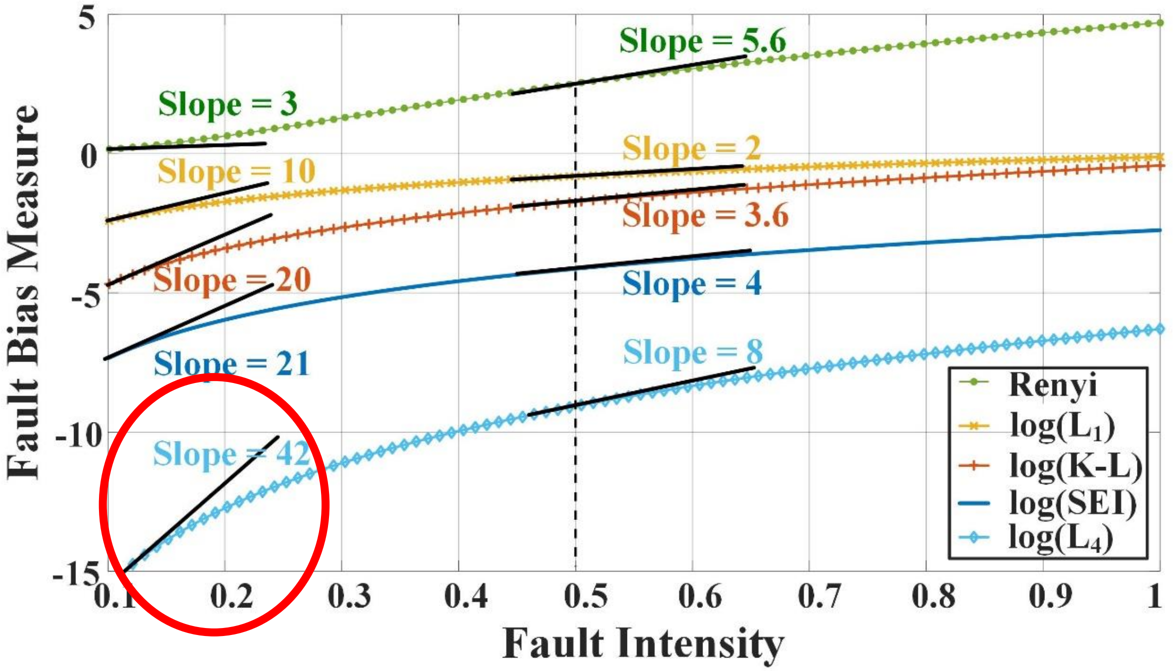


Fault Images



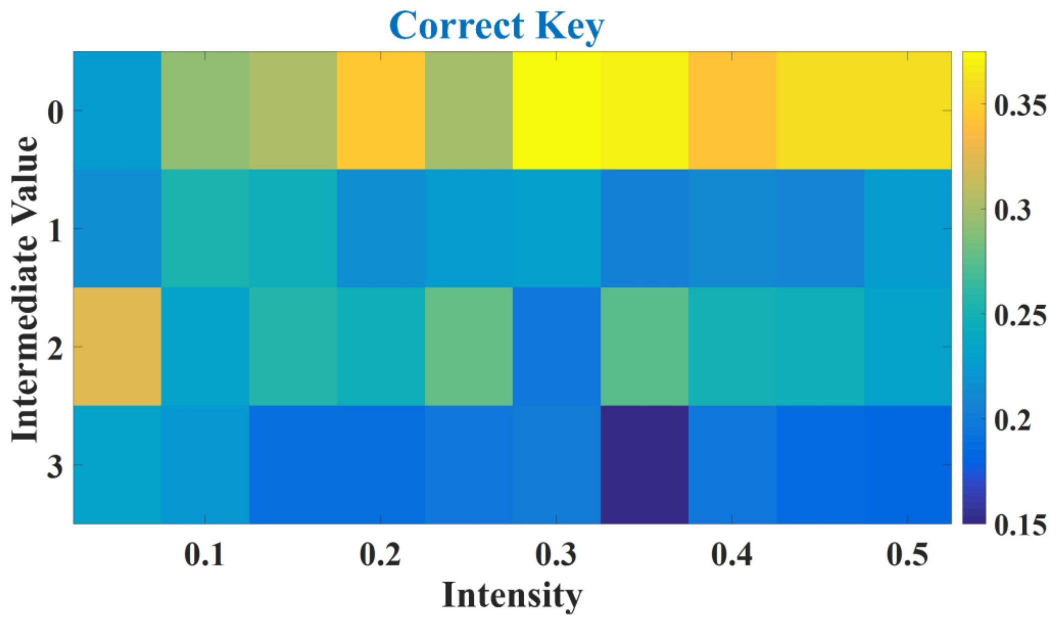
Choice of distance metric

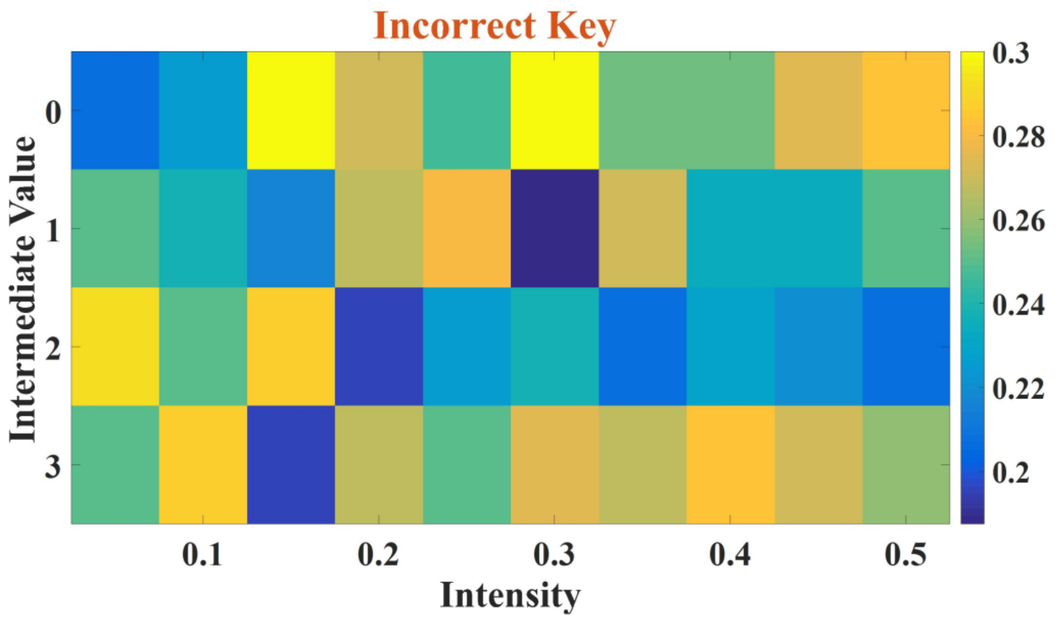




L_4-norm has highest sensitivity

Fault Images





Score value for data bias

