DOI: 10.19884/j.1672-5220.202305005

Statistical Fault Analysis of Lightweight Tweakable Block Cipher QARMA in the Internet of Everything

LI Jiayao, LI Wei*, GAO Jianning, QIN Mengyang, SUN Wenqian
College of Computer Science and Technology, Donghua University, Shanghai 201620, China

Abstract: Based on the ciphertext-only attack (COA) assumption, the statistical fault analysis (SFA) is proposed to break all versions of QARMA in the Internet of Everything (IoE), where suitable strategies are taken into consideration for the uncertainty of tweaks to cover more rounds of fault injections. It also presents the novel double distinguishers of Cramér-von Mises test-Hamming weight (CM-HW) and Kuiper's test-maximum likelihood estimation (KT-MLE) to improve the attacking efficiency. According to the experimental results, the attackers can inject 374 and 726 random faults into the deeper antepenultimate round to recover 128-bit and 256-bit secret keys of QARMA with a reliability of at least 99%, respectively. Hence, QARMA is vulnerable to the SFA in the IoE. The results offer a valuable reference for the lightweight tweakable cryptosystems with the reflection structure and the protection of the cryptographic devices.

Key words: Internet of Everything (IoE); side-channel analysis; lightweight tweakable block cipher; statistical fault analysis (SFA); QARMA
CLC number: TP309 **Document code**: A
Article ID: 1672-5220(2024)02-0172-12

Open Science Identity (OSID)

0 Introduction

The Internet of Everything (IoE) brings together data, things and people to associate networks more closely and valuably than ever before. It turns data into actions that create rich experiences, new abilities and passionate opportunities for individuals, companies, cities, regions and countries. Large-scale implementations with convenience and reliability are provided for logistics applications, intelligent buildings, animal searching, medical care, precision farming, entertainment, transportation, weather forecasts, environmental protection, etc.[1-6] The IoE comprises dimensional distributed independent objects taking sensors to track environmental or physical conditions. As Fig. 1 shows, it integrates a gateway that supplies wireless connection back to the wired world and distributed sensors.

However, due to the dynamic and connected configuration, the IoE faces numerous threats, such as interruption, interception, modification and fabrication. Thus, secure correspondence is extremely significant in the IoE. Sensors are reliable when every correspondence has been initiated from a trusty source, and the attackers do not manipulate messages. These issues in the IoE may become similar to those in the classical computer networks. However, some unique properties of the IoE, such as the mobility of sensors, the self-organized nature and the catastrophic consequences of security failures can cause more critical security. Conventional cryptosystems cannot meet the requirements of low-resource devices which have extreme power and memory constraints. It is imperative to apply the cryptosystems to satisfy high security and low power consumption in the IoE. Under this circumstance, lightweight cryptosystems have been designed and applied for encryption, authentication, digital signature, etc.[7-11] Therefore, lightweight cryptosystems can reduce power consumption for devices, and provide secure network connectivity to low-resource embedded devices.

QARMA, as a lightweight tweakable block cipher with a reflection structure, plays a vital role in data confidentiality, integrity and authentication in the IoE. The lightweight tweakable block cipher QARMA was proposed by Avanzi in 2017[12]. It can be applied to classical memory encryption, the design of keyed hash functions and the generation of short tags for hardware and software implementations. Its characteristics of low power consumption and high efficiency are suitable for data protection of the limited-resources devices in the IoE. QARMA has flexible options for the 64-bit and 128-bit block sizes and the corresponding 128-bit and 256-bit secret key sizes, respectively. The tweak can change at a lower cost, avoiding altering the secret key frequently.

Received date: 2023-05-14
Foundation items: National Natural Science Foundation of China (Nos. 61772129 and 61932014); National Cryptography Development Fund, China (No. MMJJ20180101)
* Correspondence should be addressed to LI Wei, email: weili@dhu.edu.cn
Citation: LI J Y, LI W, GAO J N, et al. Statistical fault analysis of lightweight tweakable block cipher QARMA in the Internet of Everything [J]. *Journal of Donghua University (English Edition)*, 2024, 41(2): 172-183.

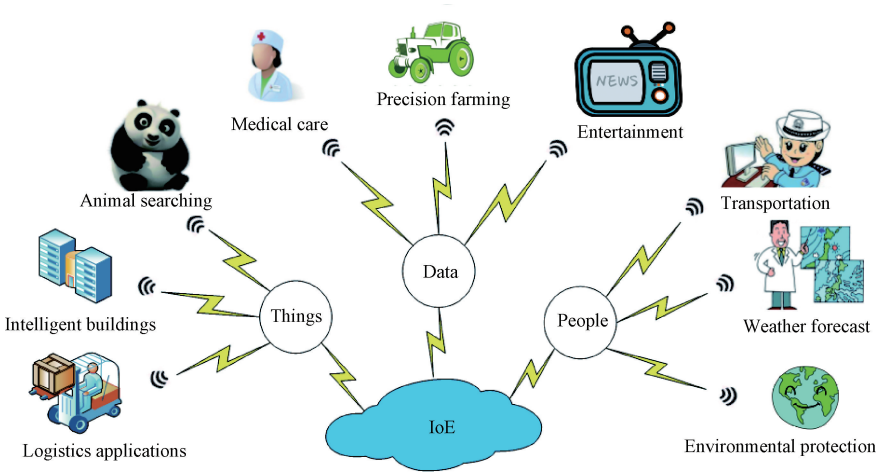

Fig. 1 The IoE scenario

QARMA is a typical reflection cipher, and its encryption is identical to the decryption[13]. In the design of a reflection cipher, the robust reflector and key expansion can affect the security of the general structure[14-15]. QARMA adopts the three-round Even-Mansour scheme to be secure against classical cryptanalysis and has the advantage of low-latency hardware implementation. Since the publication of the QARMA cipher, differential analysis, linear analysis, algebraic analysis, statistical saturation analysis, meet-in-the-middle analysis, impossible differential analysis and other cryptanalysis technologies have studied the security of all versions of QARMA, including QARMA-128 and QARMA-256[12,16-20].

1 Related Work

In 2018, Li et al.[16] leveraged the meet-in-the-middle analysis with the linear relation of the MixColumns to analyze the 9-round QARMA. In 2019, Li et al.[17] proposed the statistical saturation analysis on the 11-round QARMA with the tweak differential invariant bias. Later, Zong et al.[18] presented an impossible differential analysis with the mixed integer linear programming (MILP) technique and designed a series of 7-round related-tweak impossible differential distinguishers on the 10-round QARMA-128. In 2020, on the properties of the linear operations and the redundancy of the key schedule, Liu et al.[19] improved the impossible differential analysis of the 11-round QARMA-128 and QARMA-256, and the meet-in-the-middle analysis of 12-round QARMA-256, respectively.

As one typical side-channel analysis, fault analysis has played a crucial role in evaluating the implementation security of cryptosystems since 1997[21]. By injecting faults into the running cryptographic devices, the attackers can quickly break the cryptosystems with incorrect information. The faults are triggered by laser, magnet, voltage, glitch, or modification of the code in the hardware or software. Based on the implementation security, there are several fault analyses, such as algebraic fault analysis (AFA)[22], differential fault analysis (DFA)[23], impossible differential fault analysis (IDFA)[24], meet-in-the-middle fault analysis (MITMFA)[25], and statistical fault analysis (SFA)[26-29].

