PICARO – A Block Cipher Allowing Efficient Higher-Order Side-Channel Resistance

Gilles Piret[1], Thomas Roche[2], and Claude Carlet[3]

[1] Oberthur Technologies, 71-73, rue des Hautes Pâtures, 92726 Nanterre, France
g.piret@oberthur.com
[2] ANSSI, 51, Bd de la Tour-Maubourg, 75700 Paris 07 SP, France
thomas.roche@ssi.gouv.fr
[3] LAGA, Universities of Paris 8 and Paris 13, CNRS
2, Rue de la Liberté, 93526 Saint-Denis Cedex, France
claude.carlet@univ-paris8.fr

Abstract. Many papers deal with the problem of constructing an efficient masking scheme for existing block ciphers. We take the reverse approach: that is, given a proven masking scheme (Rivain and Prouff, CHES 2010) we design a block cipher that fits well the masking constraints. The difficulty of implementing efficient masking for a block cipher comes mainly from the S-boxes. Therefore the choice of an adequate S-box is the first and most critical step of our work. The S-box we selected is non-bijective; we discuss the resulting design and security problems. A complete design of the cipher is given, as well as some implementation results.

1 Introduction

In a *side-channel attack* (SCA for short), the attacker observes — at runtime — the execution environment (timing, power, electromagnetic radiation, etc.) of a secret-dependent operation. From this observation, the attacker might either be able to identify (part of) the secret (the attack is then called *Simple Side-Channel Analysis*, SSCA) or get noisy information about internal states of the cryptographic operation. In the latter case, when the accessed internal values are a simple combination of a known variable and (part of) the secret, the attacker will recover the secret from a statistical treatment of multiple cryptographic operation execution, the attack is then called *Differential Side-Channel Analysis* (DSCA for short). DSCA fits particularly well block ciphers which build their security from piling-up simple and cryptographically weak operations. By accessing internal states, the attack bypasses the cipher strength. Since the seminal work of Kocher *et al.* [27], DSCA (and its numerous variants and extensions) are a constant threat against embedded devices that implement cryptographic primitives. The development of DSCA countermeasures is a dynamic and challenging research domain where the ultimate goal is to find the good trade-off between security and performances. Many countermeasures focus on noise addition techniques, which should increase the attack complexity. For instance, inserting random delays during the cipher execution is a common practice in order to render

F. Bao, P. Samarati, and J. Zhou (Eds.): ACNS 2012, LNCS 7341, pp. 311–328, 2012.

difficult finding the manipulation of secret-dependent variables inside multiple side-channel traces. This kind of countermeasures indeed increases the classical DSCA attack complexity and it can be done with relatively small overhead on the algorithm complexity (see for instance [14]). However these countermeasures cannot be proven robust, *i.e.* an optimal attacker would be able to recognize and suppress the random delays. This is actually what claim Durvaux *et al.* in a very recent paper [17], where Hidden Markov Chain inference techniques are used to point out dummy operations from real ones, discarding almost perfectly the countermeasure proposed in [14]. In fact, the only known countermeasure that possesses security proofs is the so-called *Masking Schemes* where the cipher's secret-dependent internal values are randomized from one execution to another (*e.g.* [12,21]). However, such countermeasures usually induce a high performance overhead, making their implementation difficult if not impracticable in small embedded devices. Many works have been dedicated to building a masking scheme with low cost that fits the existing block ciphers (mainly DES and AES). In the present paper we will take the problem the other way around: we will study a proven masking scheme and propose a new construction of block cipher that fits well the masking constraints. Hence, we come up with the design of a cipher that ensures resistance to conventional cryptanalysis methods, with special care for the S-boxes (that are used to introduce non-linearity in the cipher design, and are usually the most challenging part to implement when a masking scheme is used) in order to lower the performance overhead of masking.

The paper is organised as follows: In the next section, the basics about masking techniques are recalled and Rivain and Prouff's Boolean masking scheme [43] is described; our block cipher construction will follow the design criteria derived from this scheme. In Section 3, we propose a new S-box having a good trade-off between efficiency, conventional security and masking efficiency. The main limitation of the new S-box is its non-bijectiveness but the use of a Feistel network allows us to build a full block cipher from the S-box. We exhibit in Section 4 a devastating attack on Feistel schemes if no special care is taken on the diffusion layer of the round function. In connection with this attack, various specific cryptanalysis techniques of Feistel networks are recalled in Section 5. The full round function is described in Section 6. Finally, in Section 7, a complete design specification of a full block cipher is proposed as well as a performance analysis compared to the AES block cipher.

2 Preliminaries on Higher-Order Masking Schemes

As recalled in introduction, a Differential Side-Channel Attack compiles leaked information from side-channel observations of internal states of a block cipher in order to recover some knowledge about the secret key. The strength of DSCA comes from the statistical treatments of the leaked information that makes the attack particularly robust to noise (from measurement setup, concurrent operations, etc...). Many improvements have been proposed on the original *Differential Power Analysis* introduced by Kocher *et al.* in 1999 [27]. Among them

the *Correlation Power Analysis* [9] and the *Mutual Information Analysis* [20] propose different statistical treatments to enhance the attack complexity with respect to the noise and leakage model. Another notable extension of DSCA is the so-called *Higher-Order DSCA* (HO-DSCA for short), already mentioned in [27], that upgrades the attacker model: in a d^{th}-order DSCA attack, it is able to observe d different internal variables in a single cipher execution.

Countermeasures by masking are certainly the most studied countermeasures against (HO-) DSCA because of their security proofs. However, the performance overhead due to a masking scheme that thwarts HO-DSCA is such that they are hardly used in practice. Our goal here is to point out the operations that make a masking scheme costly and propose a block cipher that avoids as much as possible such operations. To this end we will focus on a recent (HO-)Masking scheme introduced by Rivain and Prouff [43].

2.1 Masking Schemes

A *d^{th}-order masking scheme* is a countermeasure at the algorithmic level that thwarts d^{th}-order DSCA; the idea is to randomize the data processed by the symmetric cipher such that there exists no set of d processed data that together depend on the secret. A proof of security on a masking scheme ensures that this property holds, in addition, the data complexity of a HO-DSCA attack increases exponentially with its order, assuming the presence of noise (as showed by Chary *et al.* [12]), which always exists in practice. All together, with a high enough order with respect to the noise level, these properties make the masking approach the only sound countermeasure against DSCA.