$$\mathcal{D}_b(p_X;K) = \log\Big(\sum_i (p_i - q_i)^4\Big)$$

Score value intensity disposition

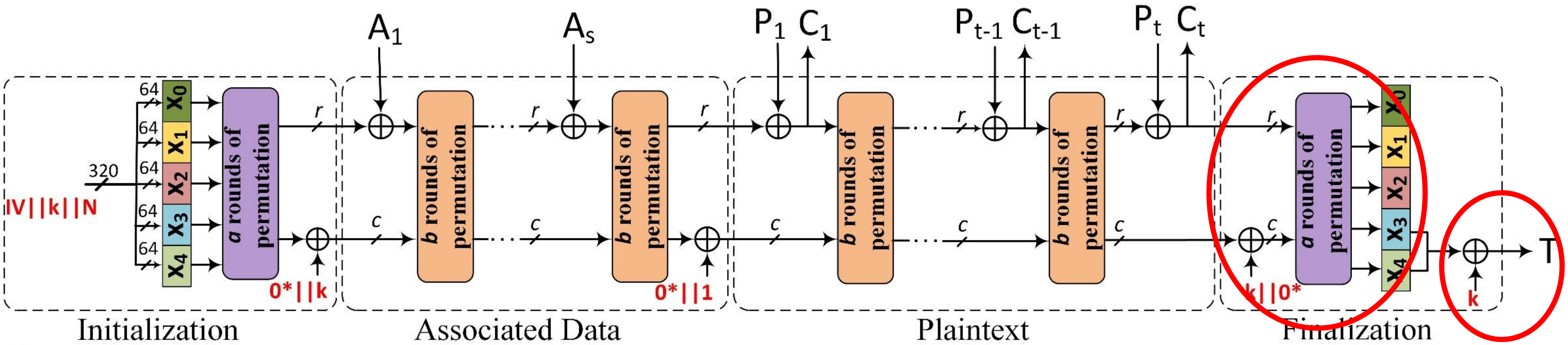
$$D_I(p_X;K) = -\log\left(\frac{1}{R-1}\sum_{i=1}^{R-1}\mathcal{D}_4\Big(p_X(x,I_{i+1};K)||p_X(x,I_i;K)\Big)\right)$$

