Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 10-B, pp. 4169–4178
DOI : 10.47974/JDMSC-3041

Fault injection attacks on finite field arithmetic implementations

Aabha Amey Patil *
Department of Information Technology
Shree L. R. Tiwari College of Engineering
Thane 401107
Maharashtra
India

Nilesh Jagdish Vispute †
Department of Artificial Intelligence and Data Science
Vidyavardhini's College of Engineering & Technology
Vasai 401202
Maharashtra
India

Uma Bhavin Goradiya §
Department of Computer Engineering
Shree L. R. Tiwari College of Engineering
Thane 401107
Maharashtra
India

Prachi Wagde ‡
Department of Artificial Intelligence & Machine Learning
Shri Bhaghubhai Mafatlal Polytechnic and College of Engineering
Ville Parle
Mumbai 400056
Maharashtra
India

* *E-mail:* aabha.patil@slrtce.in (Corresponding Author)
† *E-mail:* visputenilesh47@gmail.com
§ *E-mail:* uma.goradiya@slrtce.in
‡ *E-mail:* prachi.wagde@sbmp.ac.in

© Taru Publications

Swapna Nikhil Ambekar [@]
Department of Information Technology
Shri Bhaghubhai Mafatlal Polytechnic and College of Engineering
Ville Parle
Mumbai 400056
Maharashtra
India

Saurav Verma [#]
Department of Information Technology
Mukesh Patel School of Technology Management & Engineering
Mumbai 400056
Maharashtra
India

Abstract

Fault injection attacks are a major threat on the reliability and security of cryptographic systems using finite field arithmetic. This paper explores how arithmetic processes like addition, multiplication and inversion in finite fields are affected by deliberate errors. The paper examines the different fault injection methods such as voltage glitching, clock manipulation, and laser-based fault injection methods to discuss its usefulness in interference with computations. An in-depth analysis of vulnerabilities points to the short-term and long-term faults that might cause leakage of information and key recovery in cryptographic implementations. Also, attack methods, common fault, on symmetric and asymmetric schemes are discussed. Experimental evaluation reveals that there is quantifiable deterioration in computational accuracy and there is higher vulnerability to side-channeling. The results provide an insight into the importance of effective fault detection and mitigation measures to increase the resiliency of cryptographic hardware and software systems.

Subject Classification: *68M11, 68M25.*

Keywords: *Fault injection attacks, Finite field arithmetic, Cryptographic security, Side-channel attacks, Hardware vulnerability, Fault detection mechanisms.*

I. Introduction

Fault injection attacks have become a major challenge to the current cryptographic systems and especially those that use finite field arithmetic to make secure calculations. The mathematical basis of popular cryptographic schemes, such as elliptic curve cryptography, block ciphers, and error-correcting codes, is based on finite fields [1, 2]. These applications usually presuppose an error-free working environment, but it is possible that attackers may intentionally cause temporary or permanent

[@] *E-mail:* **swapna.ambekar@sbmp.ac.in**
[#] *E-mail:* **saurav.verma@nmims.edu**

faults during calculations to breach the integrity of the system [3, 4, 5]. Voltage glitching, clock manipulation, laser injection and electromagnetic interference are a few of the techniques that attackers can use to modify the intermediate values resulting in inaccurate outputs that can be used to extract secret keys [6, 7]. Finite field operations such as modular multiplication, inversion and exponentiation are also vulnerable to such attacks as they are the most important targets of these attacks [8]. The paper is aimed at learning about the impact of fault injection on arithmetic operations and the propagation of the induced errors in cryptographic operations [9, 10]. Through the review of attack models and vulnerability of the system, the study will serve to point out the need to integrate fault-resistance principles of design and powerful detection algorithms to have a secure and trustful cryptographic implementation [11].

II. Fault Injection Techniques and Platforms

Fault injection methods take advantage of physical or logical failures in hardware platforms to introduce computing errors so that attackers can bend cryptographic operations to obtain sensitive data [12, 13].

A. *Voltage Glitching and Clock Manipulation*

Timing or power anomalies due to voltage glitching and manipulation of clock lines used to compute the voltage can result in incorrect execution of instructions, data corruption, or even omission of an operation in the implementation of a finite field arithmetic.

