Generic Attacks on Feistel Schemes
-Extended Version-

Jacques Patarin

PRiSM, University of Versailles, 45 av. des États-Unis,
78035 Versailles Cedex, France

This paper is the extended version of the paper with the same title published at Asiacrypt'2001 and we have also included here the cryptanalysis results of the paper "Security of Random Feistel Schemes with 5 or more Rounds" published at Crypto'2004.

Abstract

Let A be a Feistel scheme with 5 rounds from $2n$ bits to $2n$ bits. In the present paper we show that for most such schemes A:

1. It is possible to distinguish A from a random permutation from $2n$ bits to $2n$ bits after doing at most $\mathcal{O}(2^n)$ computations with $\mathcal{O}(2^n)$ non-adaptive **chosen** plaintexts.
2. It is possible to distinguish A from a random permutation from $2n$ bits to $2n$ bits after doing at most $\mathcal{O}(2^{\frac{3n}{2}})$ computations with $\mathcal{O}(2^{\frac{3n}{2}})$ **random** plaintext/ciphertext pairs.

Since the complexities are smaller than the number 2^{2n} of possible inputs, they show that some generic attacks always exist on Feistel schemes with 5 rounds. Therefore we recommend in Cryptography to use Feistel schemes with at least 6 rounds in the design of pseudo-random permutations.

We will also show in this paper that it is possible to distinguish most of 6 round Feistel permutations generator from a truly random permutation generator by using a few (i.e. $\mathcal{O}(1)$) permutations of the generator and by using a total number of $\mathcal{O}(2^{2n})$ queries and a total of $\mathcal{O}(2^{2n})$ computations. This result is not really useful to attack a single 6 round Feistel permutation, but it shows that when we have to generate several pseudo-random permutations on a small number of bits we recommend to use more than 6 rounds.

We also show that it is also possible to extend these results to any number of rounds, however with an even larger complexity.

Key words: Feistel permutations, pseudo-random permutations, generic attacks on encryption schemes, Luby-Rackoff theory.

1 My results on (classical, i.e. balanced) Feistel schemes

My results of 2001-2004 on Feistel schemes are presented on 3 papers: this paper for the cryptanalysis results, paper [13] for the security results and paper [14] for a mathematical result that we need in [13] . By Feistel scheme, we mean here classical, i.e. balanced Feistel scheme (i.e. we use round functions from n bits to n bits in order to build a permutation from $2n$ bits to $2n$ bits: see Section 3 for a precise definition). In this paper we will concentrate on cryptanalysis results, i.e. on the best known attacks. This paper is the extended version of the paper with the same title published at Asiacrypt'2001, LNCS 2248, Springer, pp. 222-238, where I have added the generic attacks of the paper "Security of Random Feistel Schemes with 5 or more Rounds" published at Crypto '2004. So this paper merges the results on generic attacks on Feistel Schemes of these two papers.

2 Introduction

Many secret key algorithms used in cryptography are Feistel schemes (a precise definition of a Feistel scheme is given in section 3), for example DES, TDES, many AES candidates, etc.. In order to be as fast as possible, it is interesting to have not too many rounds. However, for security reasons it is important to have a sufficient number of rounds. Generally, when a Feistel scheme is designed for cryptography, the designer either uses many (say ≥ 16 as in DES) very simple rounds, or uses very few (for example 8 as in DFC) more complex rounds. A natural question is: what is the minimum number of rounds required in a Feistel scheme to avoid all the "generic attacks" , i.e. all the attacks effective against most of the schemes, and with a complexity negligible compared with a search on all the possible inputs of the permutation.
Let assume that we have a permutation from $2n$ bits to $2n$ bits. Then a generic attack will be an attack with a complexity negligible compared to $\mathcal{O}(2^{2n})$, since there are 2^{2n} possible inputs on $2n$ bits.
It is easy to see that for a Feistel scheme with only one round there is a generic attack with only 1 query of the permutation and $\mathcal{O}(1)$ computations: just check if the first half (n bits) of the output are equal to the second half of the input.
In [4] it was shown that for a Feistel scheme with two rounds there is also a generic attack with a complexity of $\mathcal{O}(1)$ chosen inputs (or $\mathcal{O}(2^{\frac{n}{2}})$ random inputs).
Also in [4], M. Luby and C. Rackoff have shown their famous result: for more than 3 rounds all generic attacks on Feistel schemes require at least $\mathcal{O}(2^{\frac{n}{2}})$ inputs, even for chosen inputs. If we call a Luby-Rackoff construction (a.k.a. L-R construction) a Feistel scheme instantiated with pseudo-random functions, this result says that the Luby-Rackoff construction with 3 rounds is a pseudorandom permutation.
Moreover for 4 rounds all the generic attacks on Feistel schemes require at least $\mathcal{O}(2^{\frac{n}{2}})$ inputs, even for a stronger attack that combines chosen inputs and chosen outputs (see [4] and a proof in [7], that shows that the Luby-Rackoff construction with 4 rounds is super-pseudorandom, a.k.a strong pseudorandom).
However it was discovered in [8] (and independently in [1]) that these lower bounds on 3 and 4 rounds are tight, i.e. there exist a generic attack on all Feistel schemes with 3 or 4 rounds with $\mathcal{O}(2^{\frac{n}{2}})$ chosen inputs with $\mathcal{O}(2^{\frac{n}{2}})$ computations.
For 5 rounds or more the question is difficult. In [8] it was proved that for 5 rounds (or more) the number of queries must be at least $\mathcal{O}(2^{\frac{2n}{3}})$ (even with unbounded computation complexity), and in [10] it was shown that for 6 rounds (or more) the number of queries must be at least $\mathcal{O}(2^{\frac{3n}{4}})$ (even with unbounded computations). Finally in [13], [14], it was proved that for 5 rounds (or more) the number of queries must be at least $O(2^n)$.
It can be noticed (see [8]) that if we have access to unbounded computations, then we can make an exhaustive search on all the possible round functions of the Feistel scheme, and this will give an attack with only $\mathcal{O}(2^n)$ queries (see [8]) so the bound $O(2^n)$ of the number of queries is optimal. However here we have a gigantic complexity $\geq \mathcal{O}(2^{n2^n})$. This "exhaustive search" attack always exists, but since the complexity is far much larger than the exhaustive search on plaintexts in $\mathcal{O}(2^{2n})$, it was still an open problem to know if generic attacks, with a complexity $\ll \mathcal{O}(2^{2n})$, exist on 5 rounds (or more) of Feistel schemes. This is the subject of this paper.
In this paper we will indeed show that there exist generic attacks on 5 rounds of the Feistel scheme, with a complexity $\ll \mathcal{O}(2^{2n})$. We describe two attacks on 5 round Feistel schemes:

1. An attack with $\mathcal{O}(2^{\frac{3n}{2}})$ computations on $\mathcal{O}(2^{\frac{3n}{2}})$ **random** input/output pairs.
2. An attack with $\mathcal{O}(2^n)$ computations on $\mathcal{O}(2^n)$ **chosen** inputs.