Total score is weighted sum of two components

$$\mathcal{R}(K,\gamma) = (1-\gamma)\cdot\mathcal{D}_b(p_X;K) + \gamma\cdot D_I(p_X;K)$$

Attack on Ascon

Ascon Authenticated Cipher



k: 128-bit secret key

IV: Initial Vector **N**: Nonce

$\{A_1, \ldots, A_s\}$: Blocks of associated data

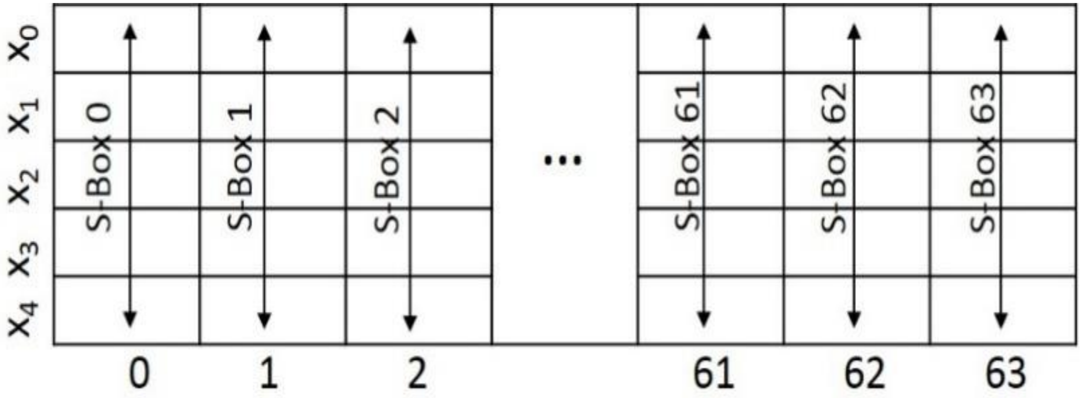
$\{P_1, \ldots, P_t\}$: Blocks of plaintext

$\{C_1, \ldots, C_t\}$: Blocks of ciphertext

T: 128-bit tag

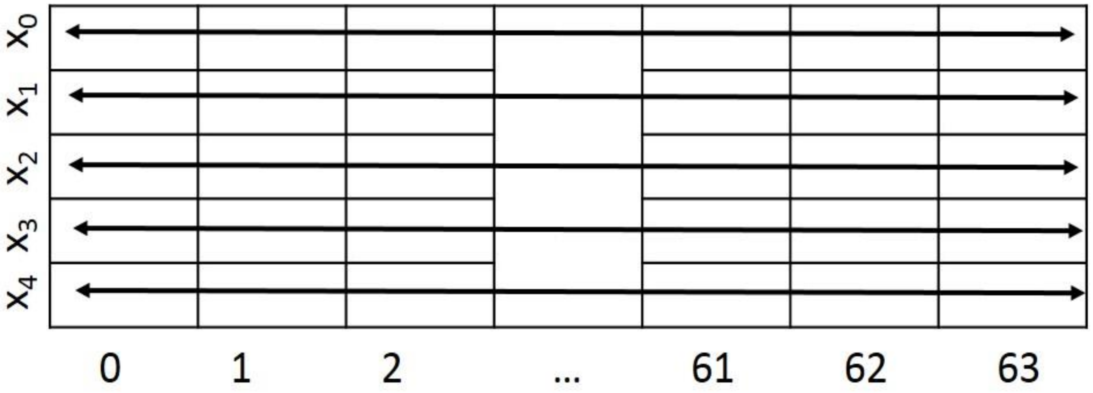
Ascon permutation

64 x 5-bit S-boxes (bits 0, 1, … 4)