Several 1st-order masking schemes have been proposed (*e.g.* [8, 34, 37]) and some specific 2nd-order masking schemes [42, 45] but until recently very few schemes could be extended to any order d. The first provable dth-order masking scheme was proposed by Ishai *et al.* in 2003 [23]; the construction has been extended in 2010 by Rivain and Prouff [43]. This work was followed by two new propositions [25, 39] for which our own work would apply just the same as for Rivain and Prouff's construction. A third publication by Genelle *et al.* [19] was also proposed in 2011; it is dedicated to very specific non-linear functions (power functions) which would not leave enough room for us in the research of new S-boxes.

2.2 Rivain-Prouff's Scheme

Let us consider an intermediate variable $V \in GF(2^n)$ of the targeted block cipher, the variable V is called *sensitive* if its value depends on a secret key K and on a known variable (*e.g.* the plaintext P), for instance $V = K \oplus P$. The manipulation of a sensitive variable should be avoided due to DSCA attacks, therefore, in a *dth-order Boolean Masking Scheme* (as Rivain and Prouff's scheme), its manipulation is replaced by the manipulation of $d+1$ shares $(V_0, V_1, \cdots, V_d)$ such that

$$V = V_0 \oplus V_1 \oplus \cdots \oplus V_d \ . \tag{1}$$

A dth-order Masking scheme is an algorithm that modifies the cipher sub-functions in order to only manipulate such sharing of sensitive variables (ideally without ever re-constructing the sensitive variables or decreasing the sharing order).

In [43], the authors propose such an algorithm for each atomic operation: affine functions ($v \mapsto A_f(v)$), addition ($(v, w) \mapsto v \oplus w$) and multiplication ($(v, w) \mapsto v \times w$).

Remark 1. Any function can be decomposed in a sequence of such atomic operations, which gives a great genericity to the masking scheme (this approach is classic in Secure Multi-Party Computations, a research area that is very close to our problem and on which most of the dth-order masking scheme are based [23,39]). The drawback is that those atomic functions shall be executed explicitly (pre-computed tables, commonly used to evaluate S-boxes are not an option).

Affine functions and additions over shared variables can be applied straightforwardly, the masking overhead will solely correspond to d times the original operation complexity. In the case of multiplication, when it is not linear over $GF(2^n)$, the masking scheme is more expensive: it costs $(d+1)^2$ field multiplications, $2d(d+1)$ XORs and the generation of $d(d+1)/2$ random n-bit values. In a block cipher like the AES, each of the 160 S-box computations needs at the least 4 such multiplications in $GF(2^8)$, making the cost of the masking scheme mostly carried by the non-linear multiplications.

This study leads naturally to the following constraints that an S-box should satisfy in order to be efficiently masked: the S-box should have a simple expression as a polynomial and minimum number of non-linear field multiplications in this form.

3 Research of a "Good" S-Box

S-boxes are non-linear functions from $\mathrm{GF}(2)^n$ to $\mathrm{GF}(2)^m$ where n and m are positive integers. We also use the terminology (n, m)-*functions*. The vector spaces $\mathrm{GF}(2)^n$ and $\mathrm{GF}(2)^m$ can be endowed with the structure of field. This gives, when m divides n (and in particular when $m = n$), the possibility of designing S-boxes as polynomial functions over finite fields.

3.1 Design Constraints

S-boxes must allow resistance to several logical attacks. The three main attacks to be withstood are the linear attack [32], the differential attack [3] and the higher order differential attack [26]. An attack which is not yet efficient but represents some threat for the design of future block ciphers is the algebraic attack [13]. Designing an S-box, which is fastly implementable, allows high resistance to the first three attacks and would not be potentially weak against a future efficient version of the fourth one is a difficult challenge. Historically, the S-boxes of the DES have been found by clever random computer search. This was possible

thanks to the relatively small size of these $(6, 4)$-functions. The S-box of the AES was too big for that; it has been the result of a theoretical work by K. Nyberg [36] on the so-called inverse function $GF(2^n) \to GF(2^n) : x \mapsto x^{-1}$. This function has very good properties: it is a permutation (which is necessary for using it as an S-box in an SPN and is quite useful for a Feistel cipher as we shall see below), it achieves the highest known nonlinearity when n is even (in the case of AES $n = 8$; it is common to choose n as a power of 2 because it makes software implementation easier), and has very high resistance to the differential attack and to the higher order differential attack. It happens that, since 1993, no other function in a number of variables equal to a power of 2 and gathering these properties could be found (see a survey in [10]). Note however that the inverse function is almost the worst possible against the algebraic attack.

The criteria listed above are those that an S-box should satisfy in black box cryptography. We need to add the requirements derived for side-channel resistance (see Section 2) and the practical design constraints:

1. Higher-Order Masking against HO-SCA attacks implementable without slowing down the cryptosystem too much (reducing the overhead leads to minimizing the number of non-linear multiplications, see Section 2, and is also related to Constraint 2).
2. Efficiency (*i.e.* reduce the number of instructions and allow the operations to be performed in small fields).
3. Function in 8 variables (the number of variables must be large enough for allowing good resistance to the three main known logical attacks; the choice of 8 helps satisfying Constraint 2 and allows compatibility with standard block size).

We describe now in more details the *function's criteria*. Given an (n, n)-function in the form:

$$f : X \to \sum_{i=0}^{2^n-1} a_i X^i,\ a_i \in GF(2^n) \tag{2}$$

its important parameters are:

- Non-linearity: $nl(f) = 2^{n-1} - \frac{1}{2} \max_{a,b\neq 0} \left| \sum_X (-1)^{b\cdot f(X)+a\cdot X} \right|$, where $a \cdot X$ is an inner product in $GF(2^n)$; in practice, $a \cdot X = tr(aX)$ where tr is the trace function $tr(a) = a + a^2 + a^{2^2} + \ldots + a^{2^{n-1}}$.
  To thwart linear cryptanalysis [32], the nonlinearity must be close to the best known nonlinearity of vectorial functions in even numbers of variables: $2^{n-1} - 2^{n/2}$ (that is 112 for $n = 8$).
- Differentiality: $\delta = \max_{a\neq 0,b} (\#\{X \mid f(a + X) + f(X) = b\})$.
  Because of the differential cryptanalysis [4], it should be 2 (then the function is called Almost Perfect Nonlinear APN [35]) or 4 (then the function is called differentially 4-uniform [35]), or at most 6.

