Building Secure Block Ciphers on Generic Attacks Assumptions

Jacques Patarin[1] and Yannick Seurin[1,2]

[1] University of Versailles, France
[2] Orange Labs, Issy-les-Moulineaux, France
jacques.patarin@prism.uvsq.fr
yannick.seurin@orange-ftgroup.com

Abstract. Up to now, the design of block ciphers has been mainly driven by heuristic arguments, and little theory is known to constitute a good guideline for the development of their architecture. Trying to remedy this situation, we introduce a new type of design for symmetric cryptographic primitives with high self-similarity. Our design strategy enables to give a reductionist security proof for the primitive based on plausible assumptions regarding the complexity of the best distinguishing attacks on random Feistel schemes or other ideal constructions. Under these assumptions, the cryptographic primitives we obtain are perfectly secure against any adversary with computational resources less than a given bound. By opposition, other provably secure symmetric primitives, as for example C [3] and KFC [4], designed using information-theoretic results, are only proved to resist a limited (though significant) range of attacks. Our construction strategy leads to a large expanded key size, though still usable in practice (around 1 MB).

Key words: block ciphers, Feistel schemes, generic attacks, provable security

1 Introduction

Provable security. Building provably secure but still efficient block ciphers is certainly the most desired but also the most challenging goal of symmetric cryptography. In the area of asymmetric cryptography, "provable security" means that one is mathematically able to reduce the security of a primitive to a well studied and presumably difficult problem such as integer factorisation or discrete logarithm (see [17] for an overview but also a critical look at "provable security" in public key cryptography). The situation in symmetric cryptography is quite different: the security of the most widely deployed primitives often relies on heuristic arguments of one of the three following types:

- lack of known attacks whose complexity is less than "brute-force" attacks or less than the desired security level (typically 2^{80} operations nowadays).

Appeared in SAC 2008.

- provable security against some classes of attacks, typically differential and linear cryptanalysis when dealing with block ciphers. For example, AES does possess such security arguments.
- provable security when some components of the primitive are replaced by "ideal" ones. This kind of arguments apply for example for all Feistel ciphers such as DES, for which the celebrated result of Luby and Rackoff [19] shows that when the internal functions are pseudorandom, the cipher is secure in the sense that it is a pseudorandom permutation. This, however, does not yield any security proof for the real primitive, but only ensures that the general structure of the algorithm does not present intrinsic weaknesses.

Provable security in symmetric cryptography in the reductionist sense discussed for asymmetric cryptography is rather rare. Most notable examples include some number-theoretic hash functions like VSH [10] and the stream cipher QUAD [6] whose security relies on the difficulty of solving systems of multivariate quadratic equations. However, there is to the best of our knowledge no block cipher with security reduction to some hard problem proposed so far. More concernedly, no difficult problems have been identified as suitable for such a design goal. We will see that the problem of distinguishing a Feistel scheme from a random permutation could be a potential candidate.

The proposal. We propose to build a block cipher whose security can be reduced to some simple and well studied problem. The hard problem we propose is not number-theoretic like for most schemes of asymmetric cryptography. We will use the problem of distinguishing a random Feistel scheme from a random permutation. The rational for such a choice is that Feistel schemes have been extensively studied in the cryptographic literature since the introduction of DES. Though most of this literature is primarily concerned with the information-theoretic properties of these schemes, some authors have studied the so-called "generic attacks" on them. The term generic attacks, introduced by Kilian and Rogaway in [16], means any attack performed on Feistel schemes instantiated using uniformly random and independent functions in each round (which we will name a "random Feistel scheme" in the following), and hence not making use of the underlying structure of the function generator of a real cipher such as DES. Though we will primarily use Feistel schemes, any well studied structure with similar properties could be used.

We propose to go beyond the intrinsic limitations of information-theoretic designs. For Feistel schemes, information theory is "stuck" at five rounds in the sense that increasing the number of rounds beyond five does not increase the number of queries needed by a computationally unbounded adversary to distinguish the Feistel scheme from a random permutation. Indeed, whatever the number $r \geq 5$ of rounds used in a random Feistel schemes from $2n$ bits to $2n$ bits, there is always an oracle adversary making $\Theta(2^n)$ queries and distinguishing a random Feistel scheme from a random permutation with high probability. However the *computational complexity* of this distinguisher can be extremely high. Taking the problem in the opposite way, we will make the hypothesis (and give arguments supporting it) that the best generic attacks described against Feistel schemes cannot be improved, and design a permutation generator such that any distinguishing attack against it would imply an improvement of the generic attacks against random Feistel schemes.

To achieve this goal, we will start from a Feistel scheme with r_1 rounds using random and independent functions at each round, and evaluate its security according to the best generic attacks. Then, rather than using independent and random functions directly as the key, we will instantiate each of these functions with independent Feistel schemes with r_2 rounds, and again estimate the security of the overall construction with respect to the best generic attacks. We will keep on using this recursive structure until the total size of the key (constituted of the random functions used at the innermost level of the construction) becomes practical. We name this design strategy the "Russian Dolls" construction. As we will see, the complexity of the best distinguisher described so far increases exponentially with the number of rounds of the Feistel scheme, so that using a reasonable number of rounds will be sufficient for a good level of security. Note that in the information-theoretic setting, the innermost Feistel schemes would be potentially weak as they have very small block size. However, any attack on the resulting block cipher would imply a better generic attack on random Feistel schemes at some level of the construction.

Related work. There have been a number of "provably secure" block ciphers proposals. We review the most prominent of them. BEAR and LION were proposed by Anderson and Biham [2]. They are constructed from an ideal stream cipher and an ideal hash function, and the authors proved that attacking the block cipher would imply an attack on one of the underlying components. Later Pat Morin [22] identified some weaknesses in BEAR and LION and proposed AARDVARK, which is based on the same design strategy.
Zheng, Matsumoto and Imai [36] presented block ciphers built on so-called Generalized Type-2 transformations (which are kinds of generalized Feistel constructions). They analysed their constructions in the information-theoretic setting and gave evidence supporting the security of their primitives, but no formal security proof.
Baignères and Finiasz built on Vaudenay's *decorrelation* theory [35] to propose two block ciphers, C [3] and KFC [4], provably secure against a wide range of attacks. This is the logical continuation of the work initiated with the NUT family [35] (COCONUT, PEANUT) and the AES proposal DFC [13]. Again, their security proof relies on information-theoretic arguments. In particular, KFC is based on a 3-round Feistel scheme using round functions with a very low decorrelation bias and is proved resistant against "d-limited" adversaries making less than $d = 8$ or 70 queries, depending on the parameters. The security proof also handles so-called "iterated attacks" of order $d/2$, where the adversary repeats independent non-adaptive $d/2$-limited attacks. However, we note that as the Feistel scheme of KFC has only 3 rounds, it is vulnerable to a distinguishing attack making only 3 chosen plaintext-ciphertext queries (see Section 4.2).