For 6 rounds (or more) we will describe some attacks with a complexity much smaller than $\mathcal{O}(2^{n2^n})$ of exhaustive search, but still $\geq \mathcal{O}(2^{2n})$. So these attacks on 6 rounds and more are generally not interesting against a single permutation. However they may be useful when several permutations are used, i.e. they will be able to distinguish some permutation generators. These attacks show for example that when several small permutations must be generated (for example in the Graph Isomorphism scheme, or as in the Permuted Kernel scheme) then we must not use a 6 round Feistel construction.

Remark The generic attacks presented here for 3, 4 and 5 rounds are effective against most Feistel schemes, or when the round functions are randomly chosen. However it can occur that for specific choices of the round function, the attacks, performed exactly as described, may fail. However in this case, very often there are modified attacks on these specific round functions.

3 Notations

We use the following notations that are very similar to those used in [4], [6] and [10].

- $I_n = \{0,1\}^n$ is the set of the 2^n binary strings of length n.
- For $a, b \in I_n$, $[a, b]$ will be the string of length $2n$ of I_{2n} which is the concatenation of a and b.
- For $a, b \in I_n$, $a \oplus b$ stands for bit by bit exclusive or of a and b.
- $\circ$ is the composition of functions.
- The set of all functions from I_n to I_n is F_n. Thus $|F_n| = 2^{n \cdot 2^n}$.
- The set of all permutations from I_n to I_n is B_n. Thus $B_n \subset F_n$, and $|B_n| = (2^n)!$
- Let f_1 be a function of F_n. Let L, R, S and T be elements of I_n. Then by definition

$$\Psi(f_1)[L,R] = [S,T] \overset{\text{def}}{\Leftrightarrow} \begin{cases} S = R \\ \text{and} \\ T = L \oplus f_1(R) \end{cases}$$

- Let $f_1, f_2, \ldots, f_k$ be k functions of F_n. Then by definition:

$$\Psi^k(f_1, \ldots, f_k) = \Psi(f_k) \circ \cdots \circ \Psi(f_2) \circ \Psi(f_1).$$

The permutation $\Psi^k(f_1, \ldots, f_k)$ is called "a Feistel scheme with k rounds" and also called Ψ^k.

4 Generic attacks on 1,2,3 and 4 rounds

Up till now, generic attacks had been discovered for Feistel schemes with 1,2,3,4 rounds. Let us shortly describe these attacks.
Let f be a permutation of B_{2n}. For a value $[L_i, R_i] \in I_{2n}$ we will denote by $[S_i, T_i] = f[L_i, R_i]$.

1 round
The attack just tests if $S_1 = R_1$. If f is a Feistel scheme with 1 round, this will happen with 100% probability, and if f is a random permutation with probability $\simeq \frac{1}{2^n}$. So with one round there is a generic attack with only 1 random query and $\mathcal{O}(1)$ computations.

2 rounds, CPA-1 with $m = 2$ messages (non-adaptive chosen plaintext attack)
Let choose $R_2 = R_1$ and $L_2 \neq L_1$. Then the attack just tests if $S_1 \oplus S_2 = L_1 \oplus L_2$. This will occur with 100% probability if f is a Feistel scheme with 2 rounds, and if f is a random permutation with probability $\simeq \frac{1}{2^n}$. So with two rounds there is a generic attack with only 2 non-adaptive chosen queries and $\mathcal{O}(1)$ computations.

2 rounds, known plaintext attack with $m \simeq 2^{n/2}$
It is possible to transform this non-adaptive chosen plaintext attack in a known plaintext attack like the following. If we have $\mathcal{O}(2^{\frac{n}{2}})$ random inputs $[L_i, R_i]$, then with a good probability we will have a collision $R_i = R_j, i \neq j$. Then we test if $S_i \oplus S_j = L_i \oplus L_j$. Now the attack requires $\mathcal{O}(2^{\frac{n}{2}})$ random queries and $\mathcal{O}(2^{\frac{n}{2}})$ computations.

Note This attack on 1 and 2 rounds was already described in [4].

3 rounds, known plaintext attack with $m \simeq 2^{n/2}$

Let ϕ be the following algorithm :

1. ϕ chooses m random distinct $[L_i, R_i]$, $1 \leq i \leq m$.
2. ϕ asks for the values $[S_i, T_i] = f[L_i, R_i], 1 \leq i \leq m$.
3. ϕ counts the number N of equalities of the form $R_i \oplus S_i = R_j \oplus S_j, i < j$.
4. Let N_0 be the expected value of N when f is a random permutation, and N_1 be the expected value of N when f is a $\Psi^3(f_1, f_2, f_3)$, with randomly chosen f_1, f_2, f_3.
   Then $N_1 \simeq 2N_0$, because when f is a $\Psi^3(f_1, f_2, f_3)$, $R_i \oplus S_i = f_2(L_i \oplus f_1(R_i))$ so $f_2(L_i \oplus f_1(R_i)) = f_2(L_j \oplus f_1(R_j)), i < j$, if $L_i \oplus f_1(R_i) \neq L_j \oplus f_1(R_j)$ and $f_2(L_i \oplus f_1(R_i)) = f_2(L_j \oplus f_1(R_j))$ or if $L_i \oplus f_1(R_i) = L_j \oplus f_1(R_j)$.

So by counting N we will obtain a way to distinguish 3 round Feistel permutations from random permutations. This generic attack requires $\mathcal{O}(2^{\frac{n}{2}})$ random queries and $\mathcal{O}(2^{\frac{n}{2}})$ computations (just store the values $R_i \oplus S_i$ and count the collisions).

Remark Here $N_1 \simeq 2 \cdot N_0$ when f_1, f_2, f_3 are randomly chosen. Therefore this attack is effective on most of 3 round Feistel schemes but not necessarily on all 3 round Feistel schemes (however very special f_1, f_2, f_3 may create other attacks, as we will see for example with the Knudsen attack in Section 5).