- Algebraic degree: $d = \max_i (\omega_2(i) \mid a_i \neq 0)$, where $\omega_2(i)$ is the Hamming weight of the binary expansion of i.
  Because of the higher differential attack [26], it should be at least 3 and preferably at least 4.
- Graph algebraic Immunity: equals the minimal algebraic degree of a nonzero Boolean function vanishing on the graph $G_f = \{(X, f(X)); X \in GF(2^n)\}$ of the function (that is, the minimal algebraic degree of an annihilator of the graph); this parameter is not related to an efficient attack yet, but 1 is definitely too small and 2, as in the case of the inverse function, is risky.
- Minimum number of non-linear multiplications.
  The evaluation of $f : X \to \sum_{i=0}^{2^n-1} a_i X^i$ involves a number, say k, of non-linear multiplications (by opposition to linear transformations such as, in characteristic 2, an exponentiation by a power of 2, i.e. a monomial of degree a power of 2). Higher Order masking schemes like the one proposed by Rivain and Prouff [43] slow down significantly the S-box implementation, the overhead being directly related to the number of non-linear multiplications.

3.2 Bijective vs Non-Bijective S-Box

Considering two comparable functions (with respect to their execution efficiency as well as the above mentioned criteria), a bijective S-box is much more interesting than a non bijective one, because in the latter case we have to solve the problem of making the cipher invertible anyway. Moreover we will see in Section 4 that a non-bijective S-box induces security flaws.

However, it is a matter of fact that the research of good (meaning efficient and cryptographically strong) non-linear functions is much harder when only considering bijective functions, especially in an number of variables that is a power of two, where the inverse function is considered the only good candidate.

The selected function is a non-bijective function, specially efficient in the number of operations necessary for evaluating it and involving only operations in a small Galois Field (of 16 elements), operations that can then be tabulated on standard platforms.

3.3 S-Box Description

A possible S-box candidate is proposed in [11]. It is not expressed as a polynomial of the form (2), but as the concatenation of two bivariate polynomials whose variables live in $GF(2^{n/2})$:

$$f : \mathrm{GF}(2^{n/2})^2 \to \mathrm{GF}(2^{n/2})^2 : (x, y) \mapsto (xy, (x^3 + \omega)(y^3 + \omega')), \tag{3}$$

where xy is the product of x and y in the field $GF(2^{n/2})$. This S-box has the desired properties when $n/2$ is even and ω, ω' and $\frac{\omega}{\omega'}$ belong to $GF(2^{n/2}) \setminus \{x^3, x \in GF(2^{n/2})\}$. In particular, for $n = 8$, we have:

- $\delta = 4$.
- $nl = 94$.
- algebraic degree: 4.
- number of non-linear multiplications: 4 in $GF(2^4)$.

S-box Instantiation. To represent elements in GF(2^4) we chose to work in field representation GF(2)$[x]/P(x)$ with $P(X) = X^4 + X^3 + 1$. Moreover we need to make a choice in the family of S-boxes described above, that is to choose ω and ω'. We took $\omega = 02_x$, $\omega' = 04_x$ (in hexadecimal notation).

3.4 Masked S-Box Cost Evaluation

As explained in Section 2, Rivain and Prouff's higher order masking scheme [43] uses a sequence of field multiplications and additions to compute the masked S-box. It can be easily checked that one needs at most 2 additions, 2 square operations and 4 multiplications in $GF(2^4)$ to evaluate our S-box. By comparison, the AES S-box is computed with 3 raisings to some power 2^i (i.e. X^{2^i}) and 4 multiplications in $GF(2^8)$ (see [43]).

The cost of the higher order masking scheme is linear in the masking order for additions and linear operations (like "X^{2^i}" operations in fields of characteristic 2) whereas it is quadratic for non-linear multiplications. Hence the overhead in the number of operations to evaluate a masked AES S-box and our S-box seems at first glance quite the same. Table 1 details the number of operations in $GF(2^4)$ that are needed to evaluate our S-box.

Table 1. Number of Operations

	# additions	# squarings	# multiplications	# random values (4-bit)
Unmasked	2	2	4	0
dth-order masked	$(8d+2)(d+1)$	$2(d+1)$	$4(d+1)^2$	$2d(d+1)$

In practice, the field size will play an important role in the runtime as a field operation cost is directly dependent on the field size. Decreasing the field size to 2^4 allows to tabulate the field multiplication in a lookup table with much less memory, which makes it possible even when it is very constrained (contrary to the case of $GF(2^8)$). As a matter of fact, using tower field methods, the evaluation of higher-order masked AES's S-box in $GF(2^4)$ by Kim *et al.* [25] has been shown to be faster (for masking order 2 and 3) than the original evaluation in $GF(2^8)$ (from [43]), even though the number of non-linear multiplications turned to 5 in $GF(2^4)$.

4 From the S-Box to the Cipher

4.1 Using a Feistel Network with SP-Type Round Function

The non-bijectivity of the S-box we selected requires us to use an adequate structure, in order to make the cipher invertible. A well-known way to use non-bijective round functions to build a block cipher is to use a Feistel network. Therefore we could think of embedding our S-box in an SP-Type F-function as considered in many papers ([46–48] and several others), and using this F-function as the round function of a Feistel network.

An *SP-type F-function* $F : \mathrm{GF}(2)^n \times \mathrm{GF}(2)^n \to \mathrm{GF}(2)^n$ is defined as follows.

Definition 1. *Let m the number of S-boxes in a round, and t the size of the S-box, with $mt = n$. Consider $\gamma, \theta : \mathrm{GF}(2)^n \to \mathrm{GF}(2)^n$ with*

- *γ the function generated by concatenating m S-boxes.*
- *θ a linear diffusion layer.*

Then an SP-type F-function *F is defined as $F(x, k) = \theta(\gamma(x \oplus k))$.*

One round of a Feistel network with round function $F : \mathrm{GF}(2)^n \to \mathrm{GF}(2)^n$ is defined as

$$\Psi(F) : \mathrm{GF}(2)^{2n} \to \mathrm{GF}(2)^{2n} : \langle L, R\rangle \to \langle R, L \oplus F(R)\rangle \tag{4}$$

Then

Definition 2. *An* SP-type R-round Feistel Network *is the composition*

$$\bigcirc_{i=1}^{R} \Psi(F(., k^i)) \tag{5}$$

where F is an SP-type F-function and the k^i's are round keys derived from the master key K by a key schedule algorithm*.*

4.2 Why It Is Not a Good Idea

This approach is actually not applicable as such with our S-box S, as it lends itself to a little-known but devastating attack.

