Using Tweaks To Design Fault Resistant Ciphers

Sikhar Patranabis, Debapriya Basu Roy and Debdeep Mukhopadhyay

Department of Computer Science and Engg.
IIT Kharagpur, India
{sikhar.patranabis, deb.basu.roy, debdeep}@cse.iitkgp.ernet.in

Abstract. Side channel analysis and active fault analysis are now major threats to even mathematically robust cryptographic algorithms that are otherwise resistant to classical cryptanalysis. It is necessary to design suitable countermeasures to protect cryptographic primitives against such attacks. This paper focuses on designing encryption schemes that are innately secure against fault analysis. The paper formally proves that one such design strategy, namely the use of key-dependent SBoxes, is only partially secure against DFA. The paper then examines the fault tolerance of encryption schemes that use a key-independent secret tweak value for randomization. In particular, the paper focuses on a linear tweak based and a non-linear tweak based version of a recently proposed block cipher DRECON. The paper demonstrates that while both versions are secure against classical DFA, the non-linear tweak based version provides greater fault coverage against stronger fault models. This fact, together with the DPA resistance provided by the use of variable S-Boxes, makes DRECON a strong candidate for the design of secure cryptographic primitives. All claims have been validated by experimental results on a SASEBO GII platform.

Keywords. Fault Attacks, Countermeasures, SBox, DFA, DPA, DRECON, Secret Tweaks, Security

1 Introduction

In recent times, side channel analysis (SCA) attacks such as the Differential Power Analysis (DPA) and the Correlation Power Analysis (CPA), as well as active fault analysis (FA) attacks such as Differential Fault Analysis (DFA) and Differential Fault Intensity Analysis (DFIA) on cryptographic devices have raised serious security issues. Several mathematically robust and classically secure block ciphers such as AES have been rendered vulnerable by SCA and FA owing to leakages from insecure implementations. FA involves injection of faults into cryptographic systems and analysis under different fault models to retrieve the secret key. For instance, DFA of block ciphers such as AES [1] exploits the relation between faulty and fault-free ciphertext pairs, and may require as few as a single fault injection to recover the entire key [2]. DFIA [3] on the other hand combines the principles of DPA with FA, and uses only faulty ciphertexts to recover the key.

With side channel analysis and fault analysis being established threats to the security of cryptosystems, sound countermeasures are needed to protect them. Traditional countermeasures against side channel analysis such as DPA include masking [4], shuffling [5] and hiding [6]. More recent approaches include the use of variable S-Boxes, such as rotational S-Boxes [7], chaotic S-Boxes [8] and tweaked S-Boxes [9]. Fault attacks such as DFA too have been countered to a certain degree of success. Classical countermeasures against DFA such as concurrent error detection (CED) techniques [10–13] use various forms of redundancy to detect faults. They, however, work under the assumption of uniform fault models and could be weakened by biased fault attacks [14, 15]. More recently proposed infective countermeasures [16] attempt to diffuse the effect of the fault using suitable randomizations. Although a wide variety of such countermeasures have been proposed, very few of them provide combined security against both side channel and fault attacks. A very recent proposition based on orthogonal direct sum masking [17] attempts to counter monovariate DPA while also detecting faults, albeit of limited Hamming weight. Thus designing countermeasures that are inherently resistant to both side channel and fault attacks is a problem of great practical interest. In this paper, we attempt to address this issue by exploring the use of tweaked block ciphers. The use of a secret tweak along with a key introduces a certain degree of randomness in the block cipher, that could be used to achieve resistance against a variety of attacks such as DPA and DFA.

Contributions: The main contributions of the paper are as follows. The paper establishes via information theoretic arguments that ciphers that use a part of the key itself as the tweak are inherently vulnerable to DFA. The paper then demonstrates that when the tweak is independent of the key, as in DRECON, it may be combined with the cipher state both linearly and non-linearly to achieve DFA resistance. Finally the paper argues that DRECON, by virtue of its non-linear use of the tweak, is a strong candidate for designing cryptosystems secure against both DPA as well as DFA.

Organization: The rest of the paper is organized as follows. Section 2 introduces the concept of tweakable ciphers. Section 3 examines the security of block ciphers that use the key itself instead of an independent tweak element to randomize the encryption. Sections 4 and 5 investigate the security against DFA of linearly and non-linearly tweakable versions of DRECON. Experimental results on a SASEBO GII board establishing DRECON as an effective countermeasure against both DPA and DFA, have been presented in Section 6. Finally, Section 7 concludes the paper.