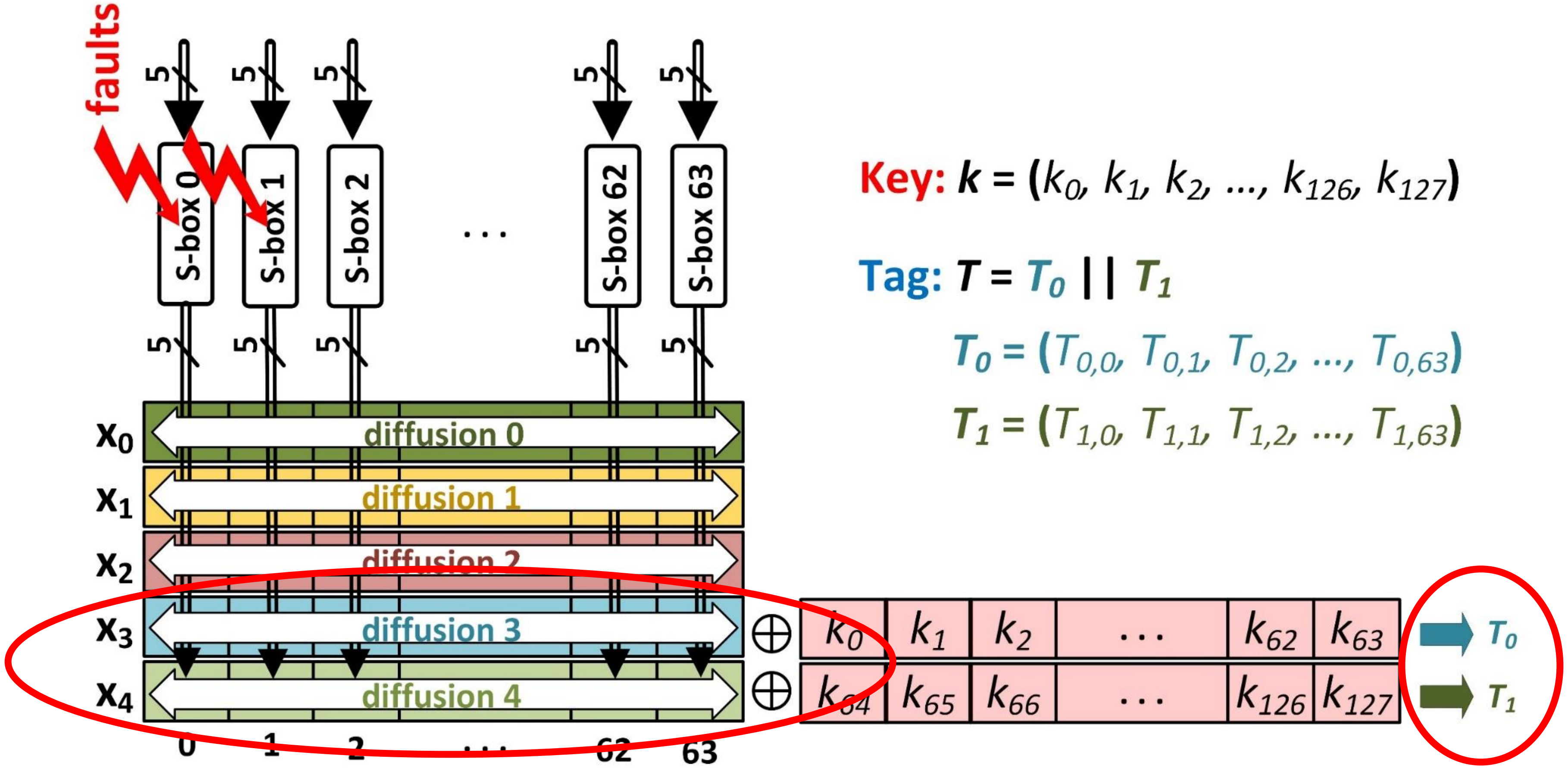
Substitution Layer

5 x 64-bit diffusions (bits 0, 1, … 63)



Diffusion Layer

Double Fault Injections



Intermediate variable x’ at (S-box j, S-box j+1) at last round of finalization

Tag collected in FIMA (not ciphertext)

Attack Algorithm

While $P_{Fail} > \varepsilon$:

For increasing intensities *I*:

1. For next message, inject fault with intensity *I*; collect tags.
2. For all target subkeys in search space (0 … 2^n-1):
   a. Calculate intermediate variable;
   b. Update fault image.
3. Adjust score weighting parameter γ.
4. Pick the correct key guess.