Consider $a, b \in \mathrm{GF}(2^8)$ such that $S(a) = S(b)$ (two such inputs always exist as S is not injective). Let us denote $\Delta = a \oplus b$. Consider two plaintexts $P = \langle L, R\rangle = \langle (l_1, \ldots, l_m), (r_1, \ldots, r_m)\rangle$ and $P' = \langle L, R'\rangle = \langle (l_1, \ldots, l_m), (r'_1, \ldots, r'_m)\rangle$ $(l_i, r_i, r'_i \in \mathrm{GF}(2^8))$ such that

$$P \oplus P' = \langle (0, \ldots, 0), (\Delta, 0, \ldots, 0)\rangle. \tag{6}$$

Assuming the first round key $k^1 = (k^1_1, ..., k^1_m)$ is uniformly distributed, with probability at least[1] $2/2^8$ we have

$$F(R, k^1) = F(R', k^1) \tag{7}$$

[1] It is greater than that if $\exists c (a \neq c \neq b)$ such that $S(c) = S(c \oplus \Delta)$.

As a matter of fact, the inputs $R \oplus k^1$ and $R' \oplus k^1$ to the S-box layer differ in their first byte only. Thus (7) is satisfied when $S(r_1 \oplus k_1^1) = S(r'_1 \oplus k_1^1)$. This equality is satisfied if $r_1 \oplus k_1^1 = a$ or $r_1 \oplus k_1^1 = b$.

Attack Complexity. Let us consider a SP-type Feistel Network with R rounds and a block size of n bits. The round function's S-Box (S) is non-injective and we denote by DP_0 its maximum 0-*output differential probability*: $DP_0 = \max_{a \neq 0}(\#\{x \text{ s.t. } S(x \oplus a) \oplus S(x) = 0\})/2^m$, where m is the size of the S-box input (the best case scenario from the security point of view is when $DP_0 = 2/2^m$, this is the case for our S-box). The differential attack we study here assumes that the attacker chooses pairs of plaintexts P, P' such that the input difference at the beginning of the first round function is null everywhere except on the input of a single S-box where the difference is equal to $\Delta = \texttt{argmax}_{a \neq 0}(\#\{x \text{ s.t. } S(x \oplus a) \oplus S(x) = 0\})$.

The differential characteristic over R rounds considered in this attack is such that the round function's input differential for even rounds (*resp.* odd rounds) is equal to $\langle (0, \ldots, 0), (\Delta, 0, \ldots, 0) \rangle$ (*resp.* null). Assuming that the round keys are independent and uniformly distributed (*i.e.* the classical Markov Cipher assumption), it is easy to evaluate the probability of such a differential characteristic Ω:

$$\Pr[\Omega] = (DP_0)^{R/2} \ . \tag{8}$$

Given the differential characteristic probability it is well known (see for instance [29]) that the differential cryptanalysis data complexity can be approximated by

$$\mathcal{C} = \frac{2}{\Pr[\Omega]} \ . \tag{9}$$

Hence, in order to get an attack complexity higher than exhaustive search, we would need to assure that $\Pr[\Omega] \leq \frac{1}{2^{n-1}}$. Considering the best non-injective S-box for $m = 8$ ($DP_0 = 2^{-7}$) and $n = 128$ this means that $(2^{-7})^{R/2} \leq 2^{-127}$, therefore the number of rounds R should be greater than 36.

4.3 Linear Counterpart to the Previous Attack

Several results show that some duality exists between linear and differential attacks [33]. Therefore it is not surprising that a linear attack exists that is as powerful as the differential attack we exposed in the previous section.

This attack is based on the following theorem (see for instance [30]).

Theorem 1. *A Boolean transformation F is invertible if and only if every output parity (i.e. every component function $\lambda \cdot F$) is a balanced binary Boolean function of input bits.*

Applying this theorem to our S-box S we obtain

Corollary 1. *There exists a linear mask $\lambda \in \mathrm{GF}(2)^8$ such that, for x random and uniformly distributed, $\lambda \cdot S(x) = 0$ is satisfied with probability $p = 1/2 + \epsilon$ where the bias, denoted by ϵ, is not null.*

Let us denote by $L^i = (l_1^i, \ldots, l_m^i)$ the left part and by $R^i = (r_1^i, \ldots, r_m^i)$ the right part of the input to round $i+1$, and by $Y^i = (y_1^i, \ldots, y_m^i)$ the output of the i^{th} F-function. Then if we use mask λ to approximate the first S-box (only) of first round, we have a linear characteristic on 2 rounds of the form

$$\lambda \cdot l_1^0 = \lambda \cdot l_1^2 \qquad \text{with probability } 1/2 + \epsilon \tag{10}$$

As a matter of fact, we have: $\lambda \cdot y_1^1 = 0 \Leftrightarrow \lambda \cdot l_1^0 \oplus \lambda \cdot r_1^1 = 0 \Leftrightarrow \lambda \cdot l_1^0 \oplus \lambda \cdot l_1^2 = 0$.

This characteristic is iterative. Therefore R rounds can be approximated with probability $1/2 + 2^{R/2-1}\epsilon^{R/2}$. It is well known (see for instance [32]) that the linear cryptanalysis data complexity, given a characteristic of bias ϵ, is given by $\mathcal{C} = \frac{1}{2(\epsilon)^2}$. Assuming that the bias associated to λ is $2^{-8/2}$ (the smallest known non-zero bias for an 8-bit S-box), 42 rounds are required to have an attack complexity above 2^{127} for the whole cipher. Moreover, in the case of our S-box, the maximal bias ϵ (over all output linear maps λ) is equal to $\frac{22}{2^8}$ which leads to $R \geq \frac{126}{7-\log(22)} > 49$.