2 Prelims: Tweakable Ciphers

In cryptographic literature, the concept of tweaks was first introduced in the design of the block cipher *Hasty Pudding* [18]. This cipher was proposed in the AES competition organized by the NIST, and required an extra input that was completely independent of the plaintext and the master key. This extra input is now formally recognized as the *tweak*. Tweakable block ciphers like *MERCY* [19] are used principally for disk encryption, where the block id/index is used as the tweak. In general, for an efficient tweakable block cipher, the cost of tweak scheduling and tweak refreshing should be less than that of changing the key [20]. Moreover, most tweakable block ciphers assume that the tweak is completely public and may even be controlled by the adversary [20].

Table 1: Notations Used

X	A discrete random variable
x_i	A specific value that X may take
$Pr(X = x)$ or $Pr(x)$	The probability that a random variable X takes a value x
$Pr(x\|y)$	The conditional probability that $X = x$ given $Y = y$
$H(X)$	The entropy of random variable X
$H(X\|Y)$	The conditional entropy of X given Y
$I(X\|Y)$	The mutual information of random variables X and Y
K	The secret key used by AES
$\varDelta_{in}$	The input differential of an S-Box
$\varDelta_{out}$	The output differential of an S-Box
$\varDelta = (\varDelta_{in}, \varDelta_{out})$	The pair of input-output differential of an S-Box
N	The total number of possible values for $K, \varDelta_{in}, \varDelta_{out}$
$\{k_1, k_2, \cdots, k_N\}$	The sample space from which K takes its values
$\{\varDelta_1, \varDelta_2, \cdots, \varDelta_{N^2}\}$	Sample space from which $\varDelta$ can take its value

The first use of tweaks to achieve side channel resistance was made in the design of the block cipher DRECON [9]. DRECON is inherently secure against DPA by construction - the source of this security being the incorporation of tweaks in the cipher design, where the tweak is secret and is refreshed for every encryption. The objective is to randomize the S-Box usage of the cipher based on the tweak value, such that a side channel adversary can not build any hypothetical power model to execute a first order correlation power attack. Hence, to ensure side channel security, the tweak value in case of DRECON must be *secret*, which is a major difference between DRECON and other tweakable block ciphers. In this paper, we demonstrate that the secret use of the tweak also makes DRECON resistant to a wide variety of fault attacks, ranging from traditional DFA to even strong multi-fault injection attacks.

3 Ciphers without Key-Independent Tweaks

In this section we look at strategies for designing ciphers that do not use additional key-independent tweaks but inherently possess a certain degree of randomness. Our aim is to evaluate their resistance to DFA. In classical DFA, the adversary introduces faults into the state of the cipher in a target round to obtain one or more faulty ciphertexts. The pair(s) of fault free and faulty ciphertexts are then used to recover the key using a system of equations. The adversary uses a fault model of her choice to introduce the fault and then uses the output differential to formulate the equations that help reduce the key search space. Thus intuitively, if a cipher is to be inherently resistant to DFA, there must exist some kind of randomization in the encryption algorithm that would enhance the confusion and would prevent the adversary from being able to appreciably reduce the key search space in spite of formulating the same equations. One such strategy that we focus on here is the use of key-dependent S-Boxes in the encryption algorithm. Table 1 summarizes some of the notations used in this section.

An example of a cipher that uses key-dependent S-Boxes is the Twofish cipher [21], which was among the finalists of the AES contest. Twofish has a Feistel structure similar to the Data Encryption Standard (DES) [22] and uses the concept of key-dependent S-boxes and Pseudo-Hadamard Transform, which make the cipher secure against traditional differential attacks. It uses a 128 bit state and has key sizes up to 256 bits. One half of the key is used for the actual encryption, while the other half is used for choosing the S-Box to be used. Recently, [23] has proposed a DFA of Twofish that uses an approximation to the modulo 2^{32} addition operation and guesses the input-output differential of the S-Box from the faulty and fault-free ciphertexts. The attack exploits the fact that in Twofish, the choice of S-Box given an input output differential pair is non-uniform. Since each choice of S-Box involves 2 bytes of the key, the probability distribution of the key given an input-output differential is also non-uniform. The attack retrieves the secret key using 320 pairs of fault-free and faulty ciphertexts and has an attack complexity of 2^{40}. We next present an information theoretic analysis to demonstrate that in such a case the mutual information of the input-output differential pair $\varDelta = (\varDelta_{in}, \varDelta_{out})$ and the key K is non-zero.