On the attackers' ability to capture information, the basic assumption of cryptanalysis can be roughly classified as the chosen-ciphertext attack (CCA), chosen-plaintext attack (CPA), known-plaintext attack (KPA), ciphertext-only attack (COA), etc. As shown in Table 1, the classical cryptanalysis of QARMA focuses on the assumptions of KPA and CPA. However, the known or designated plaintexts are hard to control in the actual IoE scenario. The attackers may have the weakest ability to obtain random ciphertexts only. It is necessary to discuss the security analysis of QARMA against the COA attackers to challenge the harsh environment.

Table 1 Summary of the cryptanalysis of the QARMA cipher

Cryptanalysis	Assumption	Round number*	Ref.
Linear analysis	KPA	3/3	[12]
Statistical saturation analysis	KPA	10/11	[17]
Differential analysis	CPA	3/3	[12]
Impossible differential analysis	CPA	11/11	[19]
Meet-in-the-middle analysis	CPA	—/12	[19]
SFA with novel distinguishers	COA	16/22	This paper

Note: * means the total number of attacking rounds.

In 2013, Fuhr et al.[26] made a significant step forward in the ciphertext-only fault analysis to capture the advanced encryption standard (AES) secret key with the assumption of COA. As shown in Table 1, based on the random byte-oriented fault model, they designed three classic distinguishers of squared Euclidean imbalance (SEI), Hamming weight (HW) and maximum likelihood estimation (MLE) to recover the secret key with 320 random faults. In 2016, Dobraunig et al.[27] leveraged the statistical fault attacks on a list of authenticated encryption schemes and implemented the attack on the hardware. Later, Li et al.[28-29] proposed the SFA with several distinguishers, including goodness-of-fit (GF), maximum a posteriori (MAP) and goodness of fit-squared Euclidean imbalance (GF-SEI), on the LED and Simeck lightweight ciphers. There is no research on the lightweight tweakable block cipher against the SFA. As for the classical cryptanalysis, the tweak is public for all parties. However, the tweak may be out of control and become unknown owing to the severity of the IoE scenario. And critically, each cipher may have its exquisite distinguishers to the challenges of attacking efficiency.

This study presents the SFA with novel distinguishers to break QARMA with a reflection structure. The attackers can use the previous six distinguishers of SEI, HW, MLE, GF, MAP and GF-SEI, and the novel distinguishers of Cramér-von Mises test-Hamming weight (CM-HW) and Kuiper's test-maximum likelihood estimation (KT-MLE), respectively. Compared with the security analysis of AES and LED in Table 2, the proposed SFA can extend fault locations into the deeper antepenultimate round of QARMA. The SFA method takes suitable strategies to deal with the known and unknown tweaks. The experimental results show that there are only 374 and 726 faults to recover 128-bit and 256-bit secret keys of QARMA on average, respectively.

The rest of this paper is structured as follows. Section 2 introduces the notations and the structure of QARMA. Section 3 proposes the SFA on QARMA, combining the different strategies with the known and unknown tweaks. Section 4 describes the simulation experiments by analyzing the accuracy, reliability, number of faults, complexity and latency. The last section summarizes this paper.

Table 2 Comparison of the SFA in recovering the secret keys of AES, LED and QARMA

Distinguishers	AES-128[26]			QARMA-256			LED-64[28]			QARMA-128		
	Model	Location①	#Faults②	Model	Location①	#Faults②	Model	Location①	#Faults②	Model	Location①	#Faults②
SEI	Byte	$L-1$	320	Byte	$L-2$	556	Nibble	$L-1$	560	Nibble	$L-2$	1 164
HW	Byte	$L-1$	288	Byte	$L-2$	470	Nibble	$L-1$	312	Nibble	$L-2$	812
MLE	Byte	$L-1$	224	Byte	$L-2$	438	Nibble	$L-1$	320	Nibble	$L-2$	800
MAP	—	—	—	Byte	$L-2$	438	Nibble	$L-1$	304	Nibble	$L-2$	812
GF	—	—	—	Byte	$L-2$	105	Nibble	$L-1$	480	Nibble	$L-2$	1 708
GF-SEI	—	—	—	Byte	$L-2$	566	Nibble	$L-1$	424	Nibble	$L-2$	1 280
KT-MLE	—	—	—	Byte	$L-2$	396	—	—	—	Nibble	$L-2$	780
CM-HW	—	—	—	Byte	$L-2$	374	—	—	—	Nibble	$L-2$	726

Notes: ①Location means the round of the first fault injection, where L represents the total number of rounds in each cipher; ② #Faults means the number of fault injections.

2 Specification of QARMA

2.1 Notations

$-$, $\sim$ and $\hat{}$ denote the inverse of an operation, the observed value, and the theoretical value of an element, respectively.

X, Y and K denote the plaintext, the ciphertext and the secret key, respectively.

k_0 and k_1 denote two subkeys.

w_0 and w_1 denote two whitening keys.

n and m denote the block size and the cell size, respectively, where $n \in \{64, 128\}$ and $m \in \{4, 8\}$.

$\mathcal{R}$, $\overline{\mathcal{R}}$ and $\mathcal{P}$ denote the forward round function, the backward round function and the pseudo-reflector, respectively.

r denotes the number of rounds in the forward or backward round.

$T_{i,j}$ denotes the jth cells of a tweak in the ith round, where $i \in [0, r-1]$ and $j \in [0, 15]$.

t_i denotes the tweakey in the ith round function, where $i \in [0, r-1]$.

h and ω denote the permutation and the linear feedback shift register (LFSR) in the tweak update function, respectively.

α and c_i denote the round constants used in QARMA, respectively, where $i \in [0, r-1]$.

$A_{i,j}$, $B_{i,j}$, $C_{i,j}$ and $D_{i,j}$ denote the jth cell of the output of $\overline{\mathrm{SC}}$, $\overline{\mathrm{MC}}$, $\overline{\mathrm{PC}}$ and $\overline{\mathrm{ART}}$ in the ith backward round, respectively, where $i \in [0, r-1]$; $j \in [0, 15]$; SC means SubCells; MC means MixColumns; PC means ShuffleCells; ART means AddRoundTweakey.

$\oplus$, $\odot$, $\|$ and $/$ denote the bitwise-XOR, bitwise-AND, the concatenation, and the division, respectively.

$\lll$, $\ggg$, $\ll$ and $\gg$ denote the left circular shift, the right circular shift, the left logical shift, and the right logical shift, respectively.

2.2 Structure of QARMA

QARMA supports 64-bit and 128-bit blocks, and has the corresponding 128-bit and 256-bit secret keys, as shown in Table 3. It composes encryption, decryption, key expansion and a tweak update function.

Table 3 Versions of QARMA

Block size	Key size	Cell size	Number of rounds
64	128	4	12, 14, 16
128	256	8	20, 22, 24

As shown in Fig. 2, the encryption structure comprises the forward rounds, the central construction and the backward rounds. The forward rounds consist of r-round functions of $\mathcal{R}$, where $r \in \{5, 6, 7\}$ in QARMA-128, and $r \in \{9, 10, 11\}$ in QARMA-256. Each $\mathcal{R}$ is composed of four operations as follows:

1) ART where the round tweakey is XORed with the state;

2) PC which has the Midori-like permutation of the state;

3) MC where the state multiplies the diffusion matrix;

4) SC which substitutes the state with the S-Box.