Fault-Affected Signal Representation

$$x'(t) = x(t) + \delta_{v(t)} + \delta_{c(t)} \tag{1}$$

Equation (1) represents the fault-corrupted signal, in which the original signal is corrupted by the components of voltage glitch and clock disturbance, which are the combined temporal errors during the execution of calculations.

Voltage Glitch Modeling

$$\delta_{v(t)} = \int (0 \; to \; t)\left[V_{glitch(\tau)} \cdot e^{-\lambda (t-\tau)} d\tau \right] \tag{2}$$

Voltage glitch behavior is defined in equation (2) as an exponentially decaying perturbation that is integrated with respect to time, where the transient variations in power are characterized and the overall effect of these variations on signal integrity.

Clock Disturbance Modeling

$$\delta_{c(t)} = \Sigma(k = 1 \; to \; N)[\, \Delta f_k \cdot \sin(2\pi \, f_k t + \varphi_k)] \tag{3}$$

Clock disturbance is modeled in equation (3) as an addition of deviations of sinusoidal frequencies, which are the timing anomalies that affect coordinated operations in the finite field arithmetic processing units.

Faulty Arithmetic Computation

$$y = (a \cdot b + \delta_v + \delta_c) mod\ p \tag{4}$$

Error Magnitude Estimation

$$e(t) = \sqrt{\int (0\ to\ T)\left[\left(x(t) - x'(t)\right)^2 dt \right]} \tag{5}$$

In equation (4), faulty arithmetic computation is represented: with injected disturbances on the multiplication operation, the result of the modular computation is erroneous and will not coincide with the correct result of finite field computation. Equation (5) measures the magnitude of errors by considering an integral-based Euclidean distance, which is a measure of error between original and faulted signals through time, thus indicating the magnitude and propagation of injected faults.

Fault Occurrence Probability

$$P_{fault} = 1 - \exp\left(- \int (0\ to\ T) \left| \delta_{v(t)} + \delta_{c(t)} \right| dt \right) \tag{6}$$

Equation (6) approximates the probability of fault occurrence with decay of accumulated disturbances being exponential, and it is clear that larger values of the integrated perturbations mean that probability of successful fault injection during execution is higher.

B. *Laser and Electromagnetic Fault Injection Methods*

Localized or non-invasive faults are induced by laser and electromagnetic techniques, and focused on particular regions of a circuit to interrupt computations and reveal internal states by means of controlled propagation of faults.

Spatial Fault Injection Representation

$$x'(i,j) = x(i,j) \oplus L(i,j) \oplus EM(i,j) \tag{7}$$

Equation (7) is spatial fault injection in which original data matrix is perturbed by laser and electromagnetic noise, with bitwise perturbations introduced that have a strong local impact on areas of computation.

Laser-Induced Fault Modeling

$$L(i,j) = \int (0\ to\ T)\left[\alpha \cdot I_{laser(t)} \cdot \delta(i - i^0, j - j^0) dt \right] \tag{8}$$

The model of laser-induced faults (Equation (8)) relies on the time-integration of energy at certain spatial coordinates to describe localized disruption due to the intensity of lasers and its effects on the targeted circuit components.

Electromagnetic Disturbance Modeling

$$EM(i,j) = \Sigma(k = 1\ to\ M)\left[\beta_k \cdot H_{k(t)} \cdot \cos(\omega_k t) \right] \tag{9}$$

According to equation (9), electromagnetic disturbance can be described as a summation of oscillatory magnetic field components, which are an induced interference that alters the behavior of signals on a variety of spatial and time frequencies.

Faulty Field Computation

$$y = (x'(i,j) \cdot k) mod\ q \tag{10}$$

Equation (10) is a form of faulty field computation in which injected spatial faults alter input values to produce faulty modular multiplication output that does not match the correct finite field arithmetic computations.

Spatial Error Estimation

$$e_{spatial} = \sqrt{\Sigma(i,j)\left[\left(x(i,j) - x'(i,j)\right)^2\right]} \tag{11}$$

Equation (11) measures spatial error based on Euclidean distance among elements of the matrix, differences between the original and faulted states in order to determine the strength and propagation of injected faults.

Injection Effectiveness Ratio

$$P_{inject} = \frac{\left(\int_A |L(i,j) + EM(i,j)| dA\right)}{\left(\int_A |x(i,j)| dA\right)} \tag{12}$$

Injection effectiveness ratio (Eq. 12) is the ratio of disturbance energy injected into the entire signal to the signal original magnitude, which determines the strength of the effects of laser and electromagnetic faults on the integrity of computations.