Let us assume that K, $\varDelta_{in}$ and $\varDelta_{out}$ can take N values each. Thus $\varDelta$ can take N^2 values. The required mutual information expression is given by $I(K; \varDelta) = H(K) - H(K \mid \varDelta)$, where

$$H(K) = -\sum_{i=1}^{N} Pr(k_i) \log \left(Pr(k_i)\right)$$

$$H(K \mid \varDelta) = -\sum_{i=1}^{N} \sum_{j=1}^{N^2} Pr(\varDelta_j) Pr(k_i \mid \varDelta_j) \log \left(Pr(k_i \mid \varDelta_j)\right)$$

Since the key K takes N values each of which are equally likely, $H(K) = \log N$. Jensen's inequality [24] states that given a concave function $\varphi(x)$,numbers $x_1, x_2, \cdots, x_n$ in its domain, and positive weights a_i,

$$\varphi\left(\frac{\sum a_i x_i}{\sum a_i}\right) \geq \frac{\sum a_i \varphi(x_i)}{\sum a_i} \tag{1}$$

Equality holds if and only if $x_1 = x_2 = \cdots = x_n$ or φ is linear. Now, as $\log(1/x)$ is a non-linear concave function and $Pr(x)$ is a positive quantity $\forall x$, we may use Jensen's inequality to state the following.

$$\begin{aligned} H(K \mid \varDelta) &= \sum_{i=1}^{N} \sum_{j=1}^{N^2} Pr(\varDelta_j) Pr(k_i \mid \varDelta_j) \log(1/Pr(k_i \mid \varDelta_j)) \\ &\leq \sum_{j=1}^{N^2} Pr(\varDelta_j) \log(\sum_{i=1}^{N} Pr(k_i \mid \varDelta_j)/Pr(k_i \mid \varDelta_j)) \\ &\leq \log N \end{aligned} \tag{2}$$

where equality is achieved only for the scenario where for a given input-output differential pair $\varDelta_j$, $Pr(k_i \mid \varDelta_j) = 1/N \; \forall i \in \{1, 2, \cdots, N\}$. So, for a given input-output differential pair, if the probability distribution of keys is non-uniform, $H(K|\varDelta) < \log N$ and $I(K; \varDelta) > 0$. This non-uniformity stems from the fact that the S-Boxes used in Twofish are key dependent, and is hence responsible for information leakage.

There have been propositions of schemes that modify AES to use key-dependent S-Boxes, such as rotational S-Boxes [7, 25], chaotic S-Boxes[8, 26], as well as reduction and switch S-Boxes [27]. Of these, reduction and switch boxes are inherently vulnerable

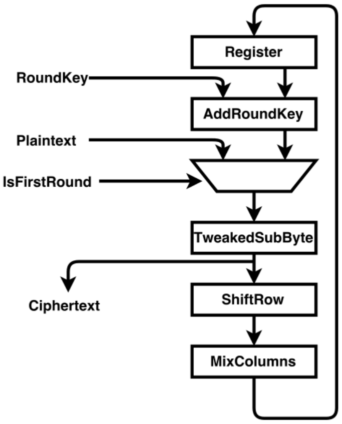

Fig. 1: A Schematic of Linear DRECON

to DFA due to their small S-Box spaces. Extended DFA strategies have also been proposed against rotational and chaotic S-Boxes [27]. Thus ciphers that use key-dependent S-Boxes seem to be inherently vulnerable to DFA.

4 Linearly Tweakable Block Ciphers

In this section, we look at ciphers that use a key-independent randomizing element as the tweak, and combine it linearly with the cipher state. A straightforward approach could be to perform a one-time XOR of the tweak with either the plaintext input P or the key input K. This would however amount to weakening the tweakable encryption function $E_K(\tau, P)$, where K, τ and P are the key, tweak and plaintext inputs respectively. This is easy to see, since scenarios such as $E_K(\tau, P) = E_K(\tau \oplus X, P \oplus X)$ and $E_K(\tau, P) = E_{K \oplus X}(\tau \oplus X, P)$ may occur in each case respectively [20]. The tweak therefore must be combined with either the cipher state or the key more often than just once at the beginning of the encryption. We use this very idea to build an alternative version of DRECON called Linear DRECON which uses a tweak that is combined linearly with the state of the cipher after every encryption round. Figure 1 describes the order of various operations take place for Linear DRECON. As in DRECON, the number of rounds in Linear DRECON is 20. The first 19 rounds have the following five operations in order - SubByte, AddRoundTweak, ShiftRow, MixColumns and AddRoundKey. The final round has only the SubByte followed by an AddRoundTweak. For details of the round operations, refer [9]. All operations are *nibble oriented*. Note that, as in DRECON, the tweak is secret for Linear DRECON also. The major differences of the Linear DRECON scheme with the original DRECON scheme proposed in [9] are :

1. Linear DRECON has a normal SubByte operation that uses a single S-Box for all nibbles for all the rounds, unlike original DRECON which has a TweakedSubByte