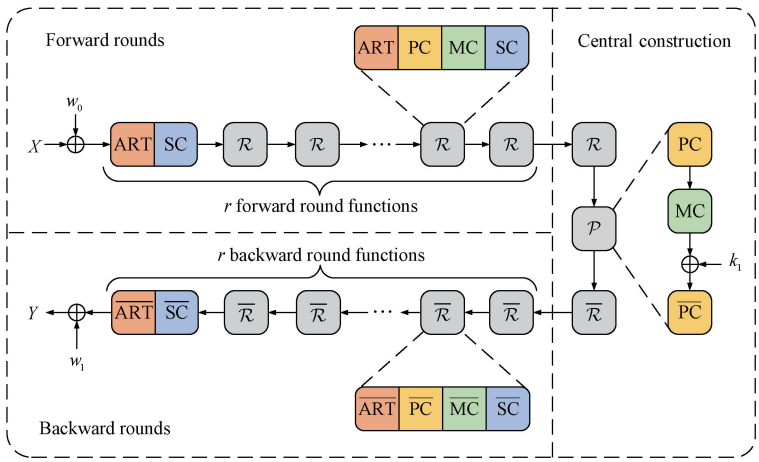

Fig. 2 Structure of QARMA cipher

Similar to the AES final round, the short version of $\mathcal{R}$ omits the ShuffleCells and the MixColumns operations. The backward rounds also comprise the r-round functions of $\overline{\mathcal{R}}$, which is in reverse order of $\mathcal{R}$ with the above inverse operations as $\overline{\text{ART}}$, $\overline{\text{PC}}$, $\overline{\text{MC}}$ and $\overline{\text{SC}}$. The central construction includes a forward round function $\mathcal{R}$, a pseudo-reflector $\mathcal{P}$ and a backward round function $\overline{\mathcal{R}}$. The tweak update function, the encryption and the key expansion are shown in Algorithm 1, Algorithm 2 and Algorithm 3.

Algorithm 1 The tweak update function

Input: T_0
Output: T_1, T_2, $\cdots$, T_r

FOR $i = 0$ TO $r - 1$ DO
  $T_{i+1} = h(T_i)$;
  FOR $j = 0, 1, 3, 4, 8, 11, 13$ DO
    $T_{i+1,j} = ((((T_{i+1,j} \odot (m/2)) \gg (m/4)) \oplus (T_{i+1,j} \odot 1)) \ll (m-1)) \oplus (T_{i+1,j} \gg 1)$;
  END FOR
END FOR
RETURN T_1, T_2, $\cdots$, T_r

Algorithm 2 The encryption

Input: X, w_0, w_1, k_0, k_1, T_0
Output: Y

Run Algorithm 1;
$S = X \oplus w_0$;
FOR $i = 0$ TO $r - 1$ DO
  $S = \mathcal{R}(S, k_0 \oplus T_i \oplus \mathfrak{c}_i)$;
END FOR
$S = \mathcal{R}(S, w_1 \oplus T_r)$;
$S = \mathcal{P}(S, k_1)$;
$S = \overline{\mathcal{R}}(S, w_0 \oplus T_r)$;
FOR $i = r - 1$ TO 0 DO
  $S = \overline{\mathcal{R}}(S, k_0 \oplus T_i \oplus \mathfrak{c}_i \oplus \alpha)$;
END FOR
$Y = S \oplus w_1$;
RETURN Y

Algorithm 3 The key expansion

Input: K
Output: k_0, k_1, w_0, w_1

$w_0 \| k_0 = K$;
$w_1 = (w_0 \ggg 1) \oplus (w_0 \gg (16m - 1))$;
$k_1 = k_0$;
RETURN k_0, k_1, w_0, w_1

3 SFA of QARMA

3.1 Basic assumption and fault model

The proposed SFA method is based on the COA assumption, where the attackers can collect the corresponding ciphertexts only with a single key. The fault model is random nibble-oriented for QARMA-128, and byte-oriented for QARMA-256, respectively. The attackers can inject faults into the encryption, and take advantage of the bitwise-AND operation with the non-uniform distribution. The distributions of the affected nibbles and bytes are depicted in Fig. 3.

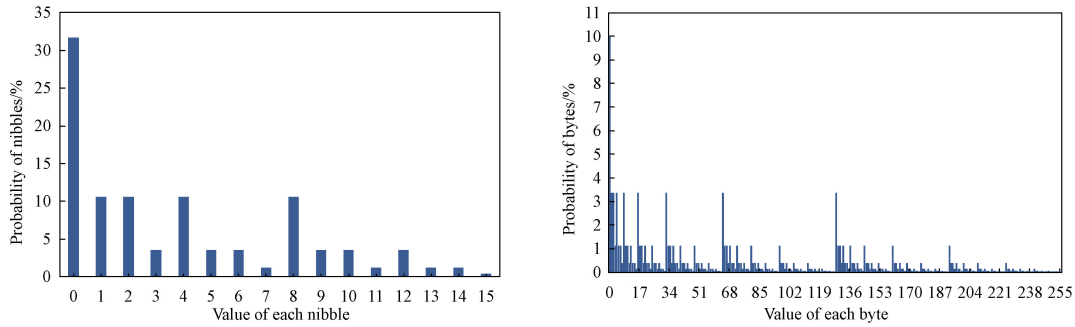

Fig. 3 Distributions of fault injection: (a) nibble fault for QARMA-128; (b) byte fault for QARMA-256

3.2 Main process with the unknown tweak

Generally, the tweak requires to be changed for each encryption to keep the secret key fixed and steady. In classical cryptanalysis, the tweak is often regarded as one public parameter, and the attackers can obtain its value at any time besides the ciphertexts. While in the remote and harsh environment of the IoE, it could be the case that the tweak may be unknown. For instance, a device can be out of control, and cannot trigger off the tweak at one time. It is hard for the attackers to get the tweak value sustainably. On the basis of various values of r, the attackers can take the corresponding strategy to recover the secret key.

3.2.1 Attacking steps with $r \in \{7, 9, 10, 11\}$

The primary process takes the differential strategy and comprises the following four steps.

Step 1 The attackers induce random m-bit faults in the encryption and collect the faulty ciphertexts, where $m \in \{4, 8\}$. The positions of faults can be $D_{2,j}$, $C_{2,j}$ or $B_{2,j}$ in the backward round function, where $j \in [0, 15]$. Figure 4 shows the propagation path of fault injections in $B_{2,0}$.

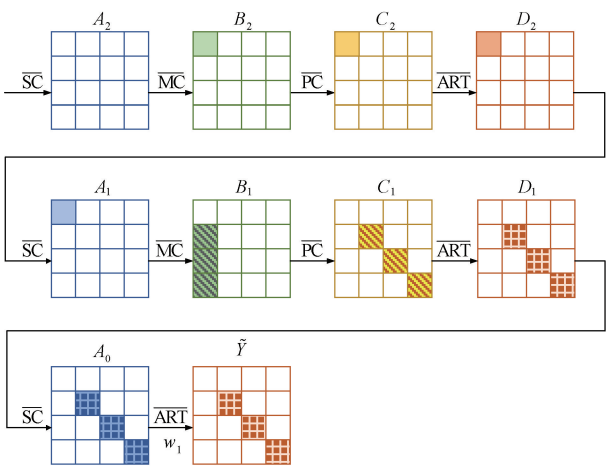

Fig. 4 Fault propagation in backward round function of QARMA

When the tweak is unknown, the attackers can utilize the tweakey candidates and the faulty ciphertexts to calculate the intermediate state $\tilde{A}_{1,j}$ as:

$$\begin{aligned}\tilde{A}_{1,j} &= \mathrm{MC}(\mathrm{PC}(\mathrm{SC}(\tilde{Y} \oplus t_0 \oplus w_1) \oplus t_1)) \\ &= \mathrm{MC}(\mathrm{PC}(\mathrm{SC}(\tilde{Y} \oplus t_0 \oplus w_1)) \oplus \mathrm{PC}(t_1)) \\ &= \mathrm{MC}(\mathrm{PC}(\mathrm{SC}(\tilde{Y} \oplus t_0 \oplus w_1))) \oplus \mathrm{MC}(\mathrm{PC}(t_1)),\end{aligned}$$

where each round tweakey candidate is the guess value of $t_0 \oplus w_1$, and corresponds to one set of $\tilde{A}_{1,j}$. The value of $\mathrm{MC}(\mathrm{PC}(t_1))$ can be omitted because the faults do not affect it. Then the attackers can calculate as follows:

$$\tilde{A}_{1,j} = \mathrm{MC}(\mathrm{PC}(\mathrm{SC}(\tilde{Y} \oplus t_0 \oplus w_1))).$$

Recovering all cells of the round tweakey relies on the position of the affected nibbles or bytes. Figure 5 summarizes the relations between the value of j in $\tilde{B}_{2,j}$ and the recoverable cells. For example, when $j = 0$, the 5th, 10th and 15th cells of the round tweakey can be recovered, respectively.