5 Comparison of Specific Attacks on Feistel Ciphers with Non-bijective Round Function

We have seen that the use of non-bijective functions introduces vulnerabilities in the design of a Feistel cipher. Some of these vulnerabilities can be found in the literature. We divide them in three categories: The differential cryptanalysis [3] and its linear counterpart (this corresponds to the attack described in Section 4), Rijmen *et al.* 's non-surjective attack [41] and the Davies and Murphy attack [2, 16, 28].

5.1 Non-injective Round Functions

The first attack described in Section 4 can be seen as a particular case of differential cryptanalysis. We already mentioned that the initial paper of Biham and Shamir on differential cryptanalysis [3] already proposed a similar differential characteristic construction for the DES cipher, but is not the best for the DES because of its expansion function. Another example is the McGuffin cipher proposed by Schneier and Blaze [7] that did not have a similar expansion transformation and was then completely exposed against such strong differential cryptanalysis. Rijmen *et al.* described the attack in [40].

5.2 Non-surjective and Unbalanced Round Function

Rijmen *et al.* propose in [41] an attack on Feistel schemes assuming that the round function is non-surjective and extend their attack when the round function is simply unbalanced. The linear attack presented in Section 4 is based on the fact that the S-box is unbalanced (Theorem 1 and then Corollary 1). This attack corresponds exactly to the Rijmen and Preneel attack where only linear masks

of the round output bits are considered. Moreover, the SP-Network structure of the round function allows us (similarly to the differential version) to consider a single S-box instead of the full round function (as in [41]). This property conceals the strength of the attack; thus an expansion transformation is needed to ensure that more than a unique S-box per 2 successive rounds is involved in the linear characteristic. This will lower the linear (*resp.* differential) characteristic bias and then increase the attack's data complexity (see Section 7.1).

Before that, it is important to note that introducing an expansion step before the round key mixing step and the S-box evaluation may open a new vulnerability in the scheme: the Davies and Murphy attack.

5.3 Unbalanced Round Functions with Key Dependent Output Distribution

Davies and Murphy proposed in [16] an attack on Feistel schemes assuming that the round functions are unbalanced and the output distribution is dependent on some key bits. This seminal paper was followed by many others, among them a first improvement by Biham and Biryukov [2] and then a second improvement proposed by Kunz-Jacques and Muller [28]. In the latter article, a parallel is drawn between Davis and Murphy attack and the linear cryptanalysis; moreover, the initial attack is optimized by the use of a distinguisher that evaluates divergence between univariate distributions (through linear projections) instead of divergence between multivariate distributions.

The use of an expansion step induces the S-box output distribution to be somewhat dependent on the secret key. In [28], Kunz-Jacques and Muller exhibit the tight relation between Davies and Murphy attack and linear cryptanalysis. As a matter of fact, in the case of DES, they could show that the classical Davies and Murphy attack would not be more efficient than a restriction of it where only linear combinations of the round outputs are considered. This restricted Davies and Murphy attack falls into linear cryptanalysis and then is naturally bounded by the linear cryptanalysis complexity bound found for the DES.

6 Expansion and Compression Function

The attacks we described in Section 4 exploit the fact that it is possible to choose (pairs of) plaintexts such that one S-box only is active in the first round (and that this property can be propagated to the following odd rounds). If we deny this possibility to the attacker, we thwart these attacks.

Using linear codes to ensure good diffusion in block ciphers is a well-known idea (see [15], and many other works). We show how to use them slightly differently from what is usually done in order to render impossible the attacks discussed in Section 4 and Section 5.

- Let $(a_1, \ldots, a_8) \in \mathrm{GF}(2^8)^8$ be the input of the round function. We encode it with a linear code of dimension 8 and length $8 + \ell$ over $\mathrm{GF}(2^8)$, before

performing the key addition and the S-boxes layer. That is, if G is the generator matrix of such $[8+\ell, 8]$ code, we compute

$$(b_1, \ldots, b_{8+\ell}) = (a_1, \ldots, a_8) \cdot G \tag{11}$$

We call this computation the *expansion layer* E. If d is the minimal distance of the code, it is trivial that the minimal number of active S-boxes is d. In order to maximize it, we use a MDS code $[8+\ell, 8, \ell+1]$. An easy way to construct such code is to use a shortened Reed-Solomon code.

- After E comes a key addition layer, which will use $8(8+\ell)$ key bits, and the non-linear layer that consists in $8+\ell$ S-boxes in parallel.
- Finally the state must be compressed from $8+\ell$ to 8 bytes. Note that the expansion layer only defeats the differential attack we exposed in Section 4.2, not the linear one described in Section 4.3. Therefore the linear *compression layer* C must ensure that every non-zero linear mask approximating the output of the S-box layer has as many active S-boxes as possible. If we denote by H the compression matrix and we write $(d_1, \ldots, d_8) = (c_1, \ldots, c_{8+\ell}) \cdot H$, a linear approximation of the output of the round can be written as $(\beta_1, \ldots, \beta_8) \cdot (d_1, \ldots, d_8)^T = (\beta_1, \ldots, \beta_8) \cdot H^T \cdot (c_1, \ldots, c_{8+\ell})^T$. We define $(\beta'_1, \ldots, \beta'_{8+\ell})$ as the linear mask at the output of the S-boxes corresponding to $(\beta_1, \ldots, \beta_8)$. Thus we have $(\beta'_1, \ldots, \beta'_{8+\ell}) = (\beta_1, \ldots, \beta_8) \cdot H^T$. In order to maximize the number of active S-boxes, we must choose H^T such as to lower bound the byte Hamming weight of $(\beta'_1, \ldots, \beta'_{8+\ell})$. The best choice H^T for this purpose is again to choose H^T as the generator matrix of an MDS code. We decide to take $H^T = G$.