Table 2: Notations Used

S	An S-Box
P	Plaintext
$P_{i,j}$	The nibble of P with index (i,j)
C	Correct ciphertext
C'	Faulty Ciphertext
f	A bit fault
$C_{i,j}$	The nibble of C with index (i,j)
τ	The 128 bit secret tweak
$\tau_{i,j}$	The nibble of τ with index (i,j)
s	An intermediate cipher state
ω	A random nibble value
$s_{i,j}$	The nibble of s with index (i,j)
K^r	The key for round r
$K^r_{i,j}$	The nibble of K^r with index (i,j)

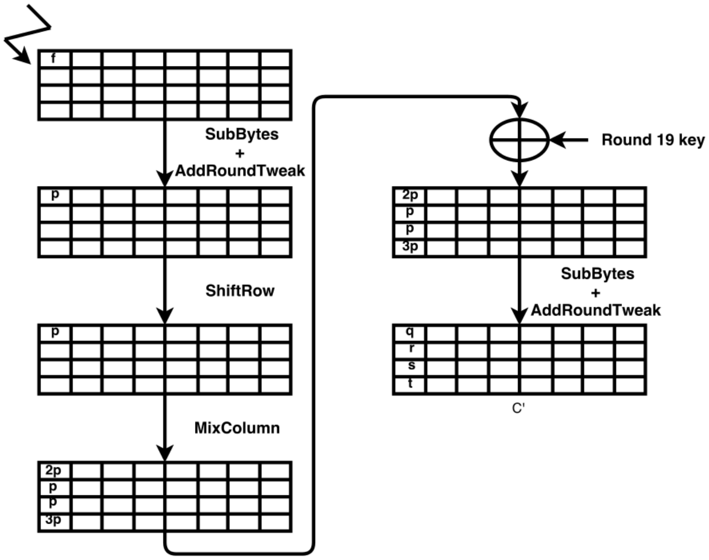

Fig. 2: DFA of Linear DRECON : Bit Fault on the Penultimate Round

operation where the S-Box for a given round and for a given nibble is chosen from a set of 16 S-Boxes based on the tweak value for that round.

2. Linear DRECON has an additional AddRoundTweak operation in every round where the 128 bit round tweak is XOR-ed with the cipher state. In DRECON, the tweak is used in the TweakedSubByte operation to choose the S-Box separately for each nibble in any given round.

4.1 Security of Linear DRECON against Bit-level DFA of the penultimate round

We now evaluate the security of this scheme against DFA. Table 2 summarizes some of the notations used in the forthcoming discussion. Figure 2 explains the attack procedure pictorially.

The Fault Model The fault model for this attack is a *bit fault model* in which the adversary upsets exactly one bit of a target nibble the cipher state just prior to the penultimate SubByte operation. The idea is that since only 4 possible faults are possible under this fault model, the search space of the corresponding key nibble could be reduced by fault injection.

The Attack Strategy The adversary obtains a fault free and faulty ciphertext C and C' corresponding to the same plaintext for the same key. Let the value of the target state where the fault was injected be s and s' for the first and second instance respectively, and let τ and τ' be the tweak values for the two instances, with $\tau \neq \tau'$. The following equations hold for a bit fault fault f injected in the penultimate round.

$$2p = S^{-1}(C_{0,0} \oplus \tau_{0,0}) \oplus S^{-1}({C'}_{0,0} \oplus \tau'_{0,0}) \oplus \omega \tag{3}$$

$$S(s_{0,0}) \oplus S({s'}_{0,0} \oplus f) \oplus \tau_{0,0} \oplus {\tau'}_{0,0} \oplus \omega' = p \tag{4}$$

$$S(K^{19}_{0,0} \oplus 2S(s_{0,0}) \oplus 3S(s_{1,1}) \oplus S(s_{2,3}) \oplus S(s_{3,4})) \oplus \tau_{0,0} = C_{0,0} \tag{5}$$

The nibble constants ω and ω' take into account the fact that the states of the cipher in the fault free and faulty computation instances are already different at the beginning of the final round and penultimate round respectively, due to AddRoundTweak operations in previous rounds. The adversary guesses the values of f, $\tau_{0,0}$, $\tau'_{0,0}$, ω, ω' and ${s'}_{0,0}$ to obtain a guess for the value of $s_{0,0}$. The other nibbles $s_{1,1}, s_{2,3}, s_{3,4}$ may be similarly guessed. Next, the value of $K^{19}_{0,0}$ is guessed from Equation 5. The adversary performs multiple fault injections till the search space is reduced to 1 and the nibble is uniquely identified. However, we claim that the adversary cannot reduce the size of the search space for the key nibble. On substituting $f' = s_{0,0} \oplus {s'}_{0,0} \oplus f$ and $p' = p \oplus \tau_{0,0} \oplus \tau'_{0,0} \oplus \omega'$ Equation 4 takes the form $S(s_{0,0}) \oplus S(s_{0,0} \oplus f') = p'$. Note that the adversary has to guess the value of p' as well as f to build the search space for $s_{0,0}$. Note that p' can take 16 possible values. Let us assume that the adversary guesses a particular value of p', say p'_1. Given that the output differential p'_1, for each guessed value of f, there are 16 possible values of f'. Each such value of f' gives rise to a solution of $s_{0,0}$ on an average. Hence the search space for $s_{0,0}$ is of size 16. This implies that the search space for each key nibble is also of size 16, which is essentially the brute force search space. Thus the DFA is not able to achieve any reduction in key search space and the attack fails.