III. Vulnerability Analysis of Finite Field Operations

Finite field computations are also susceptible to fault tolerance attacks where errors in intermediate values allow the computation of differences and the extraction of secret keys by comparing inaccurate results in multiplication, inversion and exponentiation computations.

Original Field Operation

$$z = (a \cdot b)\ mod\ p \tag{13}$$

The right-hand side of equation (13) is the correct multiplication of finite fields, where the operands are multiplied and cut off by a prime, and is the foundation of examining deviations due to faults.

Fault-Induced Computation

$$z' = (a \cdot b + \varepsilon_1)\ mod\ p \tag{14}$$

Error Term Formulation

$$\varepsilon^1 = \int (0\ to\ n)[\, e(x)^2 dx\,] + \Sigma(k = 1\ to\ n)\eta_k \tag{15}$$

In equation (14), fault-induced computation takes into account an error term in the multiplication process, which causes faulty modular output that is not equal to the desired value to be computed. The composite error term (Equation (15) represents a combination of integrated noise energy and discrete perturbations, a combined contribution of both continuous and stochastic faults to the accuracy of computations.

Key Leakage Optimization Model

$$L(k) = argmin_k \Sigma(i = 1\ to\ N)\left[\left(z_i - z'_{i(k)}\right)^2\right] \tag{16}$$

Equation (16) is a formulation of key leakage as an optimization problem to minimize squared discrepancies between correct and faulty outputs in order to determine the most likely secret key using differential analysis.

Leakage Probability Estimation

$$P_{leak} = 1 - \exp\left(-\left(\frac{||\Delta z||^2}{\sigma^2}\right)\right) \quad (17)$$

Equation (17) approximates the leakage probability based on an exponential relationship between size of deviations between the correct and faulty outputs, which implies that the bigger the gaps between the correct and faulty outputs the higher the probability of rich information leakage.

A. *Proposed Adaptive Fault-Resilient Finite Field Security Framework*

The framework being designed is an adaptive, secure finite field arithmetic implementation that is fault resilient in a hostile hardware environment. Runtime Consistency Verification and Transient Anomaly Analysis are used to continuously monitor the architecture for modular multiplication, inversion and exponentiation processes. A special purpose fault localization module recognizes voltage glitches, Clock manipulation, and Electromagnetic disturbances as causing abnormal arithmetic transitions. When the fault probability is higher than the security levels that are pre-defined in the framework, recovery and redundancy mechanisms will be activated dynamically. Adaptive monitoring greatly minimizes propagation of errors, increases the effectiveness of attacks detection and increases the level of resilience against attacks on the key extraction, in particular, differential fault analysis and side channel attacks at implementation level. In Figure 1, layered fault injection perturbs the computations of finite fields.

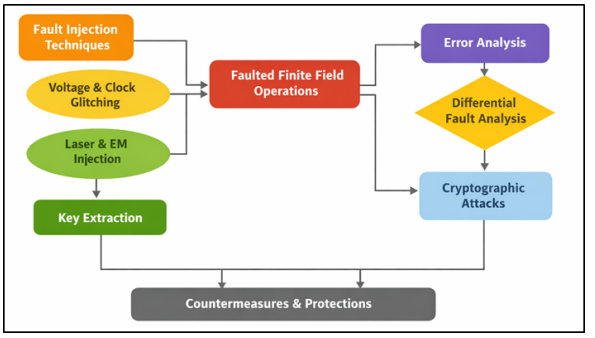


Figure 1

Multilayer Fault Injection Framework for Finite Field Arithmetic Implementations

IV. Fault Attack Methodologies on Cryptographic Schemes

To recover secret keys and break algorithmic security guarantees, fault attack methodologies take advantage of induced faults in cryptographic schemes, using differential fault analysis and statistical methods to effectively recover secret keys and compromise algorithmic security guarantees.

Encryption Process Modeling

$$C = Enc_{k(m)} = f(m,k) \tag{18}$$

Equation (18) represents the encryption process as a deterministic function of plaintext and key into the ciphertext, which sets the appropriate operating level of analyzing the deviation when there is a fault condition.

