Cryptanalysis of LOKI 91

Lars Ramkilde Knudsen

Aarhus Universitet
Comp. Science Dept.
DK-8000 Århus C.
e-mail:ramlodi@daimi.aau.dk

Abstract. In this paper we examine the redesign of LOKI, LOKI 91 proposed in [5]. First it is shown that there is no characteristic with a probability high enough to do a successful differential attack on LOKI 91. Secondly we show that the size of the image of the F-function in LOKI 91 is $\frac{8}{13} \times 2^{32}$. Finally we introduce a chosen plaintext attack that reduces an exhaustive key search on LOKI 91 by almost a factor 4 using $2^{33} + 2$ chosen plaintexts.

1 Introduction

In 1990 Brown *et al* [4] proposed a new encryption primitive, called LOKI, later renamed LOKI 89, as an alternative to the Data Encryption Standard (DES), with which it is interface compatible. Cryptanalysis showed weaknesses in LOKI 89 [2, 5, 8] and a redesign, LOKI 91 was proposed in [5]. The ciphers from the LOKI family are DES-like iterated block ciphers based on iterating a function, called the F-function, sixteen times. The block and key size is 64 bits. Each iteration is called a round. The input to each round is divided into two halves. The right half is fed into the F-function together with a 32 bit round key derived from the keyschedule algorithm. The output of the F-function is added (modulo 2) to the left half of the input and the two halves are interchanged except for the last round. The LOKI ciphers run 16 rounds. The plaintext is the input to the first round and the ciphertext is the output of the last round. The input to the F-function is the xor'ed value of a 32 bit input text and a 32 bit round key. The 32 bits are expanded to 48 bits and divided into blocks of 12 bits. The 12 bit blocks are the inputs to the 4 S-boxes in LOKI 91, each of which produces an 8 bit output. The 32 bits are permuted making the output of the F-function.

In section 2 we do differential cryptanalysis of LOKI 91 and show that there is no characteristic with a probability high enough to do a successful differential attack. Differential cryptanalysis was introduced by Biham and Shamir [1]. The underlying theory was later described by Lai and Massey [6]. For the remainder of this paper we expect the reader to be familiar with the basic concepts of differential cryptanalysis. Please consult the papers [1, 3, 6] for further details.

In section 3 we examine the size of the image of the F-function, the round function in LOKI 91. Because the key is added to the input text before the expansion in the F-function, the inputs to the 4 S-boxes are dependent. We show that this

has the effect that the size of the image of the F-function is $\frac{8}{13} \times 2^{32}$.
In section 4 we show a weakness in the keyschedule of LOKI 91, i.e. that for every key K there exists a key K^*, such that K and K^* have 14 common round keys. We exploit this weakness in a chosen plaintext attack that reduces an exhaustive key search by almost a factor 4 using $2^{33} + 2$ chosen plaintexts.

2 Differential cryptanalysis of LOKI 91

In [5] it is indicated that LOKI 91 is resistant against differential cryptanalysis, a chosen plaintext attack introduced in [1]. The first thing to do in differential cryptanalysis is to look for good characteristics or differentials. In [3] Biham and Shamir introduced an improved differential attack on DES. The attack shows how to extend an r-round characteristic to an $(r+1)$-round characteristic with unchanged probability by picking the chosen plaintexts more carefully. The cost is a more complex analysis. The improvement can be obtained in attacks on any DES-like iterated cipher. Thus the existence of a 13-round characteristic with a too high probability might enable a successful differential attack on LOKI 91. The probability of an r-round characteristic is found by multiplying the probabilities of r 1-round characteristics. As stated in [6] this way of calculating the probabilities for characteristics requires the cipher to be a Markov cipher. Since the round keys are dependent, LOKI 91 is not a Markov cipher, however tests for LOKI 89 show that the probabilities hold in practice at least for small characteristics [8]. Furthermore we have found no way of incorporating the key dependencies in the calculation of longer characteristics.