Appeared in SAC 2008.

Granboulan and Pornin [14] proposed an efficient way of generating perfectly random permutations (*i.e.* statistically very close to the uniform distribution, even for an attacker having the entire codebook) using a pseudorandom number generator, however their construction is only practical for small plaintext domains (typically less than 30-bit blocks).

The prior proposal which is the closest to our work was made by Blaze [7] but never published. He proposed the block cipher TURTLE and the stream cipher HAZE. TURTLE is simply the Russian Dolls construction where 4-rounds Feistel schemes are used at each stage, and HAZE is based on TURTLE in counter mode. Yet the security arguments proposed by Blaze are quite different from ours. He claims that retrieving the secret functions of an r-round Feistel scheme, $r \geq 3$, is NP-complete by reducing this problem to Numerical Matching with Target Sums (NMTS) [11]. However, keeping the number or rounds constant as the block-size decreases implies a dramatic loss of security.

Organization. Our paper is organized as follows. First we give our notations and some standard security definitions. Then, we describe the Russian Dolls design strategy in all generality and state theorems about its security. In Section 4 we analyse the Russian Dolls construction using balanced Feistel schemes. We highlight some promising possibilities for future work and draw our conclusions in Section 5.

2 Preliminaries

Notations. Throughout the whole paper, we will use the following notations. We will denote by $s \overset{\$}{\leftarrow} S$ the operation of selecting an element in the set S endowed with the uniform probability distribution. $\mathrm{Func}\left(\mathcal{D}, \mathcal{R}\right)$ will denote the set of all functions from $\mathcal{D}$ to $\mathcal{R}$, $\mathrm{Perm}\left(\mathcal{D}\right)$ the set of all permutations on $\mathcal{D}$, and $\mathrm{Perm}^{+}\left(\mathcal{D}\right)$ the set of all permutations on $\mathcal{D}$ with an even signature. I_n will denote the set of binary strings of length n, and we will use $\mathrm{Func}\left(n, m\right)$, $\mathrm{Perm}\left(n\right)$ and $\mathrm{Perm}^{+}\left(n\right)$ as shorthands for $\mathrm{Func}\left(I_n, I_m\right)$, $\mathrm{Perm}\left(I_n\right)$ and $\mathrm{Perm}^{+}\left(I_n\right)$ respectively.

A family of functions from $\mathcal{D}$ to $\mathcal{R}$ indexed by key space $\mathcal{K}$ is a function $E : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$. We will use the notation $E_K(X)$ as shorthand for $E(K, X)$. E is a family of permutations if $\mathcal{D} = \mathcal{R}$ and E_K is a permutation for each $K \in \mathcal{K}$. We will denote by E_K^{-1} the inverse of E_K. We will sometimes use the terms function or permutation generator instead of family of functions or permutations.

Given a function f of $\mathrm{Func}\left(n, n\right)$, the 1-round Feistel scheme Ψ_f is the element of $\mathrm{Perm}\left(2n\right)$ defined by $\Psi_f(x) = x_{\mathrm{R}} \| x_{\mathrm{L}} \oplus f(x_{\mathrm{R}})$, where x_{L} and x_{R} denote respectively the left and right halves of the $2n$-bit string x. We will note $\Psi_{f_1,\ldots,f_r}$ the r-rounds Feistel scheme $\Psi_{f_r} \circ \ldots \circ \Psi_{f_1}$. Given two non null integers n and r, $\Psi^{(r)}(2n)$ will denote the permutation generator on I_{2n} with key space $\mathrm{Func}\left(n, n\right)^r$, taking as arguments r functions $(f_1, \ldots, f_r)$ in $\mathrm{Func}\left(n, n\right)$ and $x \in I_{2n}$ and returning $\Psi_{f_1,\ldots,f_r}(x)$. When we omit the block-size, *i.e.* $\Psi^{(r)}$, it will implicitly be $2n$.

Appeared in SAC 2008.

The adversaries we will consider are probabilistic. Implicitly, when we note $\Pr[s \xleftarrow{\$} S : \mathcal{A} = 1]$ the probability will always be on S *and* the internal randomness of $\mathcal{A}$.

Pseudorandom functions and permutations. The notion of pseudorandom function (PRF) was introduced by [12], and the notion of pseudorandom and strong (or super-) pseudorandom permutation (PRP and SPRP) by [18]. Informally, a PRF is a family of functions E indexed by a key space $\mathcal{K}$ such that any efficient adversary with access to an oracle can distinguish a function associated to a random key $K \xleftarrow{\$} \mathcal{K}$ from a uniformly random function only with negligible probability. The definition of a PRP is quite similar, except that the adversary tries to distinguish the permutation family from a uniformly random permutation. For a SPRP, the adversary is given access to two oracles, either E_K and E_K^{-1} for a random K, or G and G^{-1} for a uniformly random permutation G. Rather than using the usual asymptotic notions of PRF and PRP, we will use the concrete security approach introduced in [5] where the distinguishing advantage of an adversary is measured as a function of its resources (namely, runtime and number of oracle queries). We give now the following formal definitions.

Definition 1 (PRF). *Let $E : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ be a family of functions from $\mathcal{D}$ to $\mathcal{R}$ indexed by keys $\mathcal{K}$. An adversary $\mathcal{A}$ (ϵ, T)-distinguishes E as a PRF if it runs in time at most T and*

$$\mathrm{Adv}_E^{\mathrm{prf}}(\mathcal{A}) = \Big| \Pr\Big[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{E_K} = 1\Big] \\ - \Pr\Big[G \xleftarrow{\$} \mathrm{Func}\,(\mathcal{D}, \mathcal{R}) : \mathcal{A}^{G} = 1\Big] \Big| \geq \epsilon \ .$$

We will say that E is an (ϵ, T)-secure PRF if no adversary is able to (ϵ, T)-distinguish it.

Definition 2 (PRP). *Let $E : \mathcal{K} \times \mathcal{D} \to \mathcal{D}$ be a family of permutations on $\mathcal{D}$ indexed by keys $\mathcal{K}$. An adversary $\mathcal{A}$ (ϵ, T)-distinguishes E as a PRP if it runs in time at most T and*

$$\mathrm{Adv}_E^{\mathrm{prp}}(\mathcal{A}) = \Big| \Pr\Big[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{E_K} = 1\Big] \\ - \Pr\Big[G \xleftarrow{\$} \mathrm{Perm}\,(\mathcal{D}) : \mathcal{A}^{G} = 1\Big] \Big| \geq \epsilon \ .$$

We will say that E is an (ϵ, T)-secure PRP if no adversary is able to (ϵ, T)-distinguish it.