Faulty Ciphertext Generation

$$C' = f(m,k) + \varepsilon_f \tag{19}$$

Equation (19) is an incorrect generation of ciphertext in which an induced error affects the output of the encryption function, resulting in the generation of incorrect ciphertext that can be used to do cryptanalytic fault analysis.

Ciphertext Difference Analysis

$$\Delta C = C - C' = -\varepsilon_f \tag{20}$$

Equation (20) introduces ciphertext difference, which is the difference between correct and faulty outputs and isolates the injected error term and allows differentiating fault analysis to extract information.

$$k = argmax_k \Pr(\Delta C \mid k) \tag{21}$$

The secret key is estimated by the maximization of conditional probability of observed differences in equation (21) and finding the most probable key that was consistent with induced faults and observed ciphertext differences.

$$Success = \lim(N \to \infty)\left[\left(\frac{1}{N}\right) \cdot \Sigma(i = 1\ to\ N) I(k_i = k)\right] \tag{22}$$

Attack success is defined as the ratio between the number of successfully recovered keys and the number of attempts to recover the key (statistically) by using equation (22).

V. Result and Analysis

A comparative analysis of fault injection methods on finite field arithmetic implementations is given in Table 1. The fault rate of laser injection is the highest (27.8) and the key recovery success is the highest (66.3) which means that laser injection has good attack effectiveness.

The electromagnetic injection also experiences a high impact with 62.7% key recovery and 73.4% error propagation. Clock manipulation is moderately vulnerable, and voltage glitching has a relatively low fault rate

(18.6%), as well as attack success (54.8%). Figure 2 is a comparison of performance measures of the various fault injection techniques.

Table 1
Performance Impact of Fault Injection on Finite Field Arithmetic Implementations

Fault Technique	Fault Rate (%)	Detection Rate (%)	Error Propagation (%)	Key Recovery Success (%)
Voltage Glitching	18.6	72.4	65.2	54.8
Clock Manipulation	22.1	69.7	71.5	58.6
Laser Injection	27.8	81.2	76.9	66.3
Electromagnetic Injection	24.5	78.6	73.4	62.7

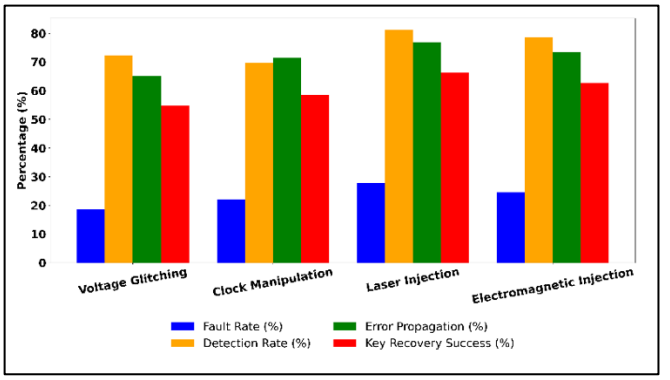


Figure 2
Fault Injection Technique Performance Comparison

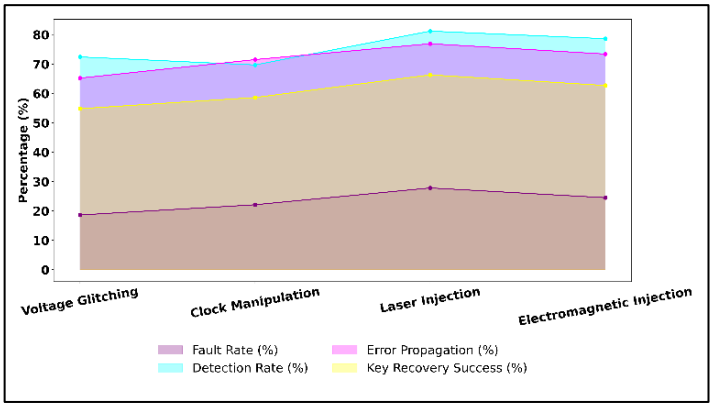


Figure 3
Fault Attack Behavior and Impact Analysis

The detection rates of techniques are quite high, especially laser injection (81.2%). In Figure 3, the behavior of fault attacks has an implication on the performance of cryptographic systems.

In general, an increase in the fault rates is associated with a higher error propagation and better key recovery efficiency.