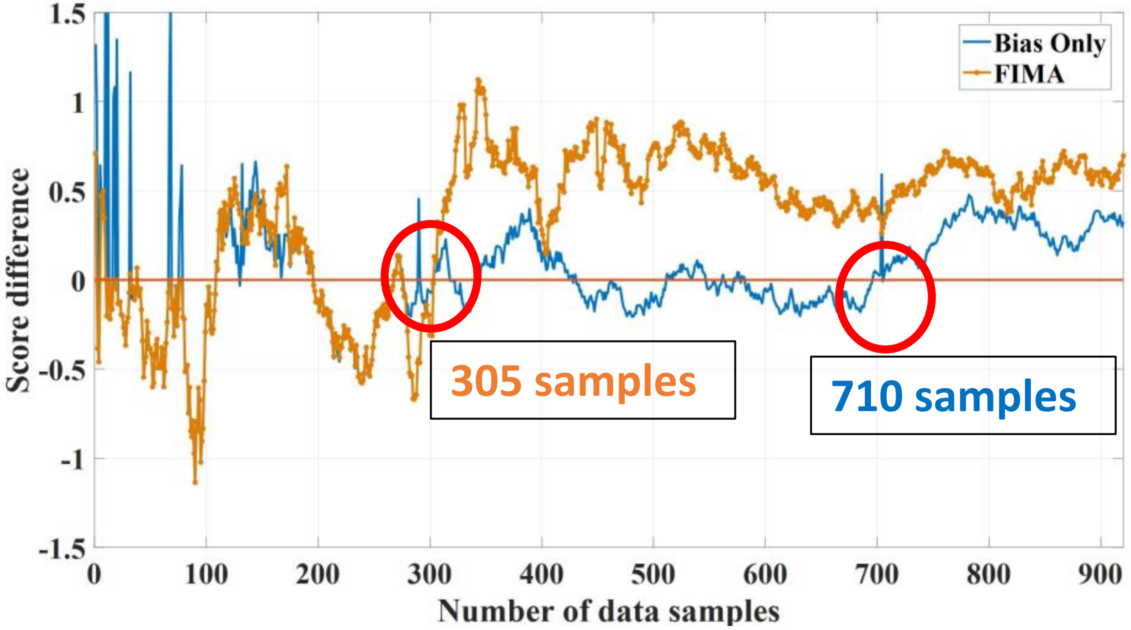
Update P_{Fail};

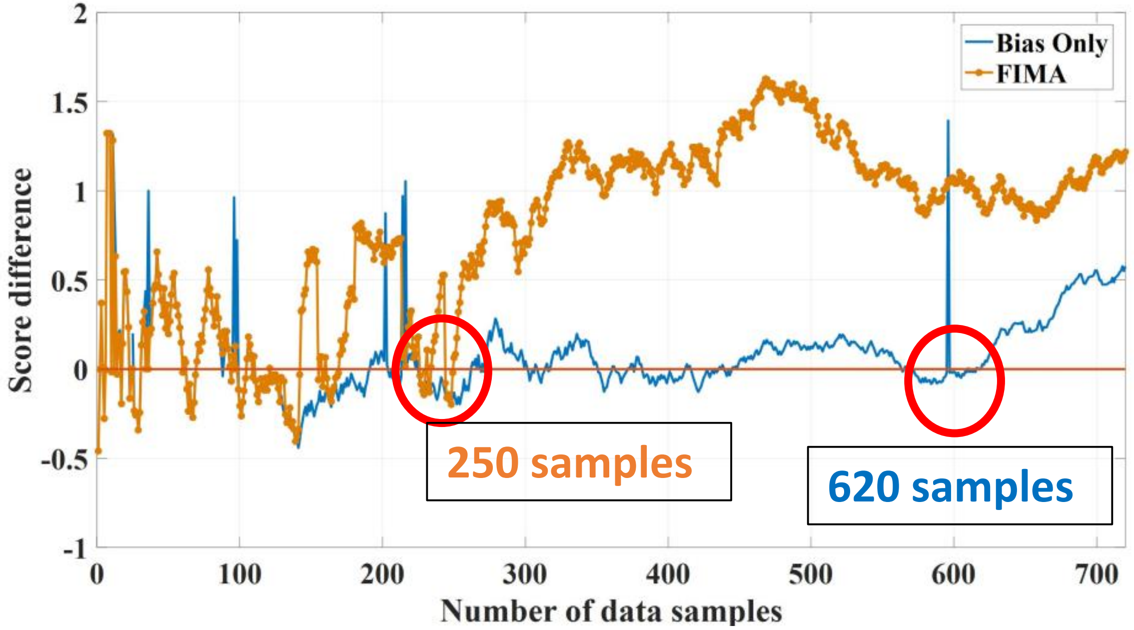
If correct key guess changes, set $P_{Fail} = 1$ and continue.