We choose $\ell = 6$, which offers a good compromise between the number of rounds and the computational cost of one round. We built the matrix G such as to make its implementation efficient. More precisely, we tried to minimize the number of non-zero coefficients, to use a small number of different coefficients, and to use coefficients with a small Hamming weight

The resulting matrix G is as follows. Its elements belong to the Galois field $\mathrm{GF}(2^8)$ defined as $\mathrm{GF}(2)[X]/(1+X^2+X^3+X^4+X^8)$.

$$G = \begin{pmatrix} 01 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 01 & 01 & 0A & 01 & 09 & 0C \\ 00 & 01 & 00 & 00 & 00 & 00 & 00 & 00 & 05 & 01 & 01 & 0A & 01 & 09 \\ 00 & 00 & 01 & 00 & 00 & 00 & 00 & 00 & 06 & 05 & 01 & 01 & 0A & 01 \\ 00 & 00 & 00 & 01 & 00 & 00 & 00 & 00 & 0C & 06 & 05 & 01 & 01 & 0A \\ 00 & 00 & 00 & 00 & 01 & 00 & 00 & 00 & 09 & 0C & 06 & 05 & 01 & 01 \\ 00 & 00 & 00 & 00 & 00 & 01 & 00 & 00 & 01 & 09 & 0C & 06 & 05 & 01 \\ 00 & 00 & 00 & 00 & 00 & 00 & 01 & 00 & 0A & 01 & 09 & 0C & 06 & 05 \\ 00 & 00 & 00 & 00 & 00 & 00 & 00 & 01 & 01 & 0A & 01 & 09 & 0C & 06 \end{pmatrix} \tag{12}$$

7 Full Description of the Block Cipher

One round of the block cipher is pictured in Figure 1. In the next section we analyze the number of rounds required to achieve good security.

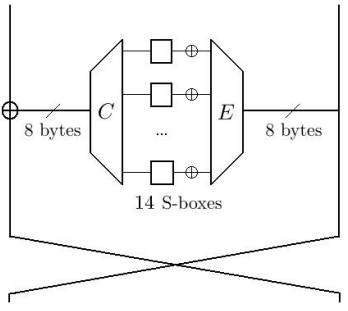

Fig. 1. One round of the cipher

7.1 Evaluation of the Number of Rounds

Differential Cryptanalysis As our S-box is differentially 4-uniform, the probability of any non-trivial 1-round characteristic is at most $(4/2^8)^7$. Therefore a differential characteristic over $2t$ rounds has probability at most $(4/2^8)^{7t}$. In order to upper bound the probability of any differential characteristic by 2^{-127}, at least $2t = \frac{127}{3 \cdot 7} \simeq 6$ rounds are necessary.

Linear Cryptanalysis Our S-box has non-linearity $nl = 94$; hence, the bias of its best linear approximation is $\frac{128-94}{256}$. Over one round the bias of any non-trivial linear characteristic is at most $1/2 \cdot (34/128)^7$, and over $2t$ rounds it is $1/2 \cdot (34/128)^{7t}$. As the data complexity of linear cryptanalysis is $\mathcal{C} = \frac{1}{2\epsilon^2}$, we must have

$$\frac{1}{2 \cdot (1/2 \cdot (\frac{34}{128})^{7t})^2} \geq 2^{128} \tag{13}$$

which gives a lower bound of $2t = \frac{127}{7(7-\log_2(34))} = 9.5$ rounds in order to ensure an attack complexity at least 2^{128}.

A security margin must be added to take linear hull effects into account, and to deal with nR- (i.e. key guess) attacks. It is why we decided to use 12 rounds.

7.2 The Key Schedule

We have to derive 12 round keys $k^1, \ldots, k^{12}$ of 112 bits each[2] from one 128-bit master key K. We want our scheme to resist known attacks on a key schedule algorithm, in particular related-key attacks [1, 24] and slide attacks [5, 6]. A detailed security analysis of the key schedule will be published in an extended version of this paper, available on the ePrint (`http://eprint.iacr.org/`).

The key schedule must also be easy to implement; one very desirable property is the ability to derive round keys *on-the-fly* in both encryption and decryption

[2] 112 bits are required because of the use of the expansion layer.

mode (which is possible for DES, but not for AES). It is our belief that designing highly complicated non-linear key schedules is not mandatory to have good security. It is why we restrict ourselves to rotations, bitwise additions and bit selection in the design of the key schedule.

Round Key Derivation in Encryption Mode. The round keys are extracted from an *extended key* $(\kappa^1, \kappa^2, ..., \kappa^{12})$ by a simple bit selection. The κ^i's are 128-bit long and computed as follows:

$$\begin{cases} \kappa^1 = K \\ \kappa^i = T(K) \ggg \Theta(i) & \text{for } i = 2, 4, 6, 8, 10, 12 \\ \kappa^i = K \ggg \Theta(i) & \text{for } i = 3, 5, 7, 9, 11 \end{cases} \tag{14}$$

where $\ggg j$ is the right-rotation of j bits, and Θ is given by the following table:

i	2	3	4	5	6	7	8	9	10	11	12
$\Theta(i)$	1	16	17	32	33	85	86	101	102	117	118

Regarding T, it is defined as follows. Let us write $K = (K^{(1)}, K^{(2)}, K^{(3)}, K^{(4)})$, where $K^{(i)} \in \mathrm{GF}(2)^{32}$. Then

$$\begin{pmatrix} T(K)^{(1)} \\ T(K)^{(2)} \\ T(K)^{(3)} \\ T(K)^{(4)} \end{pmatrix} = \begin{pmatrix} 0\,1\,1\,1 \\ 1\,0\,1\,1 \\ 1\,1\,0\,1 \\ 1\,1\,1\,0 \end{pmatrix} \cdot \begin{pmatrix} K^{(1)} \\ K^{(2)} \\ K^{(3)} \\ K^{(4)} \end{pmatrix} \tag{15}$$

We note that T is involutive. Therefore it is easy to derive κ^{i+1} from κ^i by applying T followed by a rotation of a given number of bits. We describe such iterative computation as

$$\begin{cases} \kappa^1 = K \\ \kappa^i = T(\kappa^{i-1}) \ggg \theta(i) & \text{pour } i = 2...12 \end{cases} \tag{16}$$

where θ is

i	2	3	4	5	6	7	8	9	10	11	12
$\theta(i)$	1	15	1	15	1	52	1	15	1	15	1

The round key k^i is obtained from κ^i by extracting the 112 leftmost bits of κ^i: if $\kappa^i = (\kappa^i_1, \ldots, \kappa^i_{16})$ $(\kappa^i_j \in \mathrm{GF}(2^8))$, then $k^i = (\kappa^i_1, \ldots, \kappa^i_{14})$.