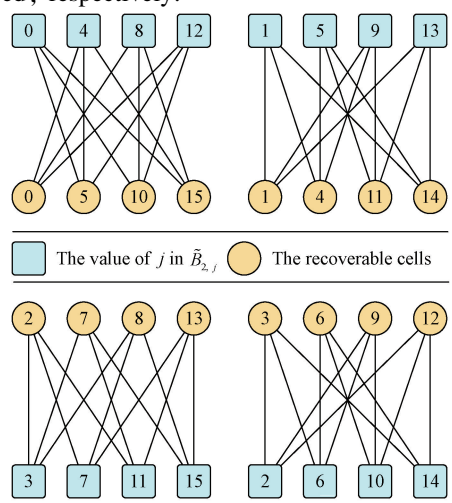

Fig. 5 Relations between recoverable cells of round tweakey and positions of fault injections

Step 2 The attackers use statistical relations to distinguish the expected round tweakey. They need to calculate the distinguishing value for every set of the intermediate states. Then, the candidate, corresponding to the minimal or the maximal value, is the expected round tweakey. The details of eight distinguishers are introduced in Sections 3.4 and 3.5, respectively. The attackers can pick up a suitable distinguisher, and recover $3m$ bits of $t_0 \oplus w_1$, as illustrated in Fig. 5. Then they modify the positions of fault injections, and repeat the above steps until all bits of $t_0 \oplus w_1$ are recovered.

Step 3 After recovering $t_0 \oplus w_1$, the attackers can calculate the value of $k_0 \oplus w_1 \oplus T_0$ by

$$k_0 \oplus w_1 \oplus T_0 = (t_0 \oplus w_1) \oplus \alpha \oplus \mathfrak{c}_0.$$

Owing to the XOR operations, the attackers cannot divide k_0 and w_1 without the help of other relations, whether the tweak T_0 is known or not. Hence, other round tweakeys must be recovered to calculate the values of k_0, w_1, and T_0.

Similarity to the recovery of $t_0 \oplus w_1$, the attackers can induce the faults in B_3, C_3, or D_3, and then repeat Steps 1 and 2 until t_1 are recovered. The value of t_2, $t_3, \cdots$ and t_6 do likewise. After recovering t_1, $t_2, \cdots$ and t_6, the attackers have

$$t_i = k_0 \oplus T_i \oplus \alpha \oplus \mathfrak{c}_i,$$

where $i \in [0, r-1]$. For example, the values of T_1 and T_2 are calculated as

$$\begin{cases} T_1 = k_0 \oplus t_1 \oplus \alpha \oplus \mathfrak{c}_1, \\ T_2 = k_0 \oplus t_2 \oplus \alpha \oplus \mathfrak{c}_2. \end{cases}$$

Then,

$$\begin{aligned} T_1 \oplus T_2 &= k_0 \oplus t_1 \oplus \alpha \oplus \mathfrak{c}_1 \oplus k_0 \oplus t_2 \oplus \alpha \oplus \mathfrak{c}_2 \\ &= t_1 \oplus t_2 \oplus \mathfrak{c}_1 \oplus \mathfrak{c}_2. \end{aligned}$$

According to the tweak update function, the relation between T_1 and T_2 is deduced as

$$T_2 = \omega(h(T_1)).$$

Thus,

$$T_1 \oplus \omega(h(T_1)) = t_1 \oplus t_2 \oplus \mathfrak{c}_1 \oplus \mathfrak{c}_2.$$

The above differential equation is helpful in calculating some cells of T_1. The cells are insoluble when the differential values are equal to zero. The reason is that the LFSR ω does not affect all the cells during the iterations in the tweak update function. Hence, the other differential equations, i.e., $T_1 \oplus T_2$, $T_1 \oplus T_3, \cdots$ and $T_1 \oplus T_6$ are necessary, which correspond to different recoverable cells in T_1. The details are shown in Table 4. After calculating all the cells of T_1, the attackers can derive the original tweak T_0 as

$$T_0 = \bar{h}(\bar{\omega}(T_1)).$$

Table 4 Relations between the differential equations and the recoverable cells in T_1

Differential equation	Recoverable cell in T_1
$T_1 \oplus T_2$	0, 4, 5, 6, 7, 9, 15
$T_1 \oplus T_3$	1, 2, 3, 4, 5, 7, 9, 11, 12, 15
$T_1 \oplus T_4$	1, 3, 7, 8, 9, 11, 12, 14
$T_1 \oplus T_5$	3, 8, 9, 10, 12
$T_1 \oplus T_6$	8, 9, 12, 13

Step 4 On the values of $t_0 \oplus w_1$, t_1, T_0 and T_1, the attackers can calculate k_0, w_1, w_0 and K as

$$\begin{cases} k_0 = t_1 \oplus T_1 \oplus \alpha \oplus \mathfrak{c}_1, \\ w_1 = (t_0 \oplus w_1) \oplus k_0 \oplus T_0 \oplus \alpha \oplus \mathfrak{c}_0, \\ w_0 = (((w_1 \gg (n-2)) \odot 1) \oplus w_1) \lll 1, \\ K = w_0 \parallel k_0. \end{cases}$$

3.2.2 Attacking steps with $r \in \{5, 6\}$

Because the LFSR ω in the tweak update function only changes the value of specific tweak cells, the permutation h must make ω modify the other tweak cells. Furthermore, the structure of h shows that all cells of the original tweak can be affected by ω if and only if the tweak update function iterates at least five times. As a result, the attackers have to recover six-round tweakeys by the SFA to construct five differential equations.

However, as shown in Table 4, the attackers can only leverage $t_1, t_2, \cdots$ and t_{r-1} to construct five differential equations, i.e., $r \geqslant 7$. And $w_0 \oplus T_r$ cannot construct the differential equations because of the whitening key w_0. Hence, the differential strategy in Section 3.2.1 is unavailable for QARMA-128 when $r \in \{5, 6\}$.