Return Correct Subkey.

Results

required data samples (no countermeasures)





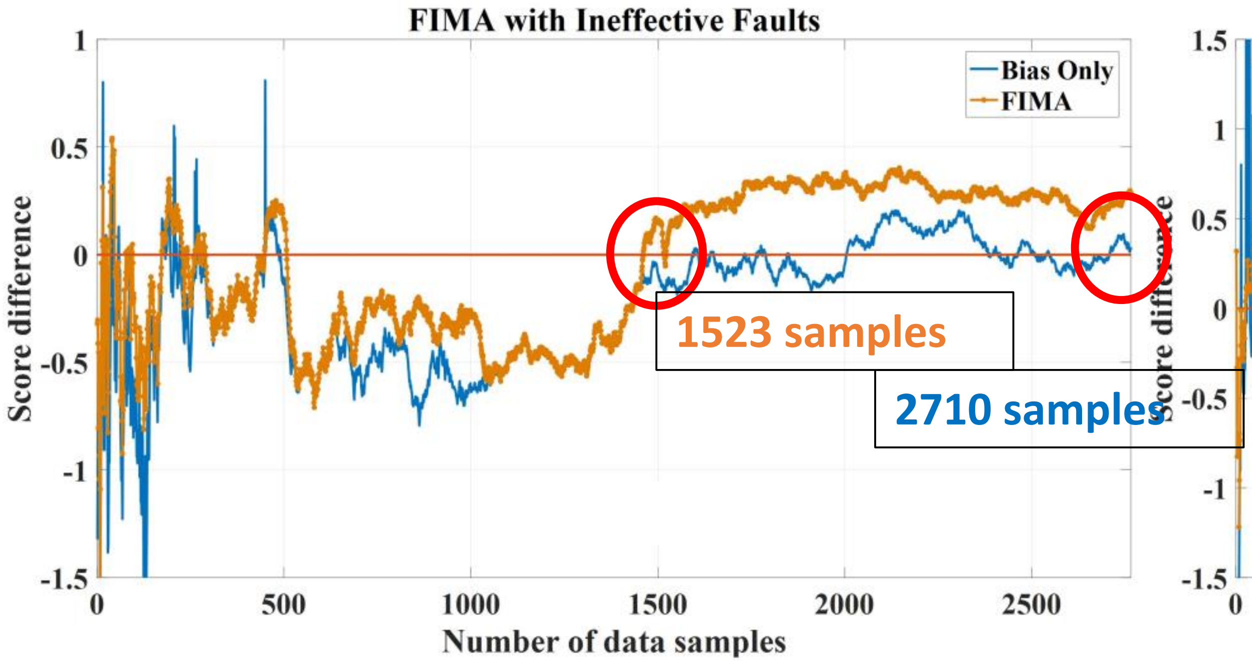
20 fault intensity values in [0, 0.2]