Round Key Derivation in Decryption Mode. Given K, the extended key *for decryption* $\kappa'^1 \ldots \kappa'^{12}$ is computed as

$$\begin{cases} \kappa'^1 = T(K) \lll 10 \\ \kappa'^i = T(\kappa'^{i-1}) \lll \theta'(i) & \text{for } i = 2...12 \end{cases} \tag{17}$$

where $\lll j$ is the left-rotation of j bits, and θ' is given by the following table:

i	2	3	4	5	6	7	8	9	10	11	12
$\theta'(i)$	1	15	1	15	1	52	1	15	1	15	1

We remark that once $k'^1 = T(K) \lll 10$ is computed, the sequences of round keys in encryption and decryption mode, respectively, only differ by the direction of the rotations (right for encryption, left for decryption). Again, k'^i ($i \in \{1, \ldots, 12\}$) is computed from κ'^i ($i \in \{1, \ldots, 12\}$) by considering the 112 leftmost bits.

7.3 Performance Analysis

Our block cipher[3] has been implemented (not by the authors, see acknowledgments) on a smart card based on an 8-bit micro-controller, with 4 different masking levels: without masking, and with maskings of order 1, 2, and 3. This implementation has been compared with state-to-the-art implementations of (masked) AES on the same platform. The results are given in Table 2.

Table 2. Implementation results of AES and our algorithm using different masking orders

Number of Kcycles: ciphering		
Version	**AES**	**Our algorithm**
Unprotected	**2**	**26**
Masked order 1	**129**	**94**
Masked order 2	**271**	**160**
Masked order 3	**470**	**253**

We remark that AES in its non-masked version is definitely much faster than the non-masked version of our algorithm. However once we consider masked versions, our algorithm takes the lead, and the difference between both algorithms increases with the order of the masking.

8 Conclusion

This article illustrates how pertinent it is to have side-channel resistance in mind when building a block cipher. To thwart higher order side-channel attacks we focus on the use of masking schemes, of which the complexity is mainly impacted by the cost of S-box implementation. We emphasize on a new criteria for the design of S-boxes and present a construction that shows a good trade-off

[3] To be precise, we need to mention that the block cipher implemented is a preliminary version, which differs from the cipher we described in the compression layer (which was also a $[14, 8]$-MDS code but different from the one used in the expansion step). We believe that this difference in the block cipher design will not significantly change the performance results given here.

between efficiency and security. The non-bijectivity of the S-box requires us to use a Feistel Network. We point out a weakness in the straightforward use of Feistel Networks when the S-boxes are non-bijective. We propose to circumvent it by the use of MDS codes to build optimal expansion and compression layers. As an achievement of our work, a new block cipher is fully specified.

Acknowledgments. This paper has been initiated during the project *Secure Algorithm*, sponsored by the French Ministry of Finances through the *Systematic Pole*. We would like to thank all members of the project (from Nagra, Telecom ParisTech, Thales, UVSQ) for the many interesting interactions we had.

More specifically, we would like to thank Emmanuel Prouff for his careful proofreading and comments on preliminary versions of this paper. We also thank Reda Najibi, Laurent Albanèse and Jean-Bernard Fisher: Reda implemented the algorithm during his internship at Nagra, under direction of Laurent and Jean-Bernard. They provided us with the performance figures of Section 7.3.

References

1. Biham, E.: New Types of Cryptanalytic Attacks Using Related Keys. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 398–409. Springer, Heidelberg (1994)
2. Biham, E., Biryukov, A.: An Improvement of Davies' Attack on DES. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 461–467. Springer, Heidelberg (1995)
3. Biham, E., Shamir, A.: Differential Cryptanalysis of DES-like Cryptosystems. In: Menezes, A., Vanstone, S.A. (eds.) CRYPTO 1990. LNCS, vol. 537, pp. 2–21. Springer, Heidelberg (1991)
4. Biham, E., Shamir, A.: Differential Cryptanalysis of DES-like Cryptosystems. Journal of Cryptology 4(1), 3–72 (1991)
5. Biryukov, A., Wagner, D.: Slide Attacks. In: Knudsen, L.R. (ed.) FSE 1999. LNCS, vol. 1636, pp. 245–259. Springer, Heidelberg (1999)
6. Biryukov, A., Wagner, D.: Advanced Slide Attacks. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 589–606. Springer, Heidelberg (2000)
7. Blaze, M., Schneier, B.: The MacGuffin Block Cipher Algorithm. In: Preneel, B. (ed.) FSE 1994. LNCS, vol. 1008, pp. 97–110. Springer, Heidelberg (1995)
8. Blömer, J., Guajardo, J., Krummel, V.: Provably Secure Masking of AES. In: Handschuh, H., Hasan, M.A. (eds.) SAC 2004. LNCS, vol. 3357, pp. 69–83. Springer, Heidelberg (2004)
9. Brier, E., Clavier, C., Olivier, F.: Correlation Power Analysis with a Leakage Model. In: Joye, M., Quisquater, J.-J. (eds.) CHES 2004. LNCS, vol. 3156, pp. 16–29. Springer, Heidelberg (2004)
10. Carlet, C.: Vectorial Boolean Functions for Cryptography (Chapter 9). In: Crama, Y., Hammer, P.L. (eds.) Boolean Models and Methods in Mathematics, Computer Science, and Engineering, pp. 398–469. Cambridge University Press (2010), Prel. version: `http://www.math.univ-paris13.fr/~carlet/pubs.html`
11. Carlet, C.: Relating three nonlinearity parameters of vectorial functions and building APN functions from bent functions. Des. Codes Cryptogr. 59(1-3), 89–109 (2011)