Similar analysis could be done for other fault models such as nibble fault models where the fault is introduced in an earlier round and similar equations are formulated to try and obtain the key. But in all such scenarios, the presence of the AddRoundTweak operation prevents the adversary from formulating a set of equations that could be used to reduce the key search space in any way. Thus Linear DRECON is secure against classical DFA.

5 Non-Linear Use of Tweaks in Block Ciphers

In this section, we look at the security of block ciphers that combine the tweak value with the cipher state in a non-linear fashion. As an example, we choose DRECON as

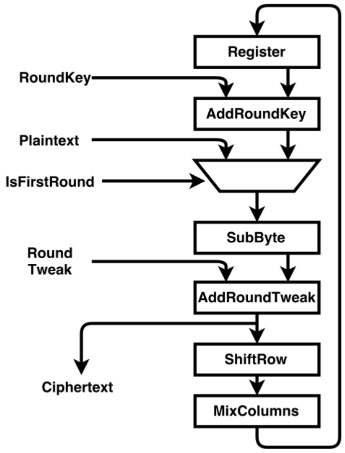

Fig. 3: A Schematic of DRECON

proposed in [9]. Figure 3 presents a schematic description for the DRECON cipher. We use the same single bit fault model as was used in the earlier section to evaluate the security of Linear DRECON. *Please note that in the forthcoming discussion $S^r_{i,j}$ refers to the S-Box used in round r for nibble with index (i, j).*

5.1 Security of DRECON against Bit-level DFA of the penultimate round

The attack strategy remains the same. Since the tweak is now manifested in the choice of S-Box, the equations 3 and 4 now takes the following form.

$$2p = S_{0,0}^{20\;-1}(C_{0,0}) \oplus S'^{20\,-1}_{0,0}(C'_{0,0}) \oplus \omega \tag{6}$$

$$S^{19}_{0,0}(s_{0,0}) \oplus S'^{19}_{0,0}(s'_{0,0} \oplus f) \oplus \omega' = p \tag{7}$$

Here ω and ω' again take into account the fact that the states of the cipher in the fault free and faulty computations are already different at the beginning of the final and penultimate rounds due to previous tweaked SubBytes operations. The adversary thus needs to guess the S-Box choices $S^{20}_{0,0}$, $S'^{20}_{0,0}$ as well as the value of the nibble ω to obtain p. Finally, after all the state nibbles have been uniquely identified, the nibbles for K^{19} may be obtained using relations similar to equation 5 for Linear DRECON. Our claim is that neither of these equations 6 and 7 can reduce the search space for $s_{0,0}$ due to the use of the tweaked S-Boxes in DRECON. We present a generalized analysis using the properties of S-Boxes that establishes this claim. Let S_1 and S_2 be two randomly chosen 4×4 S-Boxes from the set of S-Boxes for DRECON. Consider the equation $S_1(A) \oplus S_2(B) = \beta$, where A and B are independent of each other and have uniform unbiased distributions. This assumption applies in case of DRECON since the use of tweaked S-Boxes in each round implies that the cipher states in different instances are completely independent of each other. Since both S_1 and S_2 are bijective mappings, the equation is satisfied by some B for each value of A and vice versa. We now compute the mutual information of A and the output differential β. Let A takes values from the set $\{a_1, a_2, \cdots, a_{16}\}$ and β takes values from the set $\{\beta_1, \beta_2, \cdots, \beta_{16}\}$. We have $H(A) = 4$ as each value of A is equally likely. For each value of A we have a corresponding value of B such that $S_1(A) \oplus S_2(B) = \beta$ for any pair of S-Box choices (S_1, S_2) as the S-boxes are bijective functions. The conditional probability $Pr(a_i|\beta_j)$ is thus $\frac{1}{16}$ for all a_i and β_j. Thus, we have

$$\begin{aligned} H(A|\beta) &= -\sum_{j=1}^{16}\sum_{i=1}^{16} Pr(\beta_i)Pr(a_j \mid \beta_i)\log Pr(a_j \mid \beta_i) \\ &= -\sum_{j=1}^{16}\sum_{i=1}^{16} Pr(\beta_i)(1/16)\log{(1/16)} \\ &= 4 \end{aligned} \tag{8}$$

Finally, from the definition of mutual information of two random variables, we get $I(A;\beta) = 0$. One can similarly show that $I(B;\beta) = 0$. Thus the knowledge of the output differential β does not restrict the search space for either A or B. Clearly, this argument can be applied to the set of equations obtained from the DFA attack strategy as well. Despite the fact that the fault f is a bit fault meaning *it does not have a uniform unbiased distribution*, the random variables τ and τ' which result from the use of tweaked S-Boxes in all the previous rounds mask the effect of the fault f. Thus, as in case of Linear DRECON, the DFA fails in this case as well.