Definition 3 (SPRP). *Let $E : \mathcal{K} \times \mathcal{D} \to \mathcal{D}$ be a family of permutations on $\mathcal{D}$ indexed by keys $\mathcal{K}$. An adversary $\mathcal{A}$ (ϵ, T)-distinguishes E as a SPRP if it runs in time at most T and*

$$\mathrm{Adv}_E^{\mathrm{sprp}}(\mathcal{A}) = \Big| \Pr\Big[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{E_K, E_K^{-1}} = 1\Big] \\ - \Pr\Big[G \xleftarrow{\$} \mathrm{Perm}\,(\mathcal{D}) : \mathcal{A}^{G, G^{-1}} = 1\Big] \Big| \geq \epsilon \ .$$

Appeared in SAC 2008.

We will say that E is an (ϵ, T)-secure SPRP if no adversary is able to (ϵ, T)-distinguish it.

Alternatively, when a primitive is $(O(\frac{T}{f(n)}), T)$-secure for some parameter n, where O stands for some small constant independent of n, we will say that it is $\Omega(f(n))$-secure, meaning that a distinguisher must have runtime greater than $f(n)$ to have a non-negligible advantage. Note that all our definitions are stated in terms of runtime T of the adversary. The total number q of queries of the adversary to the oracle will only be constrained by the obvious inequality $q \leq T$.

As we will see later, it is always possible to distinguish a random Feistel scheme $\Psi^{(r)}(2n)$ from a uniformly random permutation with complexity $O(2^{2n})$. This comes from the fact that a Feistel scheme has always an even signature, whereas a random permutation has an even signature with probability 1/2. We will therefore sometimes consider the difficulty of distinguishing a random Feistel scheme from a random permutation with an even signature. For this reason we also define the notion of (S)PRP$^+$ (strong pseudorandom *even* permutation) by simply substituting $\mathrm{Perm}^+(\mathcal{D})$ to $\mathrm{Perm}(\mathcal{D})$ in the definitions of PRP and SPRP.

We will use sometimes the term CPA (Chosen Plaintext Attack) to qualify an adversary trying to break the pseudorandomness of a permutation generator, and CPCA (Chosen Plaintext-Ciphertext Attack) to qualify an adversary trying to break the strong pseudorandomness of a permutation generator. It will always imply *adaptive* attacks.

3 The Russian Dolls Construction

In this section we explain our design strategy in all generality. Assume one knows how to construct a secure (S)PRP E on $\mathcal{D}$ using a relatively large set of keys $\mathcal{K}$ structured as a direct product of smaller permutations spaces $\mathcal{K} = \mathrm{Perm}(\mathcal{D}_1) \times \ldots \times \mathrm{Perm}(\mathcal{D}_\lambda)$. Assume now that there exists secure PRPs $E^{(i)}$, $1 \leq i \leq \lambda$, on $\mathcal{D}_i$ with key spaces $\mathcal{K}_i$. Then it is possible to define a new (S)PRP E' on $\mathcal{D}$ with key space $\mathcal{K}' = \mathcal{K}_1 \times \ldots \times \mathcal{K}_\lambda$, by

$$E'_{(K_1,\ldots,K_\lambda)}(\cdot) = E_{(E^{(1)}_{K_1},\ldots,E^{(\lambda)}_{K_\lambda})}(\cdot) \quad . \tag{1}$$

For simplicity, we will make the assumption that when the $E^{(i)}$'s are given as oracles, ciphering or deciphering with E' requires only *direct* queries to the $E^{(i)}$'s. As will be clear from the proof of the theorem below, this enables to use only secure PRPs for the $E^{(i)}$'s. As soon as it requires access to the direct and the inverse oracle for some i, $E^{(i)}$ has to be a secure SPRP. The security of the new (S)PRP E' is characterized by the following theorem:

Theorem 1 (Security of the Russian Dolls construction). *Let E be an (ϵ, T)-secure PRP (resp. SPRP) on $\mathcal{D}$ indexed by key space $\mathcal{K} = \mathrm{Perm}(\mathcal{D}_1) \times \ldots \times \mathrm{Perm}(\mathcal{D}_\lambda)$. Let also $E^{(i)}$, $1 \leq i \leq \lambda$, be (ϵ_i, T)-secure PRPs on $\mathcal{D}_i$ with key spaces $\mathcal{K}_i$. Then the permutation generator E' defined by Equ. 1 is an $(\epsilon + \sum_{i=1}^{\lambda} \epsilon_i, T)$-secure PRP (resp. SPRP) on $\mathcal{D}$ with key space $\mathcal{K}' = \mathcal{K}_1 \times \ldots \times \mathcal{K}_\lambda$.*

Proof. The proof proceeds by a standard hybrid method. Let $\mathcal{A}$ be an oracle algorithm running in time T. We are interested in bounding its advantage in distinguishing the PRP E':

$$\Bigg| \Pr\left[(K_1,\ldots,K_\lambda) \xleftarrow{\$} \mathcal{K}_1 \times \ldots \times \mathcal{K}_\lambda \,:\, \mathcal{A}^{E_{(E^{(1)}_{K_1},\ldots,E^{(\lambda)}_{K_\lambda})}} = 1\right] \\ - \Pr\left[G \xleftarrow{\$} \mathrm{Perm}\,(\mathcal{D}) \,:\, \mathcal{A}^G = 1\right] \Bigg| \;.$$

This advantage is upper bounded through the triangular inequality by the sum of

$$\Bigg| \Pr\left[(G_1,\ldots,G_\lambda) \xleftarrow{\$} \mathcal{K} \,:\, \mathcal{A}^{E_{(G_1,\ldots,G_\lambda)}} = 1\right] \\ - \Pr\left[G \xleftarrow{\$} \mathrm{Perm}\,(\mathcal{D}) \,:\, \mathcal{A}^G = 1\right] \Bigg|$$

and the sum for $i = 1$ to λ of the following quantities (where by convention for $i = 1$ (resp. $i = \lambda$), the expressions were $i-1$ (resp. $i+1$) appears are discarded):

$$\Bigg| \Pr\Big[\, (K_1,\ldots,K_i) \xleftarrow{\$} \mathcal{K}_1 \times \ldots \times \mathcal{K}_i, \\ (G_{i+1},\ldots,G_\lambda) \xleftarrow{\$} \mathrm{Perm}\,(\mathcal{D}_{i+1}) \times \ldots \times \mathrm{Perm}\,(\mathcal{D}_\lambda) \;: \\ \mathcal{A}^{E_{(E^{(1)}_{K_1},\ldots,E^{(i)}_{K_i},G_{i+1},\ldots,G_\lambda)}} = 1\Big] \\ - \Pr\Big[\, (K_1,\ldots,K_{i-1}) \xleftarrow{\$} \mathcal{K}_1 \times \ldots \times \mathcal{K}_{i-1}, \\ (G_i,\ldots,G_\lambda) \xleftarrow{\$} \mathrm{Perm}\,(\mathcal{D}_i) \times \ldots \times \mathrm{Perm}\,(\mathcal{D}_\lambda) \;: \\ \mathcal{A}^{E_{(E^{(1)}_{K_1},\ldots,E^{(i-1)}_{K_{i-1}},G_i,\ldots,G_\lambda)}} = 1\Big] \Bigg|$$