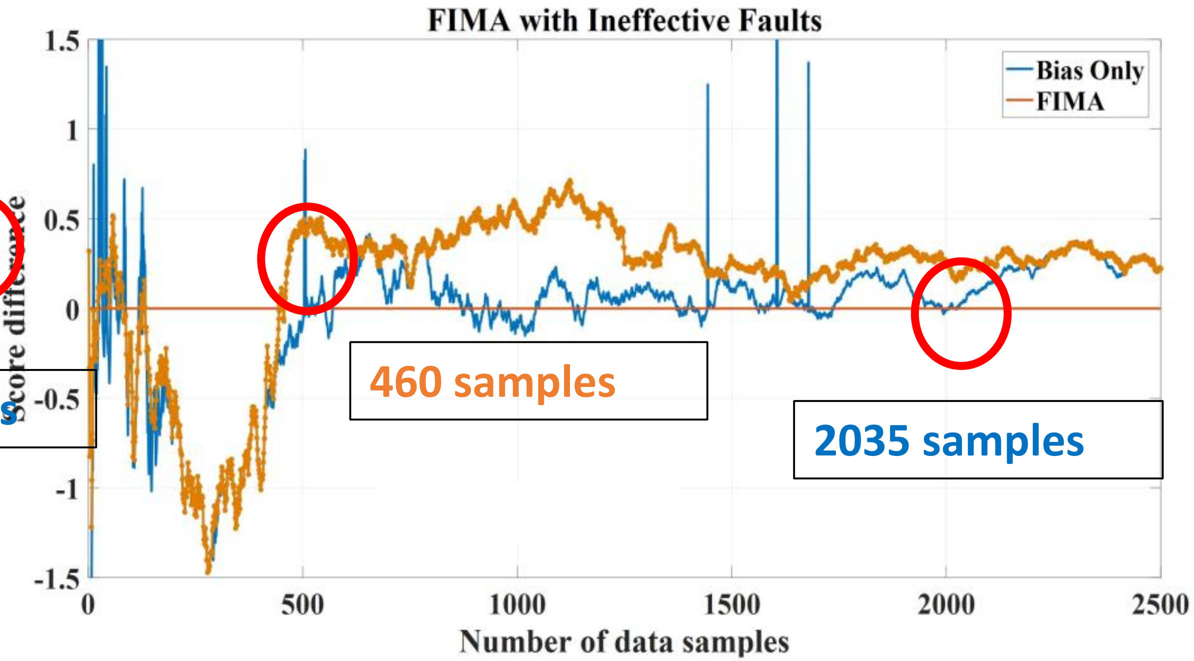
Score difference of correct key and next highest guess diverges

20 fault intensity values in [0, 0.3]

Higher fault intensity reduces number of required samples

required data samples (error-detection countermeasures)