5.2 Comparison of the Two Tweak-Based Approaches

From our previous analysis, it is immediately apparent that both the linear and non-linear versions of DRECON are secure against traditional DFA. A natural question to ask is which one to adopt when designing a block cipher. One argument in favor of the original DRECON is that it also provides security against first-order DPA, which Linear DRECON would fail to provide. We provide an additional argument here by demonstrating that Linear DRECON could be vulnerable to some attacks using strong fault models that DRECON provides security against. In the forthcoming discussion, we present a security analysis of both schemes against a strong multi-fault injection attack. *Please note that we are not claiming that this attack model is a very practical or efficient one*. It has been used just to demonstrate the high fault coverage that DRECON provides by virtue of its non-linear use of tweaks.

In this attack, the adversary injects *two faults* in the first round of the encryption algorithm. The first fault is a stuck-at-zero fault and is introduced in a target nibble of the tweak τ. The second fault involves injecting a random fault q in the round counter at the end of the first round. The aim is to try and set the round counter value to greater than 20 so that the algorithm terminates and the intermediate output of the first round is obtained as a faulty ciphertext. The adversary uses a known plaintext attack to recover the key.

We present here the strategy to obtain the key nibble $K^1_(0,0)$ for Linear DRECON. The stuck-at-zero fault is introduced in the first round tweak nibble $\tau_{0,0}$. Assuming that the plaintext is P and the ciphertext resulting from a successful injection of the aforementioned faults into the cryptosystem is C'', the following equation holds true.

$$2S(P_{0,0}) \oplus 3S(P_{1,1}) \oplus S(P_{2,3}) \oplus S(P_{3,4}) \oplus K^1_{0,0} = C''_{0,0} \tag{9}$$

Since $\tau_{0,0} = 0$ it does not feature in the equation. Thus $K^1_{0,0}$ is the only unknown in the equation and is easily obtained. By introducing similar faults in all other tweak nibbles followed by attacks on the round counter, the adversary can obtain every nibble of the first round key K_1. Next the same attack is mounted on the original DRECON. Assuming that the plaintext is still P and the ciphertext resulting from a successful injection of the aforementioned faults into the cryptosystem is C''', equation 9 changes as follows

$$2S_0(P_{0,0}) \oplus 3S_1(P_{1,1}) \oplus S_2(P_{2,3}) \oplus S_3(P_{3,4}) \oplus K^1_{0,0} = C'''_{0,0} \tag{10}$$

Note that, unlike in equation 9, multiple plaintext nibbles are involved in the equation, and the S-Box for each nibble is different. Non-linear tweaking successfully combines 4 tweak nibbles into the computation of a single key nibble, unlike the linear tweaking which has a one-to-one key nibble to tweak nibble dependency. We now focus on retrieving the value of $K^1_{0,0}$. Since $\tau_{0,0} = 0$, it is logical to assume that S_0 is fixed and is known to the adversary. However, unlike in Linear DRECON, here, $K^1_{0,0}$ is not the only unknown in the equation. The other unknowns are the three S-Boxes S_1, S_2 and S_3 which must be guessed by the adversary. Again, since the S-Boxes are bijective functions, each possible value of the key nibble gives rise to at least one potential solution, and the search space for $K^1_{0,0}$ is still 2^4. Moreover, independent fault injections into other state nibbles would give similar independent search spaces for the corresponding key nibbles. Thus, the overall key search space for all key nibbles is $2^{4\times 32} = 2^{128}$, which is essentially the brute force search space.

A closer analysis of the attack strategy reveals that the main reason why this attack works on Linear DRECON and not on DRECON is the difference in the way the two systems use the secret tweak. Thus, even for an adversarial model as strong as a multi fault injection with high precision, non-linear tweaking provides fault tolerance. This, together with the DPA resistance, makes DRECON a strong candidate for secure cryptographic implementations.