In the case of $r \in \{5, 6\}$, the attackers modify the round of fault injection and repeat Steps 1 and 2 until k_1 in the central construction is recovered. Then, they derive the values of $(t_0 \oplus w_1), t_1, t_2, \cdots, t_{r-1}, (T_r \oplus w_0)$ and k_1 one after another. Finally, the secret key K can be calculated as follows:

$$\begin{cases} k_0 = k_1, \\ T_0 = \bar{h}(\bar{\omega}(t_1 \oplus k_0 \oplus \alpha \oplus \mathfrak{c}_1)), \\ w_1 = (t_0 \oplus w_1) \oplus k_0 \oplus T_0 \oplus \alpha \oplus \mathfrak{c}_0, \\ w_0 = (((w_1 \gg (n-2)) \odot 1) \oplus w_1) \lll 1, \\ K = w_0 \parallel k_0. \end{cases}$$

3.3 Main process with the known tweak

Usually, the tweak can be altered frequently, and it is a public variable for all parties. The attackers can observe the tweak, and use it directly in the SFA of QARMA. Similar to Section 3.2.1, the attackers inject random faults in the encryption, and the fault propagation is shown in Fig. 4. They unite the subkeys candidates with the faulty ciphertexts to calculate the intermediate state $\tilde{A}_{1,j}$ as follows:

$$\begin{cases} t_0 = k_0 \oplus T_0 \oplus \alpha \oplus \mathfrak{c}_i, \\ \tilde{A}_{1,j} = \mathrm{MC}(\mathrm{PC}(\mathrm{SC}(\tilde{Y} \oplus t_0 \oplus w_1))). \end{cases}$$

Because of the known tweak T_0, the value of $T_0 \oplus \alpha \oplus \mathfrak{c}_0$ can be viewed as a constant. And

$$\tilde{A}_{1,j} = \mathrm{MC}(\mathrm{PC}(\mathrm{SC}(\tilde{Y} \oplus k_0 \oplus w_1 \oplus T_0 \oplus \alpha \oplus \mathfrak{c}_0))),$$

where each subkey candidate represents the guess value of $k_0 \oplus w_1$ and corresponds to one set of the intermediate states $\tilde{A}_{1,j}$.

Subsequently, the attackers calculate the distinguishing value of every set of $\tilde{A}_{1,j}$. And the candidate, which corresponds to the minimal or the maximal distinguishing value, is the expected subkeys. The attackers can recover $3m$ bits of $k_0 \oplus w_1$ by picking up a suitable distinguisher. All eight distinguishers are listed in Sections 3.4 and 3.5. The attackers modify the positions of fault injections and repeat the above steps until all bits of $k_0 \oplus w_1$ are recovered.

Due to the XOR operation, it is impossible to do the partition of $k_0 \oplus w_1$. To separate $k_0 \oplus w_1$ into k_0 and w_1, the attacker must induce the faults in B_3, C_3 or D_3, and repeat the above steps, until k_0 in t_1 is recovered. After separating w_1 and k_0 through the SFA, the secret key K is calculated as

$$K = w_0 \| k_0 = ((((w_1 \gg (n-2)) \odot 1) \oplus w_1) \lll 1) \| k_0.$$

3.4 Classic distinguishers

3.4.1 SEI description

SEI compares the proximity between the observed sample and the uniform distribution. In the SFA of AES, Fuhr et al.[26] applied the SEI distinguisher in the recovery of the secret key. The attackers can distinguish the expected candidates by calculating the distribution of all observed cells. The SEI distinguisher is expressed as

$$\mathfrak{D}_{\mathrm{SEI}} = \sum_{q=0}^{2^m-1} \left(\frac{\tilde{u}_q}{f} - \frac{1}{2^m} \right)^2,$$

where q denotes the value of a nibble or byte; $\tilde{u}_q$ represents the observed number of q, $q \in [0, 2^m - 1]$, $m \in \{4, 8\}$; f denotes the number of fault injections. The expected round tweakey corresponds to the maximal SEI value.

3.4.2 HW description

The HW introduced by Reed[30] is equivalent to the HW in the binary case. It calculates the distance from a binary string to an all-zero string. In other words, the HW counts the amount of '1' in a specific binary string. The HW distinguisher, applied in the SFA of AES, is described as

$$\mathfrak{D}_{\mathrm{HW}} = \frac{1}{f} \sum_{q=0}^{2^m-1} (\tilde{u}_q \cdot H(q)),$$

where $H(\cdot)$ represents the Hamming distance. The expected round tweakey corresponds to the minimal HW value.

3.4.3 MLE description

MLE estimates the unknown distribution with the application of the known distribution[31]. On the assumption in Section 3.1, the probability distribution of the bitwise-AND is shown in Fig. 3. The MLE distinguisher is denoted as

$$\mathfrak{D}_{\mathrm{MLE}} = \sum_{q=0}^{2^m-1} (\tilde{u}_q \cdot \log_2^{\mathcal{F}(q)}),$$

where $\mathcal{F}(\cdot)$ represents the probability function. The expected round tweakey corresponds to the maximal MLE value.

3.4.4 MAP description

MAP applies the Bayesian statistical model to distinguish the expected round tweakey. Different from the MLE distinguisher, the MAP distinguisher should calculate the prior hypothesis in advance, as shown in Fig. 3. The MAP distinguisher is defined as

$$\mathfrak{D}_{\mathrm{MAP}} = \frac{\mathcal{F}(\tilde{A}_{1,j} \mid k_w) \cdot \mathcal{G}(k_w)}{\sum_{w=0}^{W-1} \mathcal{F}(\tilde{A}_{1,j} \mid k_w) \cdot \mathcal{G}(k_w)},$$

where k_w represents a guess value of the round tweakey, $w \in [0, W-1]$; W represents the size of the candidates set; $\tilde{A}_{1,j}$ denotes the observed value of the internal state; $\mathcal{F}(\tilde{A}_{1,j} \mid k_w)$ denotes the distribution function of $\tilde{A}_{1,j}$ with k_w; $\mathcal{G}(k_w)$ denotes the prior function of k_w. In the SFA of QARMA, $W = 2^{3m}$. The expected round tweakey corresponds to the maximal MAP value.

3.4.5 GF description

GF tests whether a distribution fits another. In the SFA, the attackers calculate the value of GF for each round tweakey candidate between the observed distribution and the theoretical distribution. The GF distinguisher is expressed as

$$\mathfrak{D}_{\mathrm{GF}} = \sum_{q=0}^{2^m-1} \frac{(\tilde{u}_q - \hat{u}_q)^2}{\hat{u}_q},$$

where $\hat{u}_q$ represents the theoretical number of q. The expected round tweakey corresponds to the minimal GF value.

3.4.6 GF-SEI description

GF-SEI as a double distinguisher, associates the characteristics of the GF distinguisher with the SEI distinguisher. At first, the GF distinguisher can reduce the number of candidates as

$$\mathfrak{D}_{\mathrm{GF}} = \sum_{q=0}^{2^m-1} \frac{(\tilde{u}_q - \hat{u}_q)^2}{\hat{u}_q}.$$

Then, the SEI distinguisher picks up the expected round tweakey from the remaining candidates. It is calculated as

$$\mathfrak{D}_{\mathrm{SEI}} = \sum_{q=0}^{2^m-1} \left(\frac{\tilde{u}_q}{f} - \frac{1}{2^m} \right)^2.$$

The expected round tweakey corresponds to the minimal GF value and the maximal SEI value.

3.5 Novel distinguishers

3.5.1 KT-MLE description

KT-MLE combines two single distinguishers of MLE and KT to be a novel double distinguisher, aiming to improve the attacking efficiency of SFA. The novel KT is usually used to test whether an observed sample fits the given distribution. The compatibility between the designative distribution and the samples is KT's statistic. The values of the MLE distinguisher are calculated as

$$\mathfrak{D}_{\mathrm{MLE}} = \sum_{q=0}^{2^m-1} (\tilde{u}_q \cdot \log_2^{\mathcal{T}(q)}).$$

And the unexpected candidates whose values are less than the maximal value of MLE are omitted. Then, the remaining candidates are filtered by the distinguisher of KT as follows:

$$\mathfrak{D}_{\mathrm{KT}} = \sup_{q\in[0,\,2^m-1]} \left| \frac{\tilde{u}_q}{f} - \mathcal{F}(q) \right| + \sup_{q\in[0,\,2^m-1]} \left| \mathcal{F}(q+1) - \frac{\tilde{u}_{q+1}}{f} \right|.$$

On the basic assumption, the candidate is the expected round tweakey if and only if the values of the MLE and KT distinguishers are both maximal.