2.1 Characteristics for LOKI 91

The best one-round characteristic in LOKI 91 has probability 1 and comes from the fact that equal inputs always lead to equal outputs. A round with equal inputs is called a **zero round** (since the xor-sum of the inputs is zero).
The pairs xor table (see [1]) for LOKI 91 is a table with 2^{20} entries. Table 1 shows the most likely combinations for input/outputxors for one S-box isolated. Note that although inputxor 004_x leads to outputxor 01_x with probability $\frac{132}{4096}$ for one S-box it doesn't mean we can find a one round characteristic with this probability. Because the key is added to the input text before the E-expansion in LOKI 91 the inputs to two neighbouring S-boxes are dependent. In the above case a neighbouring S-box will have inputxor $4ij_x$, where $i, j \in \{0, ..., 15\}$.
The best one-round characteristic with a nonzero input difference has probability $\frac{52}{4096} \simeq 2^{-6.29}$. Therefore to find a 13-round characteristic with a probability high enough to enable a successful differential attack some of the 13 rounds must be zero rounds. The best characteristic for an attack on DES is based on a 2-round iterative characteristic [1], where every second round is a zero round. The best characteristic for an attack on LOKI 89 is based on a 3-round iterative characteristic [5, 8], where every third round is a zero round. We need a few definitions:

Input	Output	Prob. (n/4096)	Input	Output	Prob. (n/4096)
4	1	132	c	1	76
80	4	52	a0	e8	46
173	f7	46	185	90	46
37b	cd	48	3e0	24	48
42a	41	46	498	cf	56
49e	97	46	790	46	50
a20	0	46	a21	d7	48
c43	76	46	c76	f0	48
deb	c9	46	e7b	5f	48
ea6	5d	46	eec	ab	46
f33	e9	46			

Table 1. The most likely combinations from the pairs xor table.

Definition 1 *If the rounds no. $(i-1)$ and $(i+1)$ are zero-rounds, round no. (i) is of* **type A***.*
If the rounds no. $(i-1)$ and $(i+2)$ are zero-rounds, rounds no. (i) and $(i+1)$ are a **pair of type B***.*
If the rounds no. $(i-1)$ and $(i+3)$ are zero-rounds and the rounds no. (i), $(i+1)$ and $(i+2)$ are nonzero rounds, then rounds no. (i), $(i+1)$ and $(i+2)$ are a **triple of type C***.*

A round of type A must have the following form $\phi \rightarrow 0_x$. The best probability of such a round for LOKI 91 is $\frac{122}{2^{20}} \simeq 2^{-13}$ [5].
A pair of rounds of type B must have the following forms,

$$\begin{array}{ll} \text{round no. } (i)\text{:} & \phi \rightarrow \psi \\ \text{round no. } (i+1)\text{:} & \psi \rightarrow \phi \qquad [9] \end{array}$$

Lemma 1 *The highest probability for a pair of rounds of type B is* $(\frac{16}{2^{12}})^2 = 2^{-16}$.

Proof: Consider the case where ϕ and ψ differ in the input to only one S-box each, Si and Sj respectively. Because of the P-permutation (see [4]) it follows easily that the input/outputxor combinations for Si and Sj must have one of the following four forms:

Input	Output	Prob. (n/4096)
080_x	80_x	10
040_x	20_x	16
020_x	08_x	6
010_x	02_x	12

Table 2.

The highest probability for a pair of type B is therefore when the combination in both ϕ and ψ is $040_x \rightarrow 20_x$. It is exactly the situation that occurs for fixpoints [8]. From the pairs xor table it follows easily that if ϕ and ψ differ in the inputs to more than 2 S-boxes totally, we obtain probabilities smaller than 2^{-16}. □

A triple of rounds of type C must have the following forms,

$$\begin{array}{ll} \text{round } (i): & \phi \rightarrow \gamma \\ \text{round } (i+1): & \gamma \rightarrow \phi \oplus \psi \\ \text{round } (i+2): & \psi \rightarrow \gamma \end{array} \qquad [9]$$

Lemma 2 *The probability for a triple of rounds of type C is at most* 2^{-22}.