6 Experimental Results

In order to provide a perspective of the effectiveness of DRECON in thwarting both DPA and DFA, we present the success rate of the correlation-based DPA attack vs the number of traces (using attack hypothesis of 4 bits only) for DRECON at various number of tweak values $r \in \{1, 2, 4, 8, 16\}$ where $r = 1$ is the deterministic case of no tweak and $r = 16$ is the case of 4 bit tweak, and compare it with other information redundancy countermeasures. The experiments have been carried out on a SASEBO GII board. The adversary makes a hypothesis of 4 key bits at a time and is unaware of

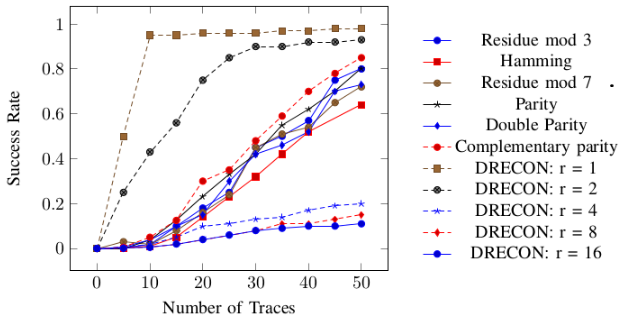

Fig. 4: Success rate of the correlation-based DPA attack vs the number of traces using attack hypothesis of 4 bits of key

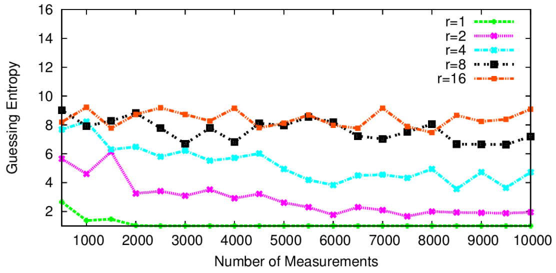

Fig. 5: Guessing Entropy vs number of measurements for different size of tweaks

the use of error detection/correction code or tweak value used. Figure 4 summarizes the results. Quite evidently, DRECON is much more resistant against DPA as compared to other information redundancy techniques for number of tweak values $r = 8$ and $r = 16$. Figure 5 presents the guessing entropy metric for different tweak sizes. When the tweak is not present ($r = 1$), the guessing entropy decreases as the number of measurements increases, and with 10000 traces the guessing entropy becomes about 5. This is significantly below the guessing entropy of random guess. When $r = 4$ (the size of the input to the S-Box), the guessing entropy is around 8 all the time, implying that the scheme is as secure as a random guess of the key.

7 Conclusions

This paper analyzes in depth strategies to design ciphers that are innately secure against fault analysis. The paper looks at two distinct classes of randomization techniques that could be incorporated into block ciphers to achieve security against differential fault analysis (DFA). The first class of ciphers uses a part of the key itself instead of an independent tweak element to randomize the encryption operations. The paper provides information theoretic arguments to establish that this class of ciphers is vulnerable to DFA due to the non-uniform probability distribution of S-Boxes for a given input-output differential pair. This non-uniformity leaks information about the key. The paper then examines two versions of a recently proposed tweakable DPA resistant cipher - DRECON. The first version uses a separate key independent tweak but combines the tweak linearly with the cipher state, while the second combines the tweak with the state in a non-linear fashion by using it to choose the S-Box. The paper demonstrates that while both versions are secure against traditional DFA, the latter provides a wider coverage of strong fault models involving high precision multiple fault injections. Experimental results on a SASEBO GII platform have been presented to demonstrate while most state of the art countermeasure schemes seem to effectively counter either DPA or DFA, DRECON provides resistance against both forms of attacks, and can thus be used as a unified countermeasure to design electronic circuits that are secured against both power as well as fault attacks.