12. Chari, S., Jutla, C., Rao, J., Rohatgi, P.: Towards Sound Approaches to Counteract Power-Analysis Attacks. In: Wiener (ed.) [49], pp. 398–412
13. Charpin, P., Pasalic, E.: On Propagation Characteristics of Resilient Functions. In: Nyberg, K., Heys, H.M. (eds.) SAC 2002. LNCS, vol. 2595, pp. 175–195. Springer, Heidelberg (2003)
14. Coron, J.-S., Kizhvatov, I.: Analysis and improvement of the random delay countermeasure of ches 2009. In: Mangard, Standaert (eds.) [31], pp. 95–109
15. Daemen, J., Rijmen, V.: The Design of Rijndael. Springer (2002)
16. Davies, D.W., Murphy, S.: Pairs and triplets of DES s-boxes. J. Cryptology 8(1), 1–25 (1995)
17. Durvaux, F., Renauld, M., Standaert, F.-X., van Oldeneel tot Oldenzeel, L., Veyrat-Charvillon, N.: Cryptanalysis of the ches 2009/2010 random delay countermeasure. Cryptology ePrint Archive, Report 2012/038 (2012), http://eprint.iacr.org/
18. Feigenbaum, J.: EUROCRYPT 1991. LNCS, vol. 547. Springer, Heidelberg (1991)
19. Genelle, L., Prouff, E., Quisquater, M.: Thwarting higher-order side channel analysis with additive and multiplicative maskings. In: Preneel, Takagi (eds.) [38], pp. 240–255
20. Gierlichs, B., Batina, L., Tuyls, P., Preneel, B.: Mutual Information Analysis. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008. LNCS, vol. 5154, pp. 426–442. Springer, Heidelberg (2008)
21. Goubin, L., Patarin, J.: DES and Differential Power Analysis. In: Koç, Ç.K., Paar, C. (eds.) CHES 1999. LNCS, vol. 1717, pp. 158–172. Springer, Heidelberg (1999)
22. Helleseth, T. (ed.): EUROCRYPT 1993. LNCS, vol. 765. Springer, Heidelberg (1994)
23. Ishai, Y., Sahai, A., Wagner, D.: Private Circuits: Securing Hardware against Probing Attacks. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 463–481. Springer, Heidelberg (2003)
24. Kelsey, J., Schneier, B., Wagner, D.: Key-Schedule Cryptanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES. In: Koblitz, N. (ed.) CRYPTO 1996. LNCS, vol. 1109, pp. 237–251. Springer, Heidelberg (1996)
25. Kim, H., Hong, S., Lim, J.: A fast and provably secure higher-order masking of AES s-box. In: Preneel, Takagi (eds.) [38], pp. 95–107
26. Knudsen, L.R.: Truncated and Higher Order Differentials. In: Preneel, B. (ed.) FSE 1994. LNCS, vol. 1008, pp. 196–211. Springer, Heidelberg (1995)
27. Kocher, P., Jaffe, J., Jun, B.: Differential Power Analysis. In: Wiener (ed.) [49], pp. 388–397
28. Kunz-Jacques, S., Muller, F.: New Improvements of Davies-Murphy Cryptanalysis. In: Roy, B. (ed.) ASIACRYPT 2005. LNCS, vol. 3788, pp. 425–442. Springer, Heidelberg (2005)
29. Lai, X., Massey, J.L., Murphy, S.: Markov ciphers and differential cryptanalysis. In: Feigenbaum (ed.) [18], pp. 17–38
30. Lidl, R., Niederreiter, H.: On orthogonal systems and permutation polynomials in several variables. Acta Arith. 22, 257–265 (1973)
31. Mangard, S., Standaert, F.-X. (eds.): CHES 2010. LNCS, vol. 6225. Springer, Heidelberg (2010)
32. Matsui, M.: Linear cryptanalysis method for DES cipher. In: Helleseth (ed.) [22], pp. 386–397
33. Matsui, M.: On correlation between the order of S-boxes and the strength of DES. In: Santis (ed.) [44], pp. 366–375
34. Nikova, S., Rijmen, V., Schläffer, M.: Secure hardware implementation of nonlinear functions in the presence of glitches. J. Cryptology 24(2), 292–321 (2011)

35. Nyberg, K.: Perfect nonlinear S-boxes. In: Feigenbaum (ed.) [18], pp. 378–386
36. Nyberg, K.: Differentially Uniform Mappings for Cryptography. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 55–64. Springer, Heidelberg (1994)
37. Oswald, E., Mangard, S., Pramstaller, N., Rijmen, V.: A Side-Channel Analysis Resistant Description of the AES S-Box. In: Gilbert, H., Handschuh, H. (eds.) FSE 2005. LNCS, vol. 3557, pp. 413–423. Springer, Heidelberg (2005)
38. Preneel, B., Takagi, T. (eds.): CHES 2011. LNCS, vol. 6917. Springer, Heidelberg (2011)
39. Prouff, E., Roche, T.: Higher-order glitches free implementation of the AES using secure multi-party computation protocols. In: Preneel, Takagi (eds.) [38], pp. 63–78
40. Rijmen, V., Preneel, B.: Cryptanalysis of McGuffin. In: Preneel, B. (ed.) FSE 1994. LNCS, vol. 1008, pp. 353–358. Springer, Heidelberg (1995)
41. Rijmen, V., Preneel, B., Win, E.D.: On weaknesses of non-surjective round functions. Des. Codes Cryptography 12(3), 253–266 (1997)
42. Rivain, M., Dottax, E., Prouff, E.: Block Ciphers Implementations Provably Secure Against Second Order Side Channel Analysis. Cryptology ePrint Archive, Report 2008/021 (2008), `http://eprint.iacr.org/`
43. Rivain, M., Prouff, E.: Provably secure higher-order masking of aes. In: Mangard, Standaert (eds.) [31], pp. 413–427
44. De Santis, A. (ed.): EUROCRYPT 1994. LNCS, vol. 950. Springer, Heidelberg (1995)
45. Schramm, K., Paar, C.: Higher Order Masking of the AES. In: Pointcheval, D. (ed.) CT-RSA 2006. LNCS, vol. 3860, pp. 208–225. Springer, Heidelberg (2006)
46. Shirai, T., Preneel, B.: On Feistel Ciphers Using Optimal Diffusion Mappings Across Multiple Rounds. In: Lee, P.J. (ed.) ASIACRYPT 2004. LNCS, vol. 3329, pp. 1–15. Springer, Heidelberg (2004)
47. Shirai, T., Shibutani, K.: Improving Immunity of Feistel Ciphers against Differential Cryptanalysis by Using Multiple MDS Matrices. In: Roy, B., Meier, W. (eds.) FSE 2004. LNCS, vol. 3017, pp. 260–278. Springer, Heidelberg (2004)
48. Shirai, T., Shibutani, K.: On Feistel Structures Using a Diffusion Switching Mechanism. In: Robshaw, M. (ed.) FSE 2006. LNCS, vol. 4047, pp. 41–56. Springer, Heidelberg (2006)
49. Wiener, M. (ed.): CRYPTO 1999. LNCS, vol. 1666. Springer, Heidelberg (1999)