3 rounds, CPCA-2 with $m = 3$ (adaptive chosen plaintext and chosen ciphertext attack)

For 3 rounds there is also an attack that uses both an encryption and decryption oracles with only 3 queries. Let ϕ be the following algorithm :

1. ϕ chooses two elements L_1 and R_1 of I_n and asks the encryption oracle for the value of $f[L_1, R_1] = [S_1, T_1]$.
2. ϕ chooses an element $L_2 \neq L_1$ and asks for the value of $f[L_2, R_1] = [S_2, T_2]$.
3. ϕ asks the decryption oracle for the value of $f^{-1}[S_2, T_2 \oplus L_1 \oplus L_2] = [L_3, R_3]$. Then ϕ tests if $R_3 = S_2 \oplus S_1 \oplus R_1$. This will always be true if f is a Ψ^3, and will appear with probability $\approx 1/2^n$ if f is a random permutation.

Remark How this attack can be found.

It is easy to check that the attack above works. It is also possible to explain how such an attack can be found, as we will do now.

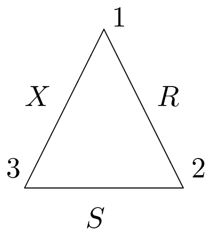


Figure 1: A circle in R, S, X.

The idea is to create a circle in R, S, X, as in figure 1, where $X_i = L_i \oplus f_1(R_i)$, i.e. to have $R_2 = R_1$, $S_3 = S_2$ and $X_3 = X_1$. We always have:

$$R_i = R_j \Rightarrow L_i \oplus L_j = X_i \oplus X_j \quad (1)$$

$$X_i = X_j \Rightarrow R_i \oplus R_j = S_i \oplus S_j \quad (2)$$

$$S_i = S_j \Rightarrow X_i \oplus X_j = T_i \oplus T_j \quad (3)$$

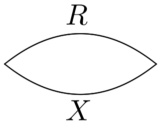
First, we choose $R_2 = R_1$ and $L_2 \neq L_1$. So from (1), we have:
$X_2 \oplus X_1 = L_1 \oplus L_2 \quad (4)$.
Second, we choose $S_3 = S_2$. So from (3), we have: $X_2 \oplus X_3 = T_2 \oplus T_3 \quad (5)$.
So from (4) and (5) we can impose $X_3 = X_1$ by choosing $T_3 = T_2 \oplus L_1 \oplus L_2$. Then from (2) we will have: $R_3 = R_1 \oplus S_1 \oplus S_3$ $(= R_1 \oplus S_1 \oplus S_2)$.

4 rounds, CPA-1 with $m \simeq 2^{n/2}$ (non-adaptive chosen plaintext attack)

This time, we take $R_i = 0$ (or R_i constant), and we count the number N of equalities of the form $S_i \oplus L_i = S_j \oplus L_j$, $i < j$. In fact, when $f = \Psi^4(f_1, f_2, f_3, f_4)$, then $S_i \oplus L_i = f_3(f_2(L_i \oplus f_1(0))) \oplus f_1(0)$. So the probability of such an equality is about the double in this case (as long as f_1, f_2, f_3 are randomly chosen) than in the case where f is a random permutation (because if $f_2(L_i \oplus f_1(0)) = f_2(L_j \oplus f_1(0))$ this equality holds, and if $\beta_i = f_2(L_i \oplus f_1(0)) \neq f_2(L_j \oplus f_1(0)) = \beta_j$ but $f_3(\beta_i) = f_3(\beta_j)$, this equality also holds).
So by counting N we will obtain a way to distinguish 4 round Feistel permutations from random permutations. This generic attack requires $\mathcal{O}(2^{\frac{n}{2}})$ non-adaptive chosen queries and $\mathcal{O}(2^{\frac{n}{2}})$ computations (just store the values $S_i \oplus L_i$ and count the collisions).

Notes

1. These attacks for 3 and 4 rounds have been first published in [8], and independently re-discovered in [1].
2. Here again the attack is effective against most of 4 round Feistel schemes but not necessarily on all 4 round Feistel schemes (however very special f_1, f_2, f_3, f_4 may create other attacks, as we will see for example with the Knudsen attack in Section 5).
3. Here, for 4 rounds the attack can be seen geometrically as a way to create a circle in R, X.



4 rounds, known plaintext attack with $m \simeq 2^n$

When $m \geq \mathcal{O}(2^n)$, it is possible to transform this attack in a known plaintext attack. We will count the number N of (i,j), $1 \leq i < j \leq m$ such that $R_i = R_j$ and $S_i \oplus L_i = S_j \oplus L_j$. For a random permutation $N \simeq \frac{m^2}{2 \cdot 2^{2n}}$, and for a Ψ^4 we have $N \simeq \frac{m^2}{2^{2n}}$ (i.e. about double).

Remark Here the number of computations to be done is $\mathcal{O}(m)$ if we have $\mathcal{O}(m)$ in memory (for all i compute $S_i \oplus L_i$ and store $+1$ at the address $R_i || S_i \oplus L_i$).

5 Generic attacks on Ψ^5

We will present here the two best generic attacks that we have found on Ψ^5:

1. A CPA-1 attack on Ψ^5 with $m \simeq 2^n$ and $\lambda = O(2^n)$ computations.
2. A KPA on Ψ^5 with $m \simeq 2^{3n/2}$ and $\lambda = O(2^{3n/2})$computations.

1. CPA-1 attack on Ψ^5.

   Let us assume that R_i =constant, $\forall i$, $1 \leq i \leq m$, $m \simeq 2^n$. We will simply count the number N of (i,j), $i < j$ such that $S_i = S_j$ and $L_i \oplus T_i = L_j \oplus T_j$. This number N will be about double for Ψ^5 compared with a truly random permutation.

   Proof:

   $$\begin{aligned} \text{If } S_i = S_j, L_i \oplus T_i = L_j \oplus T_j \quad &\Leftrightarrow \quad L_i \oplus Z_i = L_j \oplus Z_j \\ &\Leftrightarrow \quad f_1(R_1) \oplus f_3(Y_i) = f_1(R_1) \oplus f_3(Y_j) \\ &\Leftrightarrow \quad f_3(R_1 \oplus f_2(L_i \oplus f_1(R_1))) \\ &\qquad = f_3(R_1 \oplus f_2(L_j \oplus f_1(R_1))) \quad (\#) \end{aligned}$$

   This will occur if $f_2(L_i \oplus f_1(R_1)) = f_2(L_j \oplus f_1(R_1))$, or if these values are distinct but when Xored with R, they have the same images by f_3, so the probability is about two times larger.