Proof: By a similar argument as for type B, assume that ϕ, γ and ψ differ in the inputs to only one S-box each. Obviously ϕ and ψ must differ in the inputs to the same S-box. It means that the combination in round $(i+1)$ must have one of the forms from Table 2. The combination in both round (i) and $(i+2)$ must have the following form: $0Y0_x \rightarrow Z$, where $Z \in \{2_x, 8_x, 20_x, 80_x\}$ and $Y \in \{0_x, ..., f_x\}$. The two highest probabilities for $0Y0_x \rightarrow Z$ are $\frac{34}{4096}$ and $\frac{28}{4096}$, therefore the probability of a triple of type C has probability at most

$$\frac{34 * 16 * 28}{2^{36}} < 2^{-22}.$$

Note that $\phi \neq \psi$ otherwise γ would have to differ in the inputs to at least two neighbouring S-boxes [5]. □

Now we can prove the following theorem.

Theorem 1 *There is no 13-round characteristic for LOKI91 with probability higher than* 2^{-63}.

Proof: The best one-round characteristic with a nonzero input difference has probability $\frac{52}{4096} \simeq 2^{-6.29}$. Because $(\frac{52}{4096})^n > 2^{-63} \Rightarrow n \leq 10$, we must have at least 3 rounds with equal inputs in the 13-round characteristic (13R). Since two consecutive zero-rounds force all rounds to be zero-rounds we can have at most 7 zero-rounds.

<u>7 zero-rounds:</u> Every second round is of type A, therefore

$$P(13R) \leq (2^{-13})^6 = 2^{-78}$$

<u>6 zero-rounds:</u> At least 3 rounds are of type A, the remaining 6 rounds can have probability at most $2^{-6.29}$, therefore

$$P(13R) \leq (2^{-13})^3 \times (2^{-6.29})^4 = 2^{-64.2}$$

<u>5 zero-rounds:</u> There can be at most one round of type A, since

$$(2^{-13})^n \times (2^{-6.29})^{8-n} > 2^{-63} \Rightarrow n \leq 1$$

There are two cases to consider

1. No rounds of type A, thereby 4 pairs of type B:

$$P(13R) \leq (2^{-16})^4 = 2^{-64}$$

2. One round of type A, thereby at least 2 pairs of type B:

$$P(13R) \leq 2^{-13} \times (2^{-16})^2 \times (2^{-6.29})^3 = 2^{-63.9}$$

<u>4 zero-rounds:</u> There can be no rounds of type A, since

$$(2^{-13})^n \times (2^{-6.29})^{9-n} > 2^{-63} \Rightarrow n < 1$$

There can be at most one pair of type B, since

$$(2^{-16})^n \times (2^{-6.29})^{9-2n} > 2^{-63} \Rightarrow n \leq 1$$

There are two cases to consider

1. No pairs of type B, thereby 3 triples of type C:

$$P(13R) \leq (2^{-22})^3 = 2^{-66}$$

2. One pair of type B, thereby at least one triple of type C:

$$P(13R) \leq 2^{-16} \times 2^{-22} \times (2^{-6.29})^4 = 2^{-63.2}$$

<u>3 zero-rounds:</u> All 10 nonzero rounds must be based on the best combination $080_x \rightarrow 4_x$, since the second best combination has probability $2^{-6.47}$ and $(2^{-6.29})^9 \times 2^{-6.47} < 2^{-63}$. However it is not possible to construct a 13-round characteristic based solely on the best combination. □

The above does not prove that LOKI 91 is resistant against differential attacks. As stated in [7] to prove this resistance for a DES-like cipher we need to find the best possible differentials. However this seems to be extremely difficult for LOKI 91 and we have done no work in that direction.

3 The F-function of LOKI 91

In the redesign of LOKI89 [5] one of the guidelines was

- to ensure that there is no way to make all S-boxes give 0 outputs, to increase the ciphers security when used in hashing modes.