The first term is upper bounded by definition by ϵ as E is an (ϵ, T)-secure PRP. The i-th of the λ other terms is upper bounded by ϵ_i. Indeed, one can build a probabilistic distinguisher $\mathcal{A}_i$ for $E^{(i)}$ as follows. Let F be the oracle to which $\mathcal{A}_i$ has access. $\mathcal{A}_i$ draws random keys $(K_1,\ldots,K_{i-1})$ and random permutations $(G_{i+1},\ldots,G_\lambda)$ and runs $\mathcal{A}$, answering each of its queries with $E_{(E^{(1)}_{K_1},\ldots,E^{(i-1)}_{K_{i-1}},F,G_{i+1},\ldots,G_\lambda)}$. Then $\mathcal{A}_i$ runs in time T and its advantage is exactly the quantity above. Hence by hypothesis on $E^{(i)}$ it cannot be greater than ϵ_i. The theorem follows. The SPRP case is handled in a similar way. □

More restricted versions of this theorem in the information-theoretic setting can be found in [20, Theorem 1] and [35, Lemma 20]. When the key spaces $\mathcal{K}_i$ are themselves permutations spaces, the construction can be iterated to decrease the key size of the outermost PRP. This construction may use functions instead of permutations or even a mix of functions and permutations. However, we will be primarily interested in permutations. We will now see how to use the Russian Dolls construction with concrete PRP schemes.

4 Constructions with Balanced Feistel Schemes

Two main lines of research have been explored concerning Feistel schemes: one aims at giving security bounds against information-theoretic adversaries, the other tries to describe generic attacks on random Feistel schemes. We sum up some known results about these two domains.

4.1 Information-theoretic Bounds

First, we review the security results on random Feistel schemes holding in the information-theoretic setting, *i.e.* against computationally unbounded adversaries. All these results are purely combinatorial and can be restated in terms of *statistical closeness* between the output of a Feistel permutation and the output of a uniformly random permutation. Though we restate them in terms of computational runtime T, it is essential to note that they are in fact all true in terms of number of oracle queries q. The computational statement simply stems from $q \leq T$.

Luby and Rackoff started the subject by proving [19] that $\Psi^{(3)}(2n)$ is a $\Omega(2^{\frac{n}{2}})$-secure PRP, and claiming (without proof) that $\Psi^{(4)}(2n)$ is a $\Omega(2^{\frac{n}{2}})$-secure SPRP. The later was proved by Patarin in [23]. The first improvements beyond the so-called "birthday bound" (namely, $\Omega(2^{\frac{n}{2}})$-security) came from Patarin who proved respectively in [25] and [26] that $\Psi^{(5)}$ is a $\Omega(2^{\frac{2n}{3}})$-secure PRP and $\Psi^{(6)}$ is a $\Omega(2^{\frac{3n}{4}})$-secure PRP. Maurer and Pietrzak showed [21] that for r sufficiently large, $\Psi^{(r)}$ is a $\Omega(2^{(1-O(\frac{1}{r}))n})$-secure SPRP. Finally, Patarin proved in [28,29] that the information-theoretic optimal security is obtained for 5 rounds in a CPA attack (*i.e.* $\Psi^{(5)}$ is a $\Omega(2^n)$-secure PRP) and 6 rounds for a CPCA attack (*i.e.* $\Psi^{(6)}$ is a $\Omega(2^n)$-secure SPRP). It is still an open problem to improve the bound for $\Psi^{(5)}$ in a CPCA attack (for now it is only known that $\Psi^{(5)}$ is a $\Omega(2^{\frac{n}{2}})$-secure SPRP).

However, building on these results doesn't enable to construct secure schemes using the Russian Dolls construction as the security decreases with the block size. We will see in the following how we can circumvent this problem by making hypotheses on the best generic attacks on random Feistel schemes.

4.2 Generic Attacks on Feistel Schemes

Generic attacks on $\Psi^{(3)}$ and $\Psi^{(4)}$. Generic attacks on $\Psi^{(3)}$ and $\Psi^{(4)}$ matching the information-theoretic security bounds were described in [25] and later independently in [1]. In the 3-round case, for a CPA attack, the adversary gets m values $y_i = E(x_i)$ and counts the number of (i,j), $i < j$, such that $x_{i\mathrm{R}} \oplus y_{i\mathrm{L}} = x_{j\mathrm{R}} \oplus y_{j\mathrm{L}}$. It can be proved that this number will be about twice greater in the case of $\Psi^{(3)}$ than for a random permutation, and this leads to an attack with $O(2^{\frac{n}{2}})$ queries and runtime. However, there is a very efficient CPCA attack with only 3 queries: $\mathcal{A}$ asks for $y_1 = E(x_1)$ and $y_2 = E(x_2)$ where $x_{1\mathrm{R}} = x_{2\mathrm{R}}$. Then, it asks for $x_3 = E^{-1}(y_{2\mathrm{L}} || y_{2\mathrm{R}} \oplus x_{1\mathrm{L}} \oplus x_{2\mathrm{L}})$ and checks whether

$x_{3\mathrm{R}} = x_{1\mathrm{R}} \oplus y_{1\mathrm{L}} \oplus y_{2\mathrm{L}}$. This will always be the case for $\Psi^{(3)}$ but will happen only with probability $1/2^n$ for a random permutation. We note that this attack applies to KFC as it is based on a 3-round Feistel scheme. However KFC was explicitly designed to resist only chosen-plaintext attacks.

In the 4-round case, there is the following CPA attack: the adversary gets m values $y_i = E(x_i)$ such that $x_{i\mathrm{R}}$ is constant and counts the number of (i, j), $i < j$, such that $x_{i\mathrm{L}} \oplus y_{i\mathrm{L}} = x_{j\mathrm{L}} \oplus y_{j\mathrm{L}}$. Again, it can be proved that this number will be about twice greater in the case of $\Psi^{(4)}$ than for a random permutation, and this leads to an attack with $O(2^{\frac{n}{2}})$ queries and runtime.

Brute force attacks. We state the following result concerning brute force attacks on Feistel schemes, valid for any number of rounds.

Claim. Let r, n be non null integers, r fixed. Then there exists an oracle adversary, running in time $\Theta(2^{rn2^n})$ and distinguishing $\Psi^{(r)}(2n)$ from a random permutation with overwhelming probability.