   Remarks

   (a) By storing the $S_i||L_i \oplus T_i$ values and looking for collisions, the complexity is in $\lambda \simeq O(2^n)$.

   (b) With a single value for R_i, we will get very few collisions. However this attack becomes significant if we have a few values R_i and for all these values about 2^n values L_i.

2. KPA on Ψ^5.

   The CPA-1 attack can immediately be transformed in a KPA: for random $[L_i, R_i]$, we will simply count the number N of (i,j), $i < j$ such that $R_i = R_j$, $S_i = S_j$, and $L_i \oplus T_i = L_j \oplus T_j$. We will get about $\frac{m(m-1)}{2^{3n}}$ such collisions for Ψ^5, and about $\frac{m(m-1)}{2 \cdot 2^{3n}}$ for a random permutation. This KPA is efficient when m^2 becomes not negligible compared with 2^{3n}, i.e. when $m \geq$ about $2^{3n/2}$.

Remark 1 If we count the number N of (i,j), $i < j$ such that $R_i \oplus R_j = S_i \oplus S_j$, we get another KPA attack with a similar complexity.

Remark 2 These attacks are very similar with the attacks on 5-round Feistel schemes described by Knudsen (cf [2]) in the case where (unlike us) f_2 and f_3 are permutations (therefore, not random functions). Knudsen attacks are based on this theorem:

Theorem 5.1 (Knudsen, see [2]) *Let $[L_1, R_1]$ and $[L_2, R_2]$ be two inputs of a 5-round Feistel scheme, and let $[S_1, T_1]$ and $[S_2, T_2]$ be the outputs. Let us assume that the round functions f_2 and f_3 are permutations (therefore they are not random functions of F_n). Then, if $R_1 = R_2$ and $L_1 \neq L_2$, it is impossible to have simultaneously $S_1 = S_2$ and $L_1 \oplus L_2 = T_1 \oplus T_2$.*

Proof This comes immediately from (#) above.

6 Attacking Feistel Generators

In this section we will describe what is an attack against a generator of permutations (and not only against a single permutation randomly generated by a generator of permutations), i.e. we will be able to study several permutations generated by the generator. Then we will evaluate the complexity of brute force attacks and we will notice that since all Feistel permutations have an even signature, it is possible to distinguish them from a random permutation in $\mathcal{O}(2^{2n})$.
Let G be a "k round Feistel Generator", i.e. from a binary string K, G generates a k round Feistel permutation G_K of B_{2n}.
Let G' be a truly random permutation generator, i.e. from a string K, G' generates a truly random permutation G'_K of B_{2n}.
Let G'' be a truly random even permutation generator, i.e. from a string K, G'' generates a truly random permutation G''_K of A_{2n}, with A_{2n} being the group of all the permutations of B_{2n} with even signature.
We are looking for attacks that distinguish G from G', and also for attacks that will distinguish G from G''.

Adversarial model: An attacker can choose some strings $K_1, \dots K_f$, can ask for some inputs $[L_i, R_i] \in I_{2n}$, and can ask for some $G_{K_\alpha}[L_i, R_i]$ (with K_α being one of the K_i). Here the attack is more general than in the previous sections, since the attacker can have access to many different permutations generated by the same generator.

Adversarial goal: The aim of the attacker is to distinguish G from G' (or from G'') with a good probability and with a complexity as small as possible.

Brute force attacks A possible attack is the exhaustive search on the k round functions $f_1, \dots, f_k$ form I_n to I_n that have been used in the Feistel construction. This attack always exists, but since we have $2^{k \cdot n \cdot 2^n}$ possibilities for $f_1, \dots, f_k$, this attack requires about $2^{k \cdot n \cdot 2^n}$ computations (or $2^{\lceil \frac{k}{2} \rceil \cdot n \cdot 2^n}$ computations in a version "in the middle" of the attack) and about $k \cdot 2^{n-1}$ random queries[1] and only 1 permutation of the generator.

Attack by the signature

Theorem 6.1 *If $n \geq 2$ then all the Feistel schemes from $I_{2n} \to I_{2n}$ have an even signature.*

Proof
Let $\sigma : I_{2n} \to I_{2n}$
$[L, R] \mapsto [R, L]$.
Let f_1 be a function of F_n.
Let $\Psi'(f_1)[L, R] = [L \oplus f_1(R), R]$.
We will show that both σ and $\Psi'(f_1)$ have an even signature, so will have $\sigma \circ \Psi'(f_1) = \Psi(f_1)$, and thus by composition, all the Feistel schemes from $I_{2n} \to I_{2n}$ have an even signature.

For σ: All the cycles have 1 or 2 elements since $\sigma \circ \sigma = Id$. We have 2^n cycles with 1 element since $\sigma[L, R] = [L, R]$ if and only if $L = R$ (and a cycle with 1 element has an even signature). So we have $\frac{2^{2n} - 2^n}{2}$ cycles with 2 elements. When $n \geq 2$ this number is even.

For $\Psi'(f_1)$: All the cycles have 1 or 2 elements since $\Psi'(f_1) \circ \Psi'(f_1) = Id$. Moreover $\Psi'(f_1)[L, R] = [L, R]$ if and only if $f_1(R) = 0$, so the number of cycles with 2 elements is $\frac{2^n \cdot k}{2}$, with k being the number of values R such that $f_1(R) \neq 0$. So when $n \geq 2$ the signature of $\Psi'(f_1)$ is even.

[1] each query divides by about 2^{2n} the number of possible $f_1, \dots, f_k$

Theorem 6.2 *Let f be a permutation of B_{2n}. Then using $\mathcal{O}(2^{2n})$ computations on the 2^{2n} input/output values of f, we can compute the signature of f.*

Proof

Just compute all the cycles c_i of f, $f = \prod_{i=1}^{\alpha} c_i$ and use the formula:
signature$(f) = \prod_{i=1}^{\alpha} (-1)^{length(c_i)+1}$.

Theorem 6.3 *Let G be a Feistel scheme generator, then it is possible to distinguish G from a generator of truly random permutations of B_{2n} after $\mathcal{O}(2^{2n})$ computations on $\mathcal{O}(2^{2n})$ input/output values.*

Proof

It is a direct consequence of the Theorems 6.1 and 6.2 above.