VI. Conclusion

The security of cryptographic systems that are implemented using finite field arithmetic is a seriously challenging problem due to fault injection attacks. This paper has further shown the way in which different fault injection methods such as voltage glitching, clock manipulation, or laser or electromagnetic methods, can interfere in arithmetic operations and reveal sensitive information. The vulnerability analysis was able to confirm the fact that even small bugs in the middle-level calculations can spread and facilitate successful key recovery using differential fault analysis. The experimental findings revealed the great effects on the computational correctness and system reliability when there is a fault condition on the system. The results indicate that effective countermeasures such as fault detection, redundancy, and error-correcting systems should be included. In general, it is important to enhance the security of the implementation level to provide resilience to the changing physical attack vectors within the current cryptographic environment.

References

[1] X. Hu and Z. Qin, "A secure architecture for modular division over a prime field against fault injection attacks," *Appl. Sci.*, vol. 10, no. 5, Art. no. 1700 (2020), doi: 10.3390/app10051700.

[2] A. Gangolli, Q. H. Mahmoud, and A. Azim, "A systematic review of fault injection attacks on IoT systems," *Electronics*, vol. 11, no. 13, Art. no. 2023 (2022), doi: 10.3390/electronics11132023

[3] R. Sheikh and R. Cammarota, "Improving performance and mitigating fault attacks using value prediction," *Cryptography*, vol. 2, no. 4, Art. no. 27 (2018), doi: 10.3390/cryptography2040027.

[4] H. Jiang, X. Zhu, and J. Han, "Instruction-fetching attack and practice in collision fault attack on AES," *Symmetry*, vol. 14, no. 10, Art. no. 2201 (2022), doi: 10.3390/sym14102201.

[5] H. Li, K. Qiao, Y. Xu, C. Ou, and A. Wang, "General extensions and improvements of algebraic persistent fault analysis," *Cryptography,* vol. 9, no. 2, Art. no. 30 (2025), doi: 10.3390/cryptography9020030.

[6] H. Mestiri, I. Barraj, M. Bedoui, and M. Machhout, "An ASCON AOP-SystemC environment for security fault analysis," *Symmetry*, vol. 16, no. 3, Art. no. 348 (2024), doi: 10.3390/sym16030348.

[7] S. Narasimhan, M. J. Ellis, and N. H. El-Farra, "Detection of multiplicative false data injection cyberattacks on process control systems via randomized control mode switching," *Processes*, vol. 12, no. 2, Art. no. 327 (2024), doi: 10.3390/pr12020327.

[8] M. I. Huilgol and S. Asok, "PBIB-designs with constant block sums arising from different graph parameters," *J. Discrete Math. Sci. Cryptogr.*, vol. 28, no. 7, pp. 2697–2721 (2025), doi: 10.47974/JDMSC-2064.

[9] Z. Ahmed, M. Nasir, D. M. Alsekait, M. Z. H. Shah, D. S. AbdElminaam, and F. Ahmad, "Control conditions for equal power sharing in multi-area power systems for resilience against false data injection attacks," *Energies*, vol. 17, no. 22, Art. no. 5757 (2024), doi: 10.3390/en17225757.

[10] H. P. Morgan, N. Dimitrov, and P. Laurent, "A systematic review of graph-theoretic approaches to blockchain consensus mechanisms: Methods, architectures, and future research directions," *Int. J. Recent Adv. Eng. Technol.*, vol. 14, no. 2, pp. 182–190 (2025).

[11] K. Mahmoud and N. Nicolici, "ALPRI-FI: A framework for early assessment of hardware fault resiliency of DNN accelerators," *Electronics*, vol. 13, no. 16, Art. no. 3243 (2024), doi: 10.3390/electronics13163243.

[12] R. C. Poonia, R. Almakki, A. K. J. Saudagar, A. Altameem, and M. Albathan, "An optimization approach for real-time object detection in IoT devices through edge computing and deep learning," *J. Inf. Optim. Sci.*, vol. 45, no. 5, pp. 1465–1475 (2024), doi: 10.47974/JIOS-1768.

[13] R. Gangarde, A. Julian, M. S. Murali Dhar, G. I. Mary, M. Rele, and M. Vaithianathan, "Applications of partial differential equations and Green's functions in climate change modeling," *J. Interdiscip. Math.*, vol. 28, no. 6, pp. 2079–2088 (2025), doi: 10.47974/JIM-2349.

Received April, 2026