3.5.2 CM-HW description

CM-HW is a new double distinguisher that unites two single distinguishers of CM and HW. The CM test is a criterion for calculating the difference between two distributions. The novel CM distinguisher picks up the candidate, corresponding to the minimal statistic as

$$\mathfrak{D}_{\mathrm{CM}} = \frac{1}{12f} + \sum_{x=0}^{f-1} \left(\frac{2x+1}{2f} - \mathcal{F}(\tilde{A}_{1,j} \mid k_w) \right)^2.$$

Then, the HW distinguisher gets rid of the unexpected candidates by

$$\mathfrak{D}_{\mathrm{HW}} = \frac{1}{f} \cdot \sum_{q=0}^{2^m-1} (\tilde{u}_q \cdot H(q)) .$$

Eventually, the remaining candidate corresponding to the minimal HW value is the expected round tweakey.

4 Simulation

The software simulation is implemented in a personal computer with a 3.6 GHz AMD processor and 16 GB RAM. And the brute-force search of the round tweakey runs on the Cuda with the NVIDIA GeForce GTX 1660 GPU. With 10 000 experiments, accuracy, reliability, the number of faults, complexity and latency are applied to estimate the performance of distinguishers and the attacking efficiency of the SFA. Each experiment includes the fault injections, the intermediate states' derivation and the key recovery.

4.1 Accuracy

Accuracy describes how many candidates are considered the expected round tweakey by a distinguisher at the attacking procedure. A distinguisher exhibits high accuracy when only one candidate is regarded as the expected round tweakey. So the root mean-square error (RMSE) is a vital index to describe the accuracy of a distinguisher. The RMSE E_{RMS} is calculated as

$$E_{\mathrm{RMS}} = \sqrt{\frac{1}{N}\sum_{\varepsilon=1}^{N} (\varphi(\varepsilon) - 1)},$$

where $\varphi(\varepsilon)$ denotes the number of candidates considered as the expected round tweakey by a distinguisher; N represents the number of repetition experiments for a distinguisher in the same number of fault injections and $N=10\ 000$; ε denotes the numbering of an experiment. Therefore, a distinguisher is more accurate when the RMSE is closer to zero with fewer faults. Figure 6 shows the RMSE of recovering $3m$-bit of the round tweakey by the distinguishers for QARMA. The RMSEs of all distinguishers can approach zero when the number of fault injections is sufficient. The novel distinguishers of CM-HW and KT-MLE achieve the top three RMSE with fewer faults in QARMA-128 and QARMA-265, respectively.

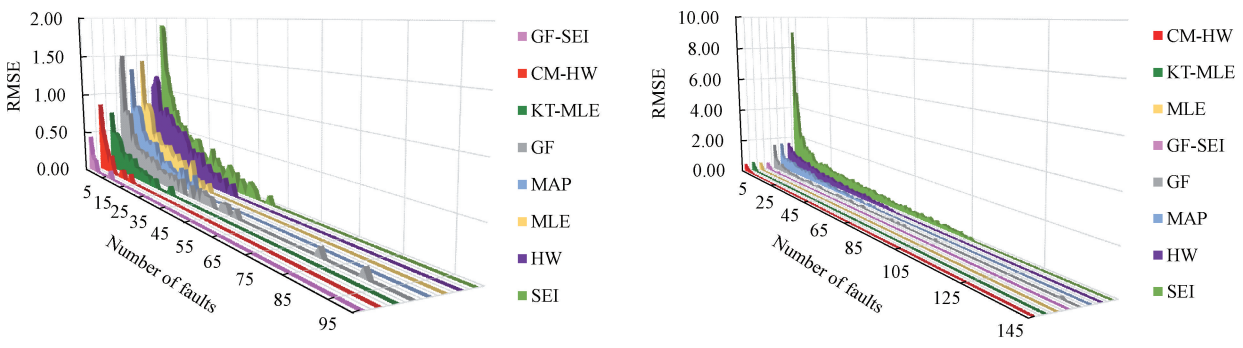

Fig. 6 RMSE of recovering $3m$-bit of the round tweakey by the distinguishers for QARMA: (a) QARMA-128; (b) QARMA-256

4.2 Reliability

Reliability represents the successful probability in the SFA of QARMA. It can estimate the performance of each distinguisher instinctively. Reliability of recovering $3m$-bit of the round tweakey by the distinguishers for QARMA is shown in Fig. 7. Two subfigures stand for two versions of QARMA, and the colored lines denote the states of all distinguishers. With the increment of fault injections, the reliabilities of SEI, HW, MLE, MAP, GF, GF-SEI, KT-MLE and CM-HW can

gradually achieve 99% in the SFA of QARMA-128. As for QARMA-256, however, the reliability of GF is less than 80%. Thus, the distinguisher of GF is not suggested in the SFA of QARMA-256. Compared with the previous distinguishers, the reliability of the novel distinguishers can reach 99% with fewer faults, as shown in Fig. 7.

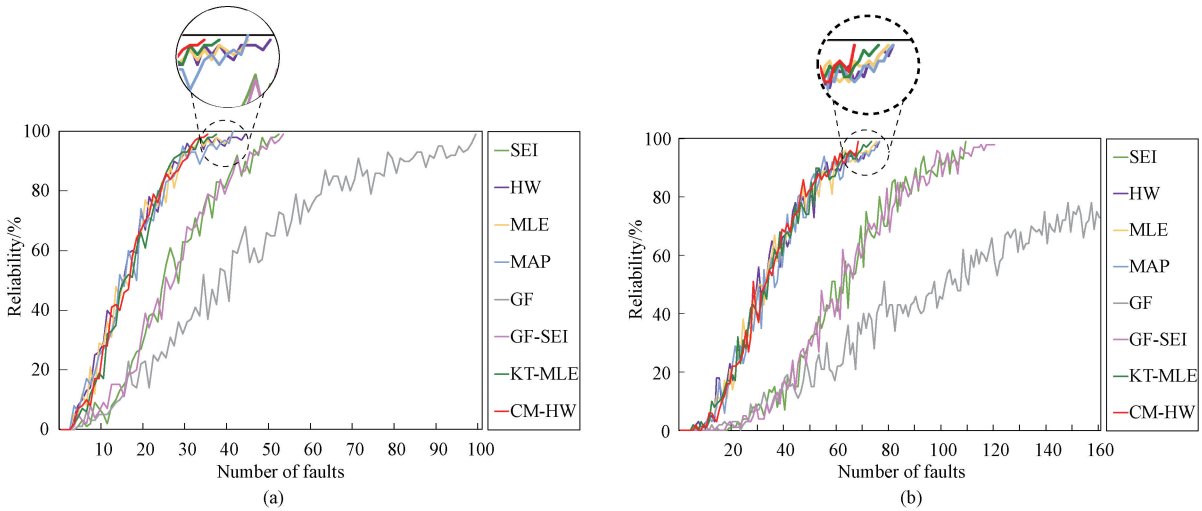

Fig. 7 Reliability of recovering $3m$-bit of the round tweakey by the distinguishers for QARMA: (a) QARMA-128; (b) QARMA-256

4.3 Number of faults

The number of faults is an important measure of the attack efficiency of the distinguishers in the SFA. Table 5 summarizes the least number of faults when each distinguisher achieves its highest reliability in the recovery of the secret key of QARMA. The distinguishers of SEI, HW, MLE, MAP, GF, GF-SEI, KT-MLE and CM-HW require 1 163, 811, 800, 811, 1 707, 1 280, 779 and 726 faults in the recovery of the whole QARMA-256 secret key with the maximal reliability, respectively. And they also require 555, 470, 438, 438, 1 056, 566, 395 and 374 faults to break the version of QARMA-128, respectively. According to the experimental result, fewer faults are required in the SFA of QARMA by the novel distinguishers of KT-MLE and CM-HW.