References

1. Christophe Giraud. DFA on AES. In Hans Dobbertin, Vincent Rijmen, and Aleksandra Sowa, editors, *Advanced Encryption Standard – AES*, volume 3373 of *Lecture Notes in Computer Science*, pages 27–41. Springer, 2005.
2. Michael Tunstall, Debdeep Mukhopadhyay, and Subidh Ali. Differential fault analysis of the advanced encryption standard using a single fault. In *Information Security Theory and Practice. Security and Privacy of Mobile Devices in Wireless Communication*, pages 224–233. Springer, 2011.
3. Nahid Ghalaty, Bilgiday Yuce, Mostafa Taha, and patrick Schaumont. Differential Fault Intensity Analysis. *2014 Workshop on Fault Diagnosis and Tolerance in Cryptography(FDTC).IEEE*, 2014.
4. Elisabeth Oswald, Stefan Mangard, Norbert Pramstaller, and Vincent Rijmen. A side-channel analysis resistant description of the aes s-box. In *Fast Software Encryption*, pages 413–423. Springer, 2005.
5. Nicolas Veyrat-Charvillon, Marcel Medwed, Stéphanie Kerckhof, and François-Xavier Standaert. Shuffling against side-channel attacks: A comprehensive study with cautionary note. In *Advances in Cryptology–ASIACRYPT 2012*, pages 740–757. Springer, 2012.
6. Robert P McEvoy, Colin C Murphy, William P Marnane, and Michael Tunstall. Isolated wddl: a hiding countermeasure for differential power analysis on fpgas. *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, 2(1):3, 2009.
7. GN Krishnamurthy and V Ramaswamy. Making aes stronger: Aes with key dependent s-box. *IJCSNS International Journal of Computer Science and Network Security*, 8(9):388–398, 2008.
8. Guoping Tang, Xiaofeng Liao, and Yong Chen. A novel method for designing s-boxes based on chaotic maps. *Chaos, Solitons & Fractals*, 23(2):413–419, 2005.
9. Suvadeep Hajra, Chester Rebeiro, Shivam Bhasin, Gaurav Bajaj, Sahil Sharma, Sylvain Guilley, and Debdeep Mukhopadhyay. Drecon: Dpa resistant encryption by construction. In *Progress in Cryptology–AFRICACRYPT 2014*, pages 420–439. Springer, 2014.
10. Guido Bertoni, Luca Breveglieri, Israel Koren, Paolo Maistri, and Vincenzo Piuri. Error analysis and detection procedures for a hardware implementation of the advanced encryption standard. *Computers, IEEE Transactions on*, 52(4):492–505, 2003.
11. P. Maistri and R Leveugle. Double-Data-Rate Computation as a Countermeasure against Fault Analysis. *IEEE Transactions on Computers*, 57(11):1528–1539, 2008.

12. T Malkin, F.X Standaert, and M Yung. A Comparative Cost/Security Analysis of Fault Attack Countermeasures. *2005 Workshop on Fault Diagnosis and Tolerance in Cryptography(FDTC),IEEE*, pages 109–123, 2005.
13. Xiaofei Guo and Ramesh Karri. Recomputing with permuted operands: A concurrent error detection approach. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 32(10):1595–1608, 2013.
14. Sikhar Patranabis, Abhishek Chakraborty, Phuong Ha Nguyen, and Debdeep Mukhopadhyay. A biased fault attack on the time redundancy countermeasure for AES. In *Constructive Side-Channel Analysis and Secure Design - 6th International Workshop, COSADE 2015, Berlin, Germany, April 13-14, 2015. Revised Selected Papers*, pages 189–203, 2015.
15. Xiaofei Guo, Debdeep Mukhopadhyay, Chenglu Jin, and Ramesh Karri. Security analysis of concurrent error detection against differential fault analysis. *Journal of Cryptographic Engineering*, pages 1–17, 2014.
16. Harshal Tupsamudre, Shikha Bisht, and Debdeep Mukhopadhyay. Destroying fault invariant with randomization. In *Cryptographic Hardware and Embedded Systems–CHES 2014*, pages 93–111. Springer, 2014.
17. Julien Bringer, Claude Carlet, Hervé Chabanne, Sylvain Guilley, and Houssem Maghrebi. Orthogonal direct sum masking. In *Information Security Theory and Practice. Securing the Internet of Things*, pages 40–56. Springer, 2014.
18. R. Schroeppel. The hasty pudding cipher, 1998.
19. Paul Crowley. Mercy: A fast large block cipher for disk sector encryption. In *Fast Software Encryption*, pages 49–63. Springer, 2001.
20. JÃl'rÃl'my Jean, Ivica NikoliÄĞ, and Thomas Peyrin. Tweaks and keys for block ciphers: the tweakey framework. Cryptology ePrint Archive, Report 2014/831, 2014. `http://eprint.iacr.org/`.
21. Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, and Niels Ferguson. Twofish: A 128-bit block cipher. *NIST AES Proposal*, 15, 1998.
22. Data Encryption Standard. Data encryption standard. *Federal Information Processing Standards Publication*, 1999.
23. Sk Subidh Ali and Debdeep Mukhopadhyay. Differential fault analysis of twofish. In *Information Security and Cryptology*, pages 10–28. Springer, 2013.
24. Marek Kuczma. *An introduction to the theory of functional equations and inequalities: Cauchy's equation and Jensen's inequality*. Springer Science & Business Media, 2009.
25. Kazys Kazlauskas and Jaunius Kazlauskas. Key-dependent s-box generation in aes block cipher system. *Informatica*, 20(1):23–34, 2009.
26. Muhammad Asim and Varun Jeoti. Efficient and simple method for designing chaotic s-boxes. *ETRI journal*, 30(1):170–172, 2008.
27. Bradley M Flamm. Extending differential fault analysis to dynamic s-box advanced encryption standard implementations. Technical report, DTIC Document, 2014.