Remark

It is however probably much more difficult to distinguish G from random permutations of A_{2n}, with A_{2n} being the group of all the permutations of B_{2n} with even signature. In the next sections we will present our best attacks for this problem.

7 An attack on 6 round Feistel Generators with $\mathcal{O}(2^{2n})$ random plaintexts and $\mathcal{O}(2^{2n})$ complexity

Attacks on 6 round Feistel If G is a generator of 6 round Feistel permutations of B_{2n}, we have found an attack (described below) that uses a few (i.e. $\mathcal{O}(1)$) permutations from the generator G, $\mathcal{O}(2^{2n})$ computations and about $\mathcal{O}(2^{2n})$ random queries. So this attack has a complexity much smaller than the exhaustive search in $2^{3n \cdot 2^n}$. However since a permutation of B_{2n} has only 2^{2n} possible inputs, this attack has no real interest against a single specific 6 round Feistel scheme used in encryption.
It is interesting only if at least a few 6 round Feistel schemes are used. This can be particularly interesting for some cryptographic schemes using many permutations on a relatively small number of bits. For example in the Graph Isomorphism authentication scheme many permutations on about 2^{14} points are used (thus $n = 7$), or in the Permuted Kernel Problem PKP of Adi Shamir many permutations on about 2^6 points ($n = 3$ here). Then, we will be able to distinguish these permutations from truly random permutations with a small complexity if a 6 round Feistel scheme generator is used. And this, whatever the size of the secret key used in the generator may be. So we do not recommend to generate small pseudorandom permutations from 6 round Feistel schemes.

The Attack:

Let $[L_i, R_i]$ be an element of I_{2n}.
Let $\Psi^6[L_i, R_i] = [S_i, T_i]$. The attack proceeds as follows:

Step 1

We choose a specific permutation $f = G_K$.
We generate m values $f[L_i, R_i] = [S_i, T_i]$, $1 \leq i \leq m$ with the random $[L_i, R_i] \in I_{2n}$ and with $m = \mathcal{O}(2^{2n})$.
Remark: Since $m = \mathcal{O}(2^{2n})$, we cover here almost all the possible inputs $[L_i, R_i]$ for this specific permutation f.

Step 2

We look if among these values we can find 4 pairwise distinct indices denoted by $1, 2, 3, 4$ such that these 8 equations are satisfied:

$$(\#)\begin{cases} R_1 = R_3 \\ R_2 = R_4 \\ S_1 = S_2 \\ S_3 = S_4 \\ L_1 \oplus L_3 = L_2 \oplus L_4 \\ L_1 \oplus L_3 = S_1 \oplus S_3 \\ T_1 \oplus T_2 = T_3 \oplus T_4 \\ T_1 \oplus T_2 = R_1 \oplus R_2 \end{cases}$$

(and with $R_2 \neq R_1$, $S_3 \neq S_1$ and $T_1 \neq T_2$).

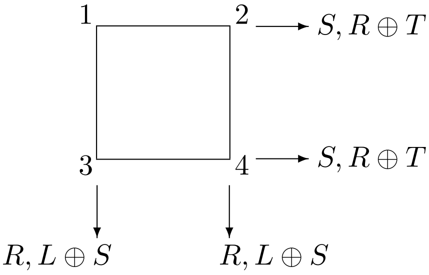


Figure 3: A representation of the 8 *equations* # *in* L, S, R, T.

It is also possible to show that all the indices that satisfy these equations can be found in $\mathcal{O}(m)$ and with $\mathcal{O}(m)$ of memory. We count the number of solutions found.

Step 3

We try again at Step 1 with another $f = G_{K'}$ and we will do this a few times, say λ times with $\lambda = \mathcal{O}(1)$. Let α be the total number of solutions found at Step 2 for all the λ functions tested. It is possible to prove that for a generator of pseudorandom permutation of B_{2n} we have

$$\alpha \simeq \frac{\lambda m^4}{2^{8n}}.$$

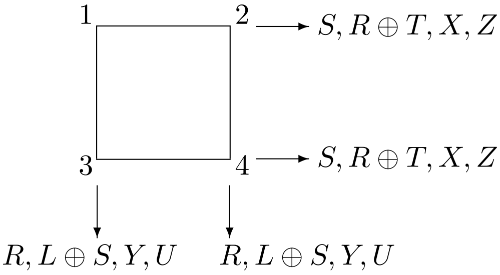
Moreover it is possible to prove that for a generator of 6 round Feistel schemes the average value we get for α is

$$\alpha \geq \text{ about } \frac{2\lambda m^4}{2^{8n}}.$$

So by counting this value α we will distinguish 6 round Feistel generators for example when $\lambda = \mathcal{O}(1)$ and $m = \mathcal{O}(2^{2n})$, as claimed.

Proof

The proof is very similar to the proof we did for Ψ^5. For Ψ^6 we can get the 8 equations # with about the same probability when all the internal variables X, Y, Z, U are pairwise distinct, or when we have the relations of figure 4 (so the probability is about double compared with random permutations).



This comes from the fact that all these equations come from these 8 equations:

$$(\mathbf{\Lambda})\begin{cases} R_1 = R_3 \;\; (1) \\ R_2 = R_4 \;\; (2) \\ X_1 = X_2 \;\; (3) \\ L_1 \oplus L_2 = L_3 \oplus L_4 \;\; (4) \\ Y_1 = Y_3 \;\; (5) \\ Z_1 = Z_2 \;\; (6) \\ U_1 = U_3 \;\; (7) \\ S_1 = S_2 \;\; (8) \end{cases}$$

and from the usual relations:

$$\begin{array}{rcrcll} R_i = R_j & \Rightarrow & X_i \oplus X_j & = & L_i \oplus L_j & \mathbf{(CR)} \\ X_i = X_j & \Rightarrow & Y_i \oplus Y_j & = & R_i \oplus R_j & \mathbf{(CX)} \\ Y_i = Y_j & \Rightarrow & Z_i \oplus Z_j & = & X_i \oplus X_j & \mathbf{(CY)} \\ Z_i = Z_j & \Rightarrow & U_i \oplus U_j & = & Y_i \oplus Y_j & \mathbf{(CZ)} \\ U_i = U_j & \Rightarrow & Z_i \oplus Z_j & = & S_i \oplus S_j & \mathbf{(CU)} \\ S_i = S_j & \Rightarrow & U_i \oplus U_j & = & T_i \oplus T_j & \mathbf{(CS)} \end{array}$$