Table 5 The number of faults in recovering the secret key of QARMA when each distinguisher achieves its highest reliability

Distinguishers	QRAMA-128		QARMA-256	
	Reliability/%	#Faults	Reliability/%	#Faults
SEI	≥99	555	≥99	1 163
HW	≥99	470	≥99	811
MLE	≥99	438	≥99	800
MAP	≥99	438	≥99	811
GF	≥99	1 056	≤80	1 707
GF-SEI	≥99	566	≥99	1 280
KT-MLE	≥99	395	≥99	779
CM-HW	≥99	374	≥99	726

4.4 Complexity

The complexities are composed of time complexity and data complexity. Attacking all versions of QARMA-256 shares the same complexities because only five round tweakeys are necessary for the secret key derivation. As for QARMA-128, different quantities of round tweakeys are recovered, which results in different complexities. Table 6 summaries complexities in recovering the secret key of QARMA when each distinguisher achieves its highest reliability. Based on the experimental results, the novel proposed distinguishers of KT-MLE and CM-HW require lower time and data complexities.

Table 6 Complexities in recovering the secret key of QARMA when each distinguisher achieves its highest reliability

Distinguishers	Time complexity				Data complexity			
	QARMA-128			QARMA-256	QARMA-128			QARMA-256
	$r=5$	$r=6$	$r=7$	$r\in\{9,10,11\}$	$r=5$	$r=6$	$r=7$	$r\in\{9,10,11\}$
SEI	$2^{26.29}$	$2^{26.51}$	$2^{26.52}$	$2^{39.58}$	$2^{22.29}$	$2^{22.51}$	$2^{22.52}$	$2^{35.58}$
HW	$2^{26.14}$	$2^{26.36}$	$2^{26.36}$	$2^{39.14}$	$2^{22.05}$	$2^{22.27}$	$2^{22.28}$	$2^{35.06}$
MLE	$2^{26.04}$	$2^{26.26}$	$2^{26.26}$	$2^{39.12}$	$2^{21.95}$	$2^{22.17}$	$2^{22.17}$	$2^{35.04}$
MAP	$2^{26.12}$	$2^{26.34}$	$2^{26.34}$	$2^{39.23}$	$2^{21.95}$	$2^{22.17}$	$2^{22.17}$	$2^{35.06}$
GF	$2^{28.22}$	$2^{28.44}$	$2^{28.45}$	$2^{41.13}$	$2^{23.22}$	$2^{23.44}$	$2^{23.45}$	$2^{36.13}$
GF-SEI	$2^{27.90}$	$2^{28.13}$	$2^{28.13}$	$2^{41.30}$	$2^{22.32}$	$2^{22.54}$	$2^{22.54}$	$2^{35.71}$
KT-MLE	$2^{25.89}$	$2^{26.11}$	$2^{26.11}$	$2^{39.08}$	$2^{21.80}$	$2^{22.02}$	$2^{22.03}$	$2^{35.00}$
CM-HW	$2^{25.81}$	$2^{26.03}$	$2^{26.03}$	$2^{38.98}$	$2^{21.72}$	$2^{21.94}$	$2^{21.95}$	$2^{34.89}$

4.5 Latency

Latency is the time consumption to recover the round tweakey in the software implementation. The latency of the attacking procedure is composed of the faults injections, the intermediate states' derivation and the key recovery. The experimental results show that the distinguishers SEI, HW, MLE, MAP, GF, GF-SEI, KT-MLE and CM-HW spent 62.4, 53.9, 50.0, 49.6, 119.6, 62.0, 59.3 and 60.4 ms in the SFA of QARMA-128, respectively (Fig. 8 (a)); and for QARMA-256, those distinguishers spend 12.69, 9.00, 17.53, 12.36, 9.94, 9.19, 12.27 and 8.83 s, respectively (Fig. 8 (b)). In summary, the novel distinguisher of CM-HW stands in the first place for the SFA of QARMA-256, and KT-MLE ranks the fourth for QARMA-128.

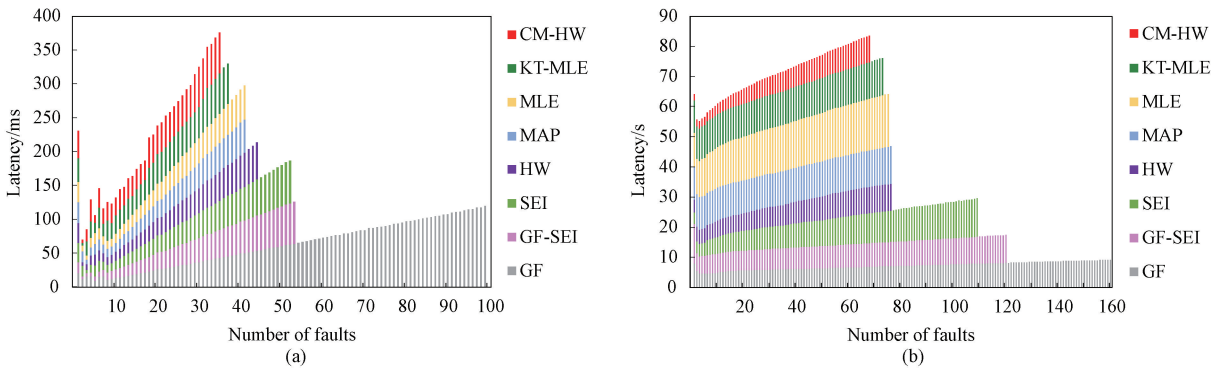

Fig. 8 Latency of recovering $3m$-bit of the round tweakey by the distinguishers for QARMA: (a) QARMA-128; (b) QARMA-256

5 Conclusions

This paper proposes the SFA of the lightweight tweakable block cipher QARMA with the novel distinguishers of CM-HW and KT-MLE. It only requires at least 374 and 726 faults to recover 128-bit and 256-bit secret keys of QARMA with a reliability of at least 99%, respectively. As for the different types of tweaks, the SFA takes the corresponding strategy to make fault injections into the deeper round of QARMA. In the practical IoE scenario, it is essential to take effective counter-measurements to protect the last three rounds of QARMA from statistical fault analysis.

References

[1] SNYDER T, BYRD G. The Internet of Everything[J]. *Computer*, 2017, 50(6): 8-9.

[2] DESAI S, ALHADAD R, CHILAMKURTI N, et al. A survey of privacy preserving schemes in IoE enabled smart grid advanced metering infrastructure[J]. *Cluster Computing*, 2019, 22(1): 43-69.

[3] LIU W, WATANABE Y, SHOJI Y. Vehicle-assisted data delivery in smart city: a deep learning approach[J]. *IEEE Transactions on Vehicular Technology*, 2020, 69(11): 13849-13860.

[4] CHINCHAWADE A J, LAMBA O S. Authentication schemes and security issues in Internet of Everything (IoE) systems[C]// 2020 12th International Conference on Computational Intelligence and Communication Networks, Bhimtal, India. New York: IEEE, 2020: 342-345.