A rigorous proof of this claim can be found in [24]. Note that a simple entropy argument [21, footnote 2] shows that the number of oracle queries required is only $r \cdot 2^n$, which is in $O(2^n)$ for any fixed r. The adversary proceeds by making an exhaustive search on the key space $\mathrm{Func}\,(n, n)^r$ to see if there is one for which all queries match. It is however highly non trivial to reduce the complexity of the distinguisher described in the above claim in the case $r \geq 5$, as we will see now.

Attacks "by the signature". As noticed by Patarin in [27], there are better attacks than the exhaustive search described above taking advantage of the fact that Feistel schemes lie in a proper subgroup of $\mathrm{Perm}\,(2n)$, namely $\mathrm{Perm}^+\,(2n)$. Indeed, it can easily be checked (see [27]) that a Feistel scheme has always an even signature. Clearly, the signature of a permutation $E \in \mathrm{Perm}\,(2n)$ can be computed in time $O(2^{2n})$ when all the cipherbook is available. As a random permutation has an even signature with probability $\frac{1}{2}$, we have the following claim:

Claim. Let r, n be non null integers. Then there exists an oracle adversary, running in time $\Theta(2^{2n})$ and distinguishing $\Psi^{(r)}(2n)$ from a uniformly random permutation with probability $\frac{1}{2}$.

However, as we will see in the following, it is much harder to distinguish $\Psi^{(r)}$ when this "global" property is suppressed, *i.e.* when the adversary tries to distinguish $\Psi^{(r)}$ from a random permutation with an even signature.

Best known attacks against $\boldsymbol{\Psi^{(r)}}$ as an SPRP$^+$ when $\boldsymbol{r \geq 5}$. The best generic attacks for distinguishing $\Psi^{(r)}$ from a random even permutation fall in the class of iterated attacks of order 2. The notion of iterated distinguisher of order d has been defined by Vaudenay [34,35]. Roughly, such a distinguisher

```
Parameters: number of iterations $N$, decision function $\Gamma : \mathcal{D}^d \times \mathcal{D}^d \to \{0,1\}$,
acceptance set $S \subset \{0,1\}^N$
Oracle: a permutation $E \in \mathrm{Perm}(\mathcal{D})$ (and possibly its inverse $E^{-1}$)

for $i = 1$ to $N$ do
    for $j = 1$ to $d$ do
        select $x_j \in \mathcal{D}$ and get $y_j = E(x_j)$ or select $y_j \in \mathcal{D}$ and get $x_j = E^{-1}(y_j)$
    end for
    set $\gamma_i = \Gamma(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{x} = (x_1, \ldots, x_d)$ and $\boldsymbol{y} = (y_1, \ldots, y_d)$
end for
if $(\gamma_1, \ldots, \gamma_N) \in S$ then output 1 else output 0
```

Fig. 1. Iterated attack of order d

obtains a number d of plaintext-ciphertext pairs (x_j, y_j), takes a binary decision γ_i depending on $\boldsymbol{x} = (x_1, \ldots, x_d)$ and $\boldsymbol{y} = (y_1, \ldots, y_d)$, and after N repetitions of this, outputs 0 or 1 depending on $(\gamma_1, \ldots, \gamma_N)$. At each iteration i, the d-tuple of plaintext-ciphertext pairs that is tested is determined, possibly adaptively, and possibly in a probabilistic way[1] by the adversary, by making only queries to E for a CPA attack, or to E and E^{-1} for a CPCA attack. It is important however that the decision function Γ such that $\gamma_i = \Gamma(\boldsymbol{x}, \boldsymbol{y})$ is fixed during all the attack. In particular, it must not depend on the previously tested d-tuples and previous decisions. Indeed, if it were the case, the i-th decision γ_i of the adversary would in fact depend on all previous d-tuples already tested and the distinguisher would in fact be a classical d'-limited adversary with $d' > d$. Note that this is only a logical description. In particular the total runtime of the adversary can be less than N. For example, the generic attack described previously on $\Psi^{(4)}$ is an iterated attack of order 2 where the attacker makes $N = m(m-1)$ tests in time m by storing the m values of $x_{i\mathrm{L}} \oplus y_{i\mathrm{L}}$ and counting the number of collisions. The total runtime of the adversary is thus $T = \sqrt{N}$. It is evident that making the same test more than one time does not increase the advantage of the adversary, hence we will assume that the distinguisher never makes twice the same test. Thus, the total number of possible tests is $2^{2n}(2^{2n}-1)\cdots(2^{2n}-d+1)$. Note that the outcomes of the tests are of course not independent.

Up to now, the best distinguishing attacks on Feistel schemes with $r \geq 5$ rounds, described in [28], are iterated attacks of order 2. They follow the general description of Fig. 1. We describe the case r even; the case r odd is handled in a similar way. The attacks need only to access the direct oracle E. To understand how these attacks work, we introduce the d-ary transition probabilities associated to a permutation generator E on $\mathcal{D}$ with key space $\mathcal{K}$ defined for any pairs of d-tuples $\boldsymbol{x} = (x_1, \ldots, x_d)$, $\boldsymbol{y} = (y_1, \ldots, y_d)$ of distinct elements of $\mathcal{D}$ by

$$\Pr[\boldsymbol{x} \xrightarrow{E_K} \boldsymbol{y}] = \Pr\left[K \overset{\$}{\leftarrow} \mathcal{K} \,:\, E_K(x_i) = y_i \text{ for all } i \in [1..d]\right] \quad . \tag{2}$$

[1] Indeed, as we consider computationally bounded adversaries, there may be an advantage for the adversary to be probabilistic.

These quantities were introduced and extensively studied by Patarin in [24,23] and are fundamental in upper bounding the advantage of information-theoretic adversaries making less than d queries and trying to distinguish E_K from a uniformly random permutation on $\mathcal{D}$. In particular, closed formula were given in the binary case $d = 2$, for any number of rounds r. Let $\Pr^* = \frac{1}{2^{2n}(2^{2n}-1)}$ denote the binary transition probability for a random even permutation for any $\boldsymbol{x}$ and $\boldsymbol{y}$. We will simply note Pr for $\Pr[\boldsymbol{x} \xrightarrow{\Psi^{(r)}} \boldsymbol{y}]$. For r even, when $x_{1\mathrm{R}} = x_{2\mathrm{R}}$, then depending on (y_1, y_2) the transition probabilities have the following values:

1. when $y_{1\mathrm{L}} = y_{2\mathrm{L}}$, $\Pr = \Pr^* \left(1 - \frac{1}{2^{(r-2)n}}\right)$
2. when $y_{1\mathrm{L}} \neq y_{2\mathrm{L}}$ and $x_{1\mathrm{L}} \oplus y_{1\mathrm{L}} \neq x_{2\mathrm{L}} \oplus y_{2\mathrm{L}}$, $\Pr \simeq \Pr^* \left(1 - \frac{1}{2^{\left(\frac{r}{2}-1\right)n}}\right)$
3. when $y_{1\mathrm{L}} \neq y_{2\mathrm{L}}$ and $x_{1\mathrm{L}} \oplus y_{1\mathrm{L}} = x_{2\mathrm{L}} \oplus y_{2\mathrm{L}}$, $\Pr \simeq \Pr^* \left(1 + \frac{1}{2^{\left(\frac{r}{2}-2\right)n}}\right)$