Proof that # comes from Λ with these usual relations

From (1), (2), (CR) we get: $X_1 \oplus X_3 = L_1 \oplus L_3$ and $X_2 \oplus X_4 = L_2 \oplus L_4$.
So from (3), (4) we get: $X_1 = X_2$ and $X_3 = X_4$.
So from (CX) we get: $Y_1 \oplus Y_2 = R_1 \oplus R_2$ and $Y_3 \oplus Y_4 = R_3 \oplus R_4$.
So from (1), (2), (5) we get: $Y_1 = Y_3$ and $Y_2 = Y_4$.
So from (CY) we get: $Z_1 \oplus Z_3 = X_1 \oplus X_3$ and $Z_2 \oplus Z_4 = X_2 \oplus X_4$.
So from (6) and $X_1 = X_2$ and $X_3 = X_4$ we get: $Z_1 = Z_2$ and $Z_3 = Z_4$.
So from (CZ) we get: $U_1 \oplus U_2 = Y_1 \oplus Y_2$ and $U_3 \oplus U_4 = Y_3 \oplus Y_4$.
So from (7) and $Y_1 = Y_3$ and $Y_2 = Y_4$ we get: $U_1 = U_3$ and $U_2 = U_4$.
So from (CU) we get: $S_1 \oplus S_3 = Z_1 \oplus Z_3$ ($= X_1 \oplus X_3 = L_1 \oplus L_3$ from above) and $S_2 \oplus S_4 = Z_2 \oplus Z_4$.
So from (8) and $Z_1 = Z_2$ and $Z_3 = Z_4$ we get: $S_1 = S_2$ and $S_3 = S_4$.
So from (CS) we get: $T_1 \oplus T_2 = U_1 \oplus U_2$ and $T_3 \oplus T_4 = U_3 \oplus U_4$.
So $T_1 \oplus T_2 (= U_1 \oplus U_2 = Y_1 \oplus Y_2) = R_1 \oplus R_2$ and $T_3 \oplus T_4 = R_3 \oplus R_4$.
So we have obtained all the 8 equations of # from the 8 equations of Λ as claimed.

Examples: Thus we are able, to distinguish between a few 6 round Feistel permutations taken from a generator, and a set of truly random permutations (or from a set of random permutations with an even signature) from 32 bits to 32, within approximately 2^{32} computations and 2^{32} chosen plaintexts.

8 First attacks on k round Feistel Generators

It is also possible to extend these attacks on more than 6 rounds, to any number of rounds k. However for more than 6 rounds, as already for 6 rounds, all our attacks require a complexity and a number of queries $\geq \mathcal{O}(2^{2n})$, so they can be interesting to attack generators of permutations, but not to attack a single permutation (the probability of success against one single permutation is generally negligible, and we need a few, or many permutations from the generator, in order to be able to distinguish the generator from a truly random permutation generator).

Example of attack on a Feistel generator with k rounds. Let k be an integer. For simplicity we will assume that k is even (the proof is very similar when k is odd). Let $\lambda = \frac{k}{2} - 1$. Let G be a generator of Feistel permutations of k rounds of B_{2n}. We will consider an attack with a set of equations in (L, R, S, T) illustrated in figure 4. For simplicity we do not write all the equations explicitly.

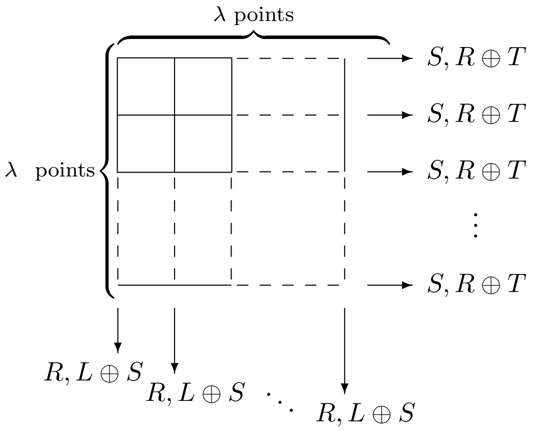


Figure 4: Modelling the $4 \cdot \lambda(\lambda-1)$ equations in L, R, S, T.

Here we have $\mu = \lambda^2 = (\frac{k}{2}-1)^2$ indices, and we have $4\lambda(\lambda-1) = k^2 - 6k + 8$ equations in L, R, S, T. Here it is possible to prove that the probability that the $4\lambda(\lambda-1)$ equations of figure 4 exist, will be about twice for a Feistel scheme with k rounds, than for a truly random permutation.
Thus, on a fixed permutation this attack succeeds with a probability in

$$\mathcal{O}\left(\frac{m^{(\frac{k}{2}-1)^2}}{2^{n \cdot 4\lambda(\lambda-1)}}\right)$$

If we take $m = \mathcal{O}(2^{2n})$ for such a permutation, it gives a probability of success in

$$\mathcal{O}\left(\frac{2^{2n(\frac{k}{2}-1)^2}}{2^{n\cdot(k^2-6k+8)}}\right)$$

So we will use $\mathcal{O}(2^{n(\frac{k^2}{2}-4k+6)})$ permutations, and the total complexity and the total number of queries on all these permutations will be $\mathcal{O}(2^{n(\frac{k^2}{2}-4k+8)})$. The total memory will be $\mathcal{O}(2^{2n})$.

Examples:

- With $k = 6$ this attack uses $\mathcal{O}(1)$ permutations and $\mathcal{O}(2^{2n})$ computations (exactly as we did in section 7).
- With $k = 8$ we need $\mathcal{O}(2^{6n})$ permutations and $\mathcal{O}(2^{8n})$ computations.

9 Improved attacks on Ψ^k generators, $k \geq 6$

Ψ^k has always an even signature. This gives an attack in 2^{2n} if we want to distinguish Ψ^k from random permutations (see section 6) and if we have all the possible cleartext/ciphertext. In this appendix, we will present the best attacks that we know when we want to distinguish Ψ^k from random permutations with an even signature, or when we do not have exactly all the possible cleartext/ciphertext.

1. KPA with k even.

   Let (i, j) be two indices, $i \neq j$, such that $R_i = R_j$ and $S_i \oplus S_j = L_i \oplus L_j$. From [8] or [9] p.146, we know the exact value of H in this case, when k is even. We have:

$$H = H^*\left(1 + \frac{1}{2^{(\frac{k}{2}-2)n}} - \frac{1}{2^{(\frac{k}{2}-1)n}} - \frac{2}{2^{\frac{kn}{2}}} + \frac{1}{2^{(k-1)n}}\right)$$

   where

$$H^* = \frac{|F_n|^k}{2^{2nm}} \cdot \frac{1}{1 - \frac{1}{2^{2n}}}$$

i.e. H^* is the average value of H on two cleartext/ciphertext. So there is a small deviation, of about $\frac{1}{2^{(\frac{k}{2}-2)n}}$, from the average value.