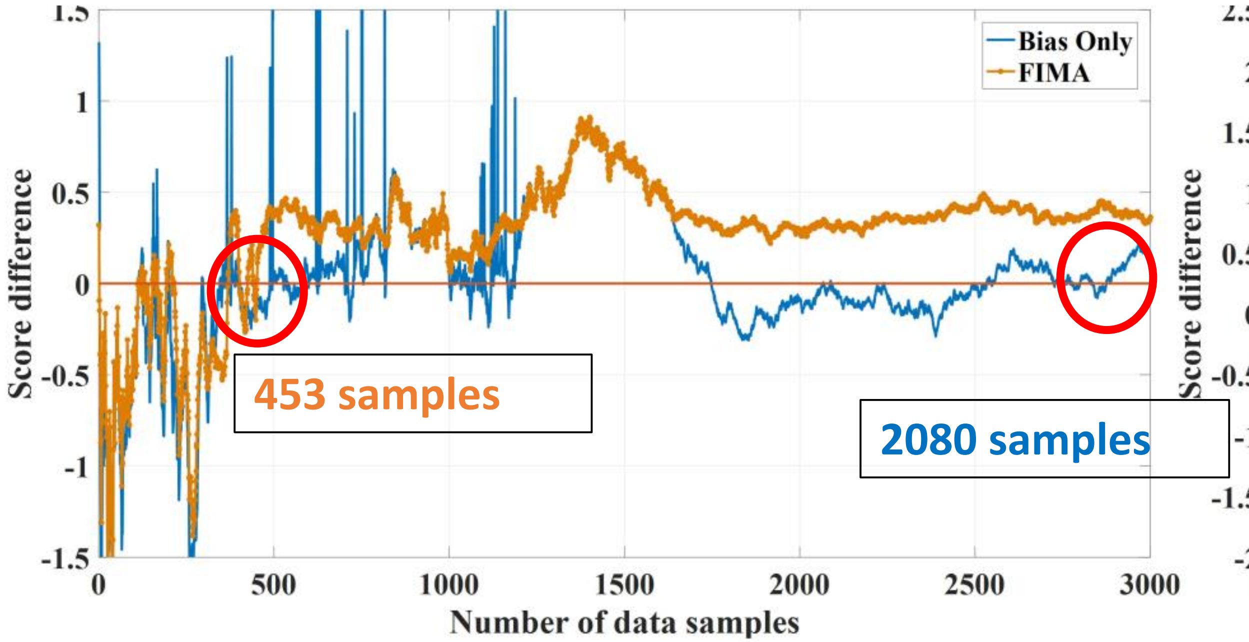




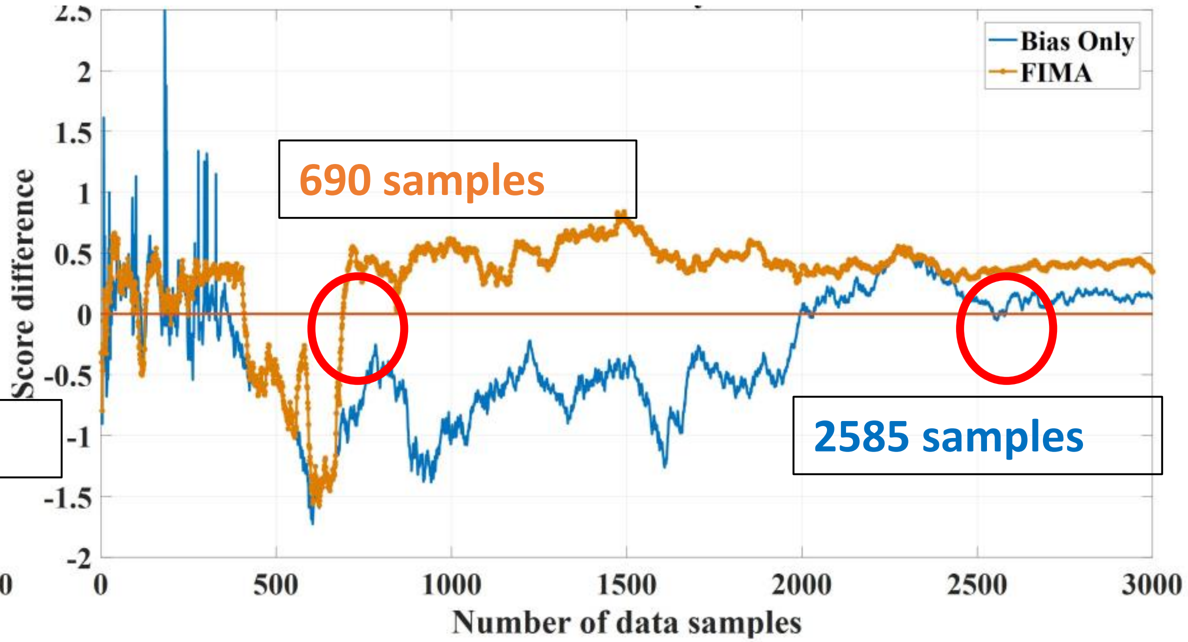
Error detection countermeasures suppress faulty output; defeat differential fault analysis attacks
Bias (SIFA and FIMA) and intensity variation (FIMA) of ineffective faults still leak information about intermediate variable
However, bias decreases, and required # of samples increases

required data samples (*infective* countermeasures or *noisy faults*)

Infective countermeasures



Noisy faults



Infective countermeasures randomize fault, which reduces bias
Therefore, intensity provides relatively much better information than bias alone.

Noisy fault injections, where attacker does not have precise control (e.g., timing, location)

Comparison to bias-based technique

Intensity Range	$p \in [0, 0.2]$		$p \in [0, 0.3]$		$p \in [0, 0.3]$		$p \in [0, 0.3]$	
Technique	FIMA	Bias	FIMA	Bias	FIMA	Bias (SIFA)	FIMA	Bias
Countermeasure	N/A		N/A		Error-Detection		Infective	
Data size	305	710	250	620	460	2035	453	2880
FIMA improvement	2.3×		2.5×		4.4×		6.3×	

Conclusions and Future Directions

Conclusions

- Introduced Fault Intensity Map Analysis (FIMA)
  - ➢Statistical analysis technique (SIFA + FSA)
  - ➢Uses fault bias + intensity disposition
- Recovered 128-bit secret key of Ascon
- Improvements over previous bias-based techniques
  - ➢More than 2x improvement in efficiency
  - ➢Grows to 6x in presence of countermeasures

Future Directions

- Improved classifiers for FIMA score
- Different assumptions on fault models
- Development of countermeasures
- Investigation of FIMA on other ciphers
- Effect of FIMA in presence of SCA countermeasures

Questions?