With these notations the attack proceeds as follows. The adversary tests N pairs (x_1, y_1), (x_2, y_2) such that $x_{1\mathrm{R}} = x_{2\mathrm{R}}$. The decision function is defined by

$$\Gamma(\boldsymbol{x}, \boldsymbol{y}) = \begin{cases} 0 \text{ if } \Pr \leq \Pr^* & \text{(cases 1 and 2)} \\ 1 \text{ if } \Pr > \Pr^* & \text{(case 3)} \end{cases}$$

Let X be the random variable defined by $X = \sum_{i=1}^{N} \gamma_i$. Let $E(X)$ and $\sigma(X)$ (resp. $E^*(X)$ and $\sigma^*(X)$) be the expected value and the standard deviation of X for a random Feistel scheme (resp. a random even permutation). One can easily check that $E^*(X) \simeq \frac{N}{2^n}$ and $E(X) \simeq \frac{N}{2^n}\left(1 + \frac{1}{2^{\left(\frac{r}{2}-2\right)n}}\right)$, and it can be proved that $\sigma^*(X) \simeq \frac{\sqrt{N}}{2^{\frac{n}{2}}}$ and $\sigma(X) \simeq \frac{\sqrt{N}}{2^{\frac{n}{2}}}$. If we let the acceptance set be $S = \{(\gamma_1, \ldots, \gamma_N) \mid \sum_{i=1}^{N} \gamma_i \geq \tau\}$ for $\tau = (E(X) - E^*(X))/2$, the adversary will have a noticeable advantage as soon as τ is larger than $\sigma(X)$ and $\sigma^*(X)$. This implies the condition $N \geq 2^{(r-3)n}$.

Because of the constraint $x_{1\mathrm{R}} = x_{2\mathrm{R}}$, the number of possible tests is only 2^{3n}. So in order to have a meaningful attack for $r \geq 7$ we have to broaden slightly the security model by letting the adversary interact with $\mu > 1$ permutations randomly outputted by the generator. The adversary will have to repeat the test on $\mu = 2^{(r-6)n}$ permutations. For each permutation, the 2^{3n} tests can in fact be implemented in time 2^{2n} by building, for each possible value of x_{R}, the list of the 2^n values for $x_{i\mathrm{L}} \oplus y_{i\mathrm{L}}$ and counting the number of collisions. Hence the total runtime of $\mathcal{A}$ is T$=\mu 2^{2n} = 2^{(r-4)n}$. Note that originally Patarin [28] described a known plaintext attack with roughly the same complexity.

We will take these best known generic attacks as a starting point to build secure PRPs by making the following conjecture:

Conjecture 1. Let $n > 1$ be an integer, r be an integer ≥ 5. Then $\Psi^{(r)}(2n)$ is a $(O(\frac{T}{2^{(r-4)n}}), T)$-secure SPRP$^+$.

Evidence in favour of this conjecture is that the best distinguishing attacks for 3 and 4 rounds, matching the information-theoretic bounds, are iterated attacks of order 2. Hence this conjecture may be viewed as a natural generalization to $r \geq 5$ of a provable result for $r < 5$. We also conjecture that for a fixed d, iterated attacks of order d are not more efficient than the best iterated attack of order 2 for sufficiently large n. Hence improving the attacks described above would require to handle large d-tuples of plaintext-ciphertext pairs, which appears to be intractable as the computation of the transition probabilities for random Feistel schemes becomes very involved as soon as $d \geq 3$.

4.3 The Russian Dolls Construction with Balanced Feistel Schemes

We now concretely describe how to construct a secure SPRP using the Russian Dolls construction and Conjecture 1. The parameters of the construction will be as follows:

- the block size of the SPRP will be $2n$,
- s will denote the number of iterations of the Russian Dolls construction,
- $r_1, r_2, \ldots, r_s =$ will denote the number of rounds of the Feistel schemes used at the i-th iteration of the process.

We start with the outermost Feistel scheme, which will have r_1 rounds. If it were to be instantiated with r_1 random functions, the obtained permutation generator would be a $(O(\frac{T}{2^{(r_1-4)n}}), T)$-secure SPRP$^+$. However, the size of the key would be $r_1 n 2^n$ bits, which is impractical for usual values of n. Using the Russian Dolls construction, one can decrease the size of the key while maintaining a good level of security by instantiating each function inside the Feistel scheme $\Psi^{(r_1)}$ with independent Feistel schemes with r_2 rounds. Again, each function used in the r_1 Feistel schemes $\Psi^{(r_2)}$ can be instantiated using independent Feistel schemes with r_3 rounds, and so on... Note that we implicitly make here the assumption that the security of a Feistel scheme with internal random permutations is close to the security obtained when using internal random functions. A security proof by Piret [33] as well as preliminary results on generic attacks on Feistel schemes with internal permutations [32] point towards the validity of this assumption.

Consider the permutation generator obtained after s iterations of the nesting process. The innermost Feistel schemes use random functions from $\frac{n}{2^{s-1}}$ bits to $\frac{n}{2^{s-1}}$ bits which will constitute the key for the global permutation generator. It can easily be seen that the total number of functions needed to define the global permutation is $r_1 \cdot r_2 \ldots r_s$. Hence the size of the key defining a permutation is

$$\log_2(|\mathcal{K}|) = r_1 \cdot r_2 \cdots r_s \cdot \frac{n}{2^{s-1}} \cdot 2^{\frac{n}{2^{s-1}}} \quad .$$

Suppose now that the numbers of rounds r_i were chosen as the minimal integers to satisfy, for some α, the following inequality:

$$(r_i - 4)\frac{n}{2^{i-1}} \geq \alpha \quad i.e. \quad r_i = \left\lceil \frac{2^{i-1}\alpha}{n} + 4 \right\rceil \quad . \qquad (3)$$

Appeared in SAC 2008.

According to Conjecture 1, any Feistel scheme used in the construction is a $(\frac{T}{2^\alpha}, T)$-secure SPRP. Then, according to Theorem 1, any adversary running in time T and trying to distinguish a permutation resulting from the overall construction from a uniformly random even permutation has an advantage upper bounded by

$$\left(\frac{T}{2^\alpha} + r_1\left(\frac{T}{2^\alpha} + r_2\left(\ldots\left(\frac{T}{2^\alpha} + r_s \cdot \frac{T}{2^\alpha}\right)\ldots\right)\right)\right) = \left(1 + \sum_{i=1}^{s}\prod_{j=1}^{i} r_j\right)\frac{T}{2^\alpha} \ .$$

Suppose that n is a power of 2. From an asymptotic point of view, if we set $\alpha = \text{poly}(n)$, Equation 3 shows that for a logarithmic number on iterations $s = \log_2(n) - c$, for some constant c, (which means that the key is constituted of functions from 2^{c+1} bits to 2^{c+1} bits), the numbers of rounds r_i will all be polynomials in n. Hence the size of the key will be in $\text{poly}(n)^{\log n} = e^{O((\log n)^2)}$, which is quasi-polynomial, whereas the security is in $(e^{O((\log n)^2)} \frac{T}{2^{\text{poly}(n)}}, T)$. So the Russian Dolls construction will be quite efficient *and* secure.