So in a KPA, when the $[L_i, R_i]$ are chosen at random, and if the f_i functions are chosen at random, we will get slightly more (i,j), $i<j$, with $R_i = R_j$ and $S_i \oplus S_j = L_i \oplus L_j$ from a Ψ^k (with k even) than from a truly random permutation. This can be detected if we have enough cleartext/ciphertext pairs from many Ψ^k permutations. In first approximation, these relations will act like independent Bernoulli variables (in reality the equations are not truly independent, but this is expected to create only a modification of second order).

If we have N possibilities for (i,j), $i<j$, and if X is the number of (i,j), $i<j/R_i = R_j$ and $S_i \oplus S_j = L_i \oplus L_j$, we expect to have:

$E(X) \simeq \frac{N}{2^{2n}}$

$V(X) \simeq \frac{N}{2^{2n}}$

$\sigma(X) \simeq \frac{\sqrt{N}}{2^n}$.

We want $\sigma(X) \leq \frac{N}{2^{(\frac{k}{2}-2)n}} \cdot \frac{1}{2^{2n}}$ in order to distinguish Ψ^k from a random permutation. So we want $\frac{\sqrt{N}}{2^n} \leq \frac{N}{2^{\frac{k}{2}n}}$ i.e. $N \geq 2^{(k-2)n}$.

However, if we have μ available permutations, with about 2^{2n} cleartext/ciphertext for each of these permutations, then $N \simeq 2^{4n}\mu$ (here we know these μ permutations almost on every possible cleartext. If not, μ will be larger and we will do more computations). $N \geq 2^{(k-2)n}$ gives $\mu \geq 2^{(k-6)n}$. This is an attack with $2^{(k-6)n}$ permutations and $2^{2n}\mu \simeq 2^{(k-4)n}$ computations.

2. KPA with k odd.

Let (i,j) be two indices, $i \neq j$, such that $R_i = R_j$, $S_i = S_j$ and $L_i \oplus L_j = T_i \oplus T_j$. From [9] p.147, we know the exact value of H in this case, when k is odd. We have:

$$H = H^* \left(1 + \frac{1}{2^{(\frac{k}{2}-\frac{5}{2})n}} - \frac{1}{2^{(\frac{k}{2}-\frac{3}{2})n}} - \frac{2}{2^{(\frac{k}{2}-\frac{1}{2})n}} + \frac{1}{2^{(k-2)n}}\right)$$

where H^* is the average value of H on two cleartext/ciphertext. So there is a small deviation, of about $\frac{1}{2^{(\frac{k}{2}-\frac{5}{2})n}}$, from the average value.

So in a KPA, when the $[L_i, R_i]$ are chosen at random, and if the f_i functions are chosen at random, we will get slightly more (i,j), $i<j$, with $R_i = R_j$, $S_i = S_j$ and $L_i \oplus L_j = T_i \oplus T_j$ from a Ψ^k (with k odd) than from a truly random permutation. In first approximation, these relations will act like independent Bernoulli variables (in reality the equations are not truly independent, but this is expected to create only a modification of second order).

If we have N possibilities for (i,j), $i<j$, and if X is the number of (i,j), $i<j/R_i = R_j$, $S_i = S_j$ and $L_i \oplus L_j = T_i \oplus T_j$, we expect to have:

$E(X) \simeq \frac{N}{2^{3n}}$

$V(X) \simeq \frac{N}{2^{3n}}$

$\sigma(X) \simeq \frac{\sqrt{N}}{2^{\frac{3n}{2}}}$.

We want $\sigma(X) \leq \frac{N}{2^{(\frac{k}{2}-\frac{5}{2})n}} \cdot \frac{1}{2^{3n}}$ in order to distinguish Ψ^k from a random permutation. So we want $\frac{\sqrt{N}}{2^{\frac{3n}{2}}} \leq \frac{N}{2^{(\frac{k}{2}-\frac{1}{2})n}}$ i.e. $N \geq 2^{(k-2)n}$.

However, if we have μ available permutations, with about 2^{2n} cleartext/ciphertext for each of these permutations, then $N \simeq 2^{4n}\mu$ (here we know these μ permutations almost on every possible cleartext.

If not, μ will be larger and we will do more computations). So $N \geq 2^{(k-2)n}$ gives $\mu \geq 2^{(k-6)n}$. This is an attack with $2^{(k-6)n}$ permutations and $2^{2n}\mu \simeq 2^{(k-4)n}$ computations.

Remark If we count the number N of (i,j), $i < j$ such that $R_i \oplus R_j = S_i \oplus S_j$, then we get another KPA with the same complexity.

3. CPA and CPCA attacks.

   For CPA or CPCA attacks we have not found anything really better than these KPA attacks when we have $k \geq 6$ rounds.

10 Conclusion

Up till now, generic attacks on Feistel schemes were known only for 1,2,3 or 4 rounds. In this paper we have seen that some generic attacks also do exist on 5 round Feistel schemes. So we do not recommend to use 5 round Feistel schemes in cryptography for general purposes. Our first attack requires $\mathcal{O}(2^{\frac{3n}{2}})$ **random** plaintext/ciphertext pairs and the same amount of computation time. Our second attack requires $\mathcal{O}(2^n)$ **chosen** plaintext/ciphertext pairs and the same amount of computation time. For example, it is possible to distinguish most of 5 round Feistel ciphers with blocks of 64 bits, from a random permutation from 64 bits to 64 bits, within about 2^{32} chosen queries and 2^{32} computations.
We have also seen that when we have to generate several small pseudo-random permutations we do not recommend to use a Feistel scheme generator with only 6 rounds (whatever the length of the secret key may be). As an example, it is possible to distinguish most generators of 6 round Feistel permutations from truly random permutations on 32 bits, within approximately 2^{32} computations and 2^{32} chosen plaintexts (and this whatever the length of the secret key may be).
Similar attacks can be generalised for any number of rounds k, but they require to analyse much more permutations and they have a larger complexity when k increases.