[5] PENG P, SOLJANIN E. Covert, low-delay, coded message passing in mobile (IoT) networks

[J]. *IEEE Transactions on Information Forensics and Security*, 2022, 17: 599-611.
[6] NOSOUHI M R, SOOD K, GROBLER M, et al. Towards spoofing resistant next generation IoT networks [J]. *IEEE Transactions on Information Forensics and Security*, 2022, 17: 1669-1683.
[7] BOGDANOV A, KNUDSEN L R, LEANDER G, et al. PRESENT: an ultra-lightweight block cipher [C]// International Workshop on Cryptographic Hardware and Embedded Systems, Vienna, Austria. Berlin: Springer, 2007: 450-466.
[8] GUO J, PEYRIN T, POSCHMANN A, et al. The LED block cipher [C]//International Workshop on Cryptographic Hardware and Embedded Systems, Nara, Japan. Berlin: Springer, 2011: 326-341.
[9] WU W L, ZHANG L. LBlock: a lightweight block cipher [C]//International Conference on Applied Cryptography and Network Security, Nerja, Spain. Berlin: Springer, 2011: 327-344.
[10] LARA-NINO C A, DIAZ-PEREZ A, MORALES-SANDOVAL M. Lightweight hardware architectures for the present cipher in FPGA[J]. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2017, 64(9): 2544-2555.
[11] MOHD B J, HAYAJNEH T, AHMAD YOUSEF K M, et al. Hardware design and modeling of lightweight block ciphers for secure communications [J]. *Future Generation Computer Systems*, 2018, 83: 510-521.
[12] AVANZI R. The QARMA block cipher family: almost MDS matrices over rings with zero divisors, nearly symmetric Even-Mansour constructions with non-involutory central rounds, and search heuristics for low-latency S-boxes[J]. *IACR Transactions on Symmetric Cryptology*, 2017, 2017(1): 4-44.
[13] CHEN Y L. Pseudorandom permutations and functions for lightweight applications [D/OL]. Belgium: KU Leuven, 2022 [2023-04-25]. https://kuleuven. limo. libis. be/discovery/fulldisplay? docid = lirias3682687 & context = Search Webhook&vid = 32KUL _ KUL: Lirias & search _ scope = lirias _ profile&tab = LIRIAS & adaptor = SearchWebhook&lang = en.
[14] BORGHOFF J, CANTEAUT A, GÜNEYSU T, et al. PRINCE: a low-latency block cipher for pervasive computing applications [C]// International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China. Berlin: Springer, 2012: 208-225.
[15] BOURA C, CANTEAUT A, KNUDSEN L R, et al. Reflection ciphers [J]. *Designs, Codes and Cryptography*, 2017, 82(1): 3-25.
[16] LI R J, JIN C H. Meet-in-the-middle attacks on reduced-round QARMA-64/128 [J]. *The Computer Journal*, 2018, 61(8): 1158-1165.
[17] LI M Z, HU K, WANG M Q. Related-tweak statistical saturation cryptanalysis and its application on QARMA[J]. *IACR Transactions on Symmetric Cryptology*, 2019(1): 236-263.
[18] ZONG R, DONG X Y. MILP-aided related-tweak/key impossible differential attack and its applications to QARMA, Joltik-BC[J]. *IEEE Access*, 2019, 7: 153683-153693.
[19] LIU Y, ZANG T D, GU D W, et al. Improved cryptanalysis of reduced-version QARMA-64/128 [J]. *IEEE Access*, 2020, 8: 8361-8370.
[20] DU J, WANG W, LI M Z, et al. Related-tweakey impossible differential attack on QARMA-128 [J]. *Science China Information Sciences*, 2021, 65(2): 129102.
[21] BONEH D, DEMILLO R A, LIPTON R. On the importance of checking cryptographic protocols for faults [C]//International Conference on the Theory and Applications of Cryptographic Techniques, Konstanz, Germany. Berlin: Springer, 1997: 37-51.
[22] ZHANG F, ZHAO X J, GUO S Z, et al. Improved algebraic fault analysis: a case study on Piccolo and applications to other lightweight block ciphers [C]//International Workshop on Constructive Side-Channel Analysis and Secure Design, Paris, France. Berlin: Springer, 2013: 62-79.
[23] MORADI A, SHALMANI M T M, SALMASIZADEH M. A generalized method of differential fault attack against AES cryptosystem [C]//International Workshop on Cryptographic Hardware and Embedded Systems, Yokohama, Japan. Berlin: Springer, 2006: 91-100.
[24] BIHAM E, GRANBOULAN L, NGUYỄN P Q. Impossible fault analysis of RC_4 and differential fault analysis of RC_4 [C]//International Workshop on Fast Software Encryption, Paris, France. Berlin: Springer, 2005: 359-367.
[25] DERBEZ P, FOUQUE P-A, LERESTEUX D. Meet-in-the-middle and impossible differential fault analysis on AES [C]//International Workshop on Cryptographic Hardware and Embedded Systems, Nara, Japan. Berlin: Springer, 2011: 274-291.
[26] FUHR T, JAULMES E, LOMNÉ V, et al. Fault attacks on AES with faulty ciphertexts only[C]// Workshop on Fault Diagnosis and Tolerance in Cryptography, Los Alamitos, USA. New York: IEEE, 2013: 108-118.
[27] DOBRAUNIG C, EICHLSEDER M, KORAK T, et al. Statistical fault attacks on nonce-based

authenticated encryption schemes [C]// International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam. Berlin: Springer, 2016: 369-395.
[28] LI W, LIAO L F, GU D W, et al. Ciphertext-only fault analysis on the LED lightweight cryptosystem in the Internet of Things[J]. *IEEE Transactions on Dependable and Secure Computing*, 2019, 16(3): 454-461.
[29] LI W, LI J Y, GU D W, et al. Statistical fault analysis of the Simeck lightweight cipher in the ubiquitous sensor networks [J]. *IEEE Transactions on Information Forensics and Security*, 2021, 16: 4224-4233.
[30] REED I. A class of multiple-error-correcting codes and the decoding scheme[J]. *Transactions of the IRE Professional Group on Information Theory*, 1954, 4(4): 38-49.
[31] WILKS S S. The large-sample distribution of the likelihood ratio for testing composite hypotheses [J]. *The Annals of Mathematical Statistics*, 1938, 9(1): 60-62.

万物网中轻量级可调分组密码 QARMA 的统计故障分析

李嘉耀, 李　玮*, 高建宁, 秦梦洋, 孙文倩
东华大学 计算机科学与技术学院, 上海 201620

摘　要: 基于唯密文攻击（ciphertext-only attack, COA）假设, 提出了能够破译万物网（Internet of Everything, IoE）中 QARMA 密码算法所有版本的统计故障分析（statistical fault analysis, SFA）。针对调柄的不确定性, 利用多种分析策略有助于将故障注入更深的轮数。为了提高分析效率, 提出了两种新型区分器: 克米试验—汉明重量区分器（Cramér-von Mises test-Hamming weight, CM-HW）和柯伊伯检验—极大似然估计（Kuiper's test-maximum likelihood estimation, KT-MLE）区分器。试验结果表明, 攻击者仅需将 374 个或者 726 个随机故障分别注入到两个版本的 QARMA 密码的倒数第三轮, 即可以 99%的可靠度恢复其 128 比特或者 256 比特子密钥。综上所述, 在万物网的应用环境中, QARMA 容易受到统计故障分析的影响。研究结果可为具有反射结构的轻量级可调分组密码和密码设备的保护提供参考。

关键词: 万物网（IoE）; 侧信道分析; 轻量级可调分组密码; 统计故障分析（SFA）; QARMA