In practice, the optimal number of iterations is determined the following way. Assume that s iterations have been made, and we want to know whether the following iteration will increase or decrease the size of the key (we suppose that the loss of security coming from the next iteration is negligible). Up to now, the number of bits needed to store one of the functions constituting the key is $\frac{n}{2^{s-1}} \cdot 2^{\frac{n}{2^{s-1}}}$. Iterating the construction one more time would require to instantiate each of these functions with Feistel schemes with r_{s+1} rounds, where r_{s+1} verifies Equ. 3. Hence the storage requirements for each function would become $r_{s+1} \cdot \frac{n}{2^s} \cdot 2^{\frac{n}{2^s}}$. Consequently, it is unfavourable to iterate again as soon as

$$r_{s+1} \cdot \frac{n}{2^s} \cdot 2^{\frac{n}{2^s}} \geq \frac{n}{2^{s-1}} \cdot 2^{\frac{n}{2^{s-1}}}, \quad \textit{i.e.} \quad r_{s+1} \geq 2^{\frac{n}{2^s}+1} \ .$$

4.4 Concrete Instantiations

We give now some concrete values for the parameters (n, s, r_i). We describe a block cipher with 128-bit blocks, hence $n = 64$. We aim roughly at 80-bit security, meaning that the cipher has to be a $(T/2^{80}, T)$-secure SPRP. After some optimizations, one can verify that $s = 5$ iterations, with the following number of rounds: $r_1 = 6$, $r_2 = 7$, $r_3 = 10$, $r_4 = 16$ and $r_5 = 28$, is optimal and gives the desired level of security. The size of the expanded key, constituted of functions from 4 bits to 4 bits, is

$$\log_2(|\mathcal{K}|) = 6 \times 7 \times 10 \times 16 \times 28 \times 4 \times 2^4 \simeq 1.5 \text{ MB} \ ,$$

which is quite practical. Note however that stopping at $s = 4$ iterations (with the same number of rounds r_1 to r_4) yields an expanded key size of $\simeq$ 1.7 MB, which is close to the previous size. Yet the resulting block cipher would be much faster as the number of table accesses to encrypt or decrypt one plaintext would only be $6 \times 7 \times 10 \times 16 = 6,720$ instead of $6 \times 7 \times 10 \times 16 \times 28 = 188,160$, which shows that trade-offs are possible.

Key schedule. It is arguable that such a block cipher as we just described would be implemented using pseudorandom bits for the expanded key. We did not consider this problem in details and expect that a provably secure pseudorandom number generator, such as BBS [8] or QUAD [6] would be used to expand a smaller key. It may even be possible to design a key expansion procedure relying itself on the Russian Dolls construction with PRFs rather than PRPs. Besides, we'd like to underline that the nonexistence of short keys may be turned into an advantage in some cases, particularly in a white-box context of operation [9]. We leave this as topics for further research.

5 Conclusion and further work

We described a general recursive strategy enabling to build secure PRFs or PRPs and applied this design approach with random balanced Feistel schemes in order to obtain symmetric primitives provably secure under plausible conjectures about generic attacks on random Feistel schemes. The schemes we obtain look very promising: the size of the expanded key required for our proposed constructions is of the order of 1 MB, and hence compares very favorably with other proposals of provably secure block ciphers such as KFC which may require in extreme cases up to 4 GB of expanded key. Moreover our schemes should be very fast in software as they require only XOR operations and table look-ups.

Other structures are potentially very interesting to use inside the Russian Dolls construction. In the case of PRP constructions, unbalanced Feistel schemes could be suitable. They have been studied in [15,30,31] and could lead to expanded key size savings and efficiency improvements. Such schemes are currently under investigation.

Finally, proving results in the vein of Conjecture 1 may be very difficult because of its connexions with the "P vs. NP" problem. However it may be possible to obtain more restricted security results by considering weaker models of adversary (such as iterated attacks of order d). Such results would greatly reinforce the confidence in the primitives based on the Russian Dolls construction. Exploring new kinds of attacks on random Feistel schemes (*e.g.*, by studying the cycle structure of the permutation) might also be a fruitful avenue of research.

References

1. William Aiello and Ramarathnam Venkatesan, *Foiling Birthday Attacks in Length-Doubling Transformations - Benes: A Non-Reversible Alternative to Feistel*, Advances in Cryptology - EUROCRYPT 1996 (Ueli M. Maurer, ed.), Lecture Notes in Computer Science, vol. 1070, Springer, 1996, pp. 307–320.
2. Ross J. Anderson and Eli Biham, *Two Practical and Provably Secure Block Ciphers: BEAR and LION*, Fast Software Encryption (Dieter Gollmann, ed.), Lecture Notes in Computer Science, vol. 1039, Springer, 1996, pp. 113–120.
3. Thomas Baignères and Matthieu Finiasz, *Dial C for Cipher*, Selected Areas in Cryptography - SAC 2006 (E. Biham and A.M. Youssef, eds.), Lecture Notes in Computer Science, vol. 4356, Springer, 2006, pp. 76–95.