11 Acknowledgments

I would like to thank Jean-Jacques Quisquater who allowed me to do this work, as it has been done during my invited stay at the university of Louvain-La-Neuve. I also would like to thank the anonymous referee of Asiacrypt'2001, for pointing out the references [2, 3], and for observing that my attack against 5 round Feistel schemes will not in general apply as it is, against some specific round functions such as permutations.

References

[1] William Aiollo, Ramarathnam Venkatesan: *Foiling Birthday Attacks in Length-Doubling Transformations - Benes: A Non-Reversible Alternative to Feistel.* Eurocrypt 96, LNCS 1070, Springer, pp. 307-320.

[2] L.R. Knudsen: *DEAL - A 128-bit Block Cipher*, Technical report #151, University of Bergen, Department of Informatics, Norway, February 1998. Submitted as a candidate for the Advanced Encryption Standard. Available at http://www.ii.uib.no/∼larsr/newblock.html

[3] L.R. Knudsen, V. Rijmen: *On the Decorrelated Fast Cipher (DFC) and its Theory.* Fast Software Encryption (FSE'99), Sixth International Workshop, Rome, Italy, March 1999, LNCS 1636, pp. 81-94, Springer, 1999.

[4] M. Luby, C. Rackoff, *How to construct pseudorandom permutations from pseudorandom functions*, SIAM Journal on Computing, vol. 17, n. 2, pp. 373-386, April 1988.

[5] V. Nachef. *Random Feistel schemes for $m = 3$*, available from the author at: Valerie.nachef@math.u-cergy.fr.

[6] Moni Naor and Omer Reingold, *On the construction of pseudo-random permutations: Luby-Rackoff revisited*, J. of Cryptology, vol 12, 1999, pp. 29-66. Extended abstract in: Proc. 29th Ann. ACM Symp. on Theory of Computing, 1997, pp. 189-199.

[7] J. Patarin, *Pseudorandom Permutations based on the DES Scheme*, Eurocode'90, LNCS 514, Springer, pp. 193-204.

[8] J. Patarin, *New results on pseudorandom permutation generators based on the DES scheme*, Crypto'91, Springer, pp. 301-312.

[9] J. Patarin *Etude des générateurs de permutations basés sur le schéma du DES*, Ph. D. Thesis, INRIA, Domaine de Voluceau, Le Chesnay, France, 1991.

[10] J. Patarin *About Feistel Schemes with Six (or More) Rounds*, in Fast Software Encryption 1998, pp. 103-121.

[11] J. Patarin. *About Feistel Schemes with 6 (or More) Rounds. Fast Software Encryption 1998*, pp. 103–121.

[12] J. Patarin. *Generic Attacks on Feistel Schemes. Asiacrypt '01* (Lecture Notes in Computer Science 2248), pp. 222–238, Springer.

[13] J. Patarin *Security of Random Feistel Schemes with 5 or more Rounds*, Extended version of the Crypto '04 paper. This extended version is available from the author or from e-print.

[14] J. Patarin *On linear systems of equations with distinct variables and small block size.* This paper is available from the author or from e-print.

Appendices

A Summary of the known results on random Feistel schemes

KPA denotes known plaintext attacks. CPA-1 denotes non-adaptive chosen plaintext attacks. CPA-2 denotes adaptive chosen plaintext attacks. CPCA-1 denotes non-adaptive chosen plaintext and ciphertext attacks. CPCA-2 denotes adaptive chosen plaintext and chosen ciphertext attacks. Non-Homogeneous properties are defined in [11].
This figure 1 present the best known results against unbounded adversaries limited by m oracle queries.

	KPA	CPA-1	CPA-2	CPCA-1	CPCA-2	Non-Homogeneous
Ψ	1	1	1	1	1	1
Ψ^2	$2^{n/2}$	2	2	2	2	2
Ψ^3	$2^{n/2}$	$2^{n/2}$	$2^{n/2}$	$2^{n/2}$	3	2
Ψ^4	2^n	$2^{n/2}$	$2^{n/2}$	$2^{n/2}$	$2^{n/2}$	2
Ψ^5	2^n	2^n	2^n	2^n	2^n	2
Ψ^6	2^n	2^n	2^n	2^n	2^n	4 *
$\Psi^k, k \geq 6$	2^n	2^n	2^n	2^n	2^n	$\leq \left(\frac{k}{2}-1\right)^2$ **

Figure 1: Minimum number m of queries to distinguish Ψ^k from a random permutation of $I_n \rightarrow I_n$. For simplicity we denote 2^α for $O(2^\alpha)$ i.e. when we have security as long as $m \ll 2^\alpha$.

* ≤ 4 comes from [12] and ≥ 4 comes from [5].
** with k even and with $(k-2)(k-4)$ exceptional equations, so if $k \geq 7$ we need more than one permutation for this property.

	KPA	CPA-1	CPA-2	CPCA-1	CPCA-2
Ψ	1	1	1	1	1
Ψ^2	$2^{n/2}$	2	2	2	2
Ψ^3	$2^{n/2}$	$2^{n/2}$	$2^{n/2}$	$2^{n/2}$	3
Ψ^4	2^n	$2^{n/2}$	$2^{n/2}$	$2^{n/2}$	$2^{n/2}$
Ψ^5	$\leq 2^{3n/2}$	2^n	2^n	2^n	2^n
Ψ^6	$\leq 2^{2n}$	$\leq 2^{2n}$	$\leq 2^{2n}$	$\leq 2^{2n}$	$\leq 2^{2n}$
Ψ^7	$\leq 2^{3n}$	$\leq 2^{3n}$	$\leq 2^{3n}$	$\leq 2^{3n}$	$\leq 2^{3n}$
Ψ^8	$\leq 2^{4n}$	$\leq 2^{4n}$	$\leq 2^{4n}$	$\leq 2^{4n}$	$\leq 2^{4n}$
$\Psi^k, k \geq 6$ *	$\leq 2^{(k-4)n}$	$\leq 2^{(k-4)n}$	$\leq 2^{(k-4)n}$	$\leq 2^{(k-4)n}$	$\leq 2^{(k-4)n}$

Figure 2: Minimum number λ of computations needed to distinguish a generator Ψ^k (with one or many such permutations available) from random permutations with an even signature of $I_n \rightarrow I_n$. For simplicity we denote α for $O(\alpha)$. $\leq$ means best known attack.

* If $k \geq 7$ these attacks analyze about $2^{(k-6)n}$ permutations of the generator. If $k \geq 6$ then $\geq 2^{2n}$ computations are needed: this is shown by a line in Figure 2.