4. ______, *KFC - The Krazy Feistel Cipher*, Advances in Cryptology - ASIACRYPT 2006 (Xuejia Lai and Kefei Chen, eds.), Lecture Notes in Computer Science, vol. 4284, Springer, 2006, pp. 380–395.
5. Mihir Bellare, Joe Kilian, and Phillip Rogaway, *The Security of the Cipher Block Chaining Message Authentication Code*, J. Comput. Syst. Sci. **61** (2000), no. 3, 362–399.
6. Côme Berbain, Henri Gilbert, and Jacques Patarin, *QUAD: A Practical Stream Cipher with Provable Security*, Advances in Cryptology - EUROCRYPT 2006 (Serge Vaudenay, ed.), Lecture Notes in Computer Science, vol. 4004, Springer, 2006, pp. 109–128.
7. Matt Blaze, *Efficient Symmetric-Key Ciphers Based on an NP-Complete Subproblem*, 1996, available at http://www.crypto.com/papers/turtle.pdf.
8. Lenore Blum, Manuel Blum, and Mike Shub, *A Simple Unpredictable Pseudo-Random Number Generator*, SIAM J. Comput. **15** (1986), no. 2, 364–383.
9. Stanley Chow, Philip A. Eisen, Harold Johnson, and Paul C. van Oorschot, *White-Box Cryptography and an AES Implementation*, Selected Areas in Cryptography - SAC 2002 (Kaisa Nyberg and Howard M. Heys, eds.), Lecture Notes in Computer Science, vol. 2595, Springer, 2002, pp. 250–270.
10. Scott Contini, Arjen K. Lenstra, and Ron Steinfeld, *VSH, an Efficient and Provable Collision-Resistant Hash Function*, Advances in Cryptology - EUROCRYPT 2006 (Serge Vaudenay, ed.), Lecture Notes in Computer Science, vol. 4004, Springer, 2006, pp. 165–182.
11. M. R. Garey and David S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, 1979.
12. Oded Goldreich, Shafi Goldwasser, and Silvio Micali, *How to Construct Random Functions*, J. ACM **33** (1986), no. 4, 792–807.
13. Louis Granboulan, Phong Q. Nguyen, Fabrice Noilhan, and Serge Vaudenay, *DFCv2*, Selected Areas in Cryptography (Douglas R. Stinson and Stafford E. Tavares, eds.), Lecture Notes in Computer Science, vol. 2012, Springer, 2000, pp. 57–71.
14. Louis Granboulan and Thomas Pornin, *Perfect Block Ciphers with Small Blocks*, Fast Software Encryption (Alex Biryukov, ed.), Lecture Notes in Computer Science, vol. 4593, Springer, 2007, pp. 452–465.
15. Charanjit S. Jutla, *Generalized Birthday Attacks on Unbalanced Feistel Networks*, Advances in Cryptology - CRYPTO '98 (Hugo Krawczyk, ed.), Lecture Notes in Computer Science, vol. 1462, Springer, 1998, pp. 186–199.
16. Joe Kilian and Phillip Rogaway, *How to Protect DES Against Exhaustive Key Search*, Advances in Cryptology - CRYPTO '96 (Neal Koblitz, ed.), Lecture Notes in Computer Science, vol. 1109, Springer, 1996, pp. 252–267.
17. Neal Koblitz and Alfred Menezes, *Another Look at "Provable Cryptography"*, J. Cryptology **20** (2007), no. 1, 3–37.
18. Michael Luby and Charles Rackoff, *Pseudo-random Permutation Generators and Cryptographic Composition*, STOC, ACM, 1986, pp. 356–363.
19. ______, *How to Construct Pseudorandom Permutations from Pseudorandom Functions*, SIAM J. Comput. **17** (1988), no. 2, 373–386.
20. Ueli M. Maurer, *A Simplified and Generalized Treatment of Luby-Rackoff Pseudorandom Permutation Generator*, Advances in Cryptology - EUROCRYPT '92 (Rainer A. Rueppel, ed.), Lecture Notes in Computer Science, vol. 658, Springer, 1992, pp. 239–255.

Appeared in SAC 2008.

21. Ueli M. Maurer and Krzysztof Pietrzak, *The Security of Many-Round Luby-Rackoff Pseudo-Random Permutations*, Advances in Cryptology - EUROCRYPT 2003 (Eli Biham, ed.), Lecture Notes in Computer Science, vol. 2656, Springer, 2003, pp. 544–561.
22. Pat Morin, *Provably Secure and Efficient Block Ciphers*, Selected Areas in Cryptography - SAC 1996, 1996, pp. 30–37.
23. Jacques Patarin, *Pseudorandom Permutations Based on the DES Scheme*, EUROCODE '90 (Gérard D. Cohen and Pascale Charpin, eds.), Lecture Notes in Computer Science, vol. 514, Springer, 1990, pp. 193–204.
24. ———, *Etude des générateurs de permutations basés sur le schéma du DES*, Ph.D. thesis, INRIA, Domaine de Voluceau, Le Chesnay, France, 1991.
25. ———, *New Results on Pseudorandom Permutation Generators Based on the DES Scheme*, Advances in Cryptology - CRYPTO '91 (Joan Feigenbaum, ed.), Lecture Notes in Computer Science, vol. 576, Springer, 1991, pp. 301–312.
26. ———, *About Feistel Schemes with Six (or More) Rounds*, Fast Software Encryption (Serge Vaudenay, ed.), Lecture Notes in Computer Science, vol. 1372, Springer, 1998, pp. 103–121.
27. ———, *Generic Attacks on Feistel Schemes*, Advances in Cryptology - ASIACRYPT 2001 (Matthew K. Franklin, ed.), Lecture Notes in Computer Science, vol. 2248, Springer, 2001, pp. 222–238.
28. ———, *Security of Random Feistel Schemes with 5 or More Rounds*, Advances in Cryptology - CRYPTO 2004 (Matthew K. Franklin, ed.), Lecture Notes in Computer Science, vol. 3152, Springer, 2004, pp. 106–122.
29. ———, *On Linear Systems of Equations with Distinct Variables and Small Block Size*, ICISC (Dongho Won and Seungjoo Kim, eds.), Lecture Notes in Computer Science, vol. 3935, Springer, 2005, pp. 299–321.
30. Jacques Patarin, Valérie Nachef, and Côme Berbain, *Generic Attacks on Unbalanced Feistel Schemes with Contracting Functions*, Advances in Cryptology - ASIACRYPT 2006 (Xuejia Lai and Kefei Chen, eds.), Lecture Notes in Computer Science, vol. 4284, Springer, 2006, pp. 396–411.
31. ———, *Generic Attacks on Unbalanced Feistel Schemes with Expanding Functions*, Advances in Cryptology - ASIACRYPT 2007 (Kaoru Kurosawa, ed.), Lecture Notes in Computer Science, vol. 4833, Springer, 2007, pp. 325–341.
32. Jacques Patarin and Joana Treger, *Generic Attacks on Feistel Networks with Internal Permutations*, 2008, in submission.
33. Gilles Piret, *Luby-Rackoff Revisited: On the Use of Permutations as Inner Functions of a Feistel Scheme*, Des. Codes Cryptography **39** (2006), no. 2, 233–245.
34. Serge Vaudenay, *Resistance Against General Iterated Attacks*, Advances in Cryptology - EUROCRYPT '99 (Jacques Stern, ed.), Lecture Notes in Computer Science, vol. 1592, Springer, 1999, pp. 255–271.
35. ———, *Decorrelation: A Theory for Block Cipher Security*, J. Cryptology **16** (2003), no. 4, 249–286.
36. Yuliang Zheng, Tsutomu Matsumoto, and Hideki Imai, *On the Construction of Block Ciphers Provably Secure and Not Relying on Any Unproved Hypotheses*, Advances in Cryptology - CRYPTO '89 (Gilles Brassard, ed.), Lecture Notes in Computer Science, vol. 435, Springer, 1989, pp. 461–480.