The 4 S-boxes in LOKI 91 are equal. Each S-box takes a 12 bit input and produces an 8 bit output. Each output value occurs exactly 16 times. The inputs to one S-box that result in a 0 output are listed in Table 1. Because the key is added to the input text before the E-expansion, the input to one S-box is dependent on the inputs to neighbouring S-boxes. Let the input to one S-box be hij_x, then the input to one of the neighbouring S-boxes is jkl_x. From Table 1 we see that to get 0 output from both S-boxes we must have $h, j \in \{0, 5, a, c, f\}$ and $j, l \in \{3, 4, 9, e\}$ leaving no possible values for j. Therefore we cannot get 0

4	49	8e	d3	514	559	59e	5e3
a24	a69	aae	af3	c03	f34	f79	fbe

Table 3. Inputs yielding 0 output for one S-box (hex notation)

outputs from any two neighbouring S-boxes. Let the output from the F-function be $B = \{b_1, b_2, b_3, b_4\}$ where b_i represents the output byte from S-box i. Then $B = \{0, 0, *, *\}$, $B = \{*, 0, 0, *\}$, $B = \{*, *, 0, 0\}$ and $B = \{0, *, *, 0\}$, where '$*$' represents any byte value, are impossible values in the image of the F-function. We found a lot of other impossible values. Therefore we made an exhaustive search for the size of the image of the F-function in LOKI 91.

Theorem 2 *The F-function is not surjective, indeed the size of the image of F is about* $2^{31.3}$.

Note that once we found that $B = \{b_1, b_2, b_3, b_4\}$, where b_i represents the output byte from S-box i, is not in the image of F, then because the 4 S-boxes in LOKI 91 are equal we know that any rotation of the four bytes yields a value not in the image of F. The exact number of impossible values is $1,638,383,180 \simeq \frac{5}{13} \times 2^{32}$. It means that about 5 out of 13 values are never hit in the output of the F-function. In DES we do not have that the inputs to the S-boxes are dependent, because the key is added after the E-expansion. Therefore the size of the image of the F-function in DES is 2^{32}. We have not found any ways to exploit the above observation in an attack on LOKI 91. A consequence of the observation is that the left and right halves of a ciphertext reveals 0.7 bit of information about the inputs (before addition of the keys) to the second last round respectively the third last round of the encryption.

4 A chosen plaintext attack reducing key search

We begin by giving the notation used in this section.
Notation:

- $\overline{X}$ is the bitwise complement of X.
- $Rol_n(X)$ is bitwise rotation of X n positions to the left.
- $E_{16}(P, K)$ is a full 16 round encryption of P using K.
- $E_2(P, K')$ is a 2 round encryption of P using the 32 bit key K' in the first round and $Rol_{13}(K')$ in the second round.
- $Swap(X, Y)$ is the swapping of X and Y.
- $Swap(Z)$ is the swapping of the left and right halves of Z.
- $X||Y$ is the concatenation of X and Y.

The attack we are to describe makes use of a property of the key schedule in LOKI 91. The key size is 64 bits. The key is divided into two 32 bit halves K_L, K_R and the 16 round keys $K(i)$, $i = 1, ..., 16$, are derived as follows:

```
1. $i := 1$
2. $K(i) = K_L$ ; $i = i + 1$
3. $K_L = Rol_{13}(K_L)$
4. $K(i) = K_L$ ; $i = i + 1$
5. $K_L = Rol_{12}(K_L)$
6. $Swap(K_L, K_R)$
7. go to 2.
```

The key schedule allows two different keys to have several round keys in common.

Theorem 3 *For every key K there exists a key K^*, such that K and K^* have 14 round keys in common.*

Proof: Let $K(1), \ldots\ldots, K(16)$ be the roundkeys for $K = K_L || K_R$. Let $K^* = K_R || Rol_{25}(K_L)$. Then $K(2+i) = K^*(i)$, $i = 1, .., 14$. □
If $K = K^*$ Theorem 3 is trivial, but this happens for only two keys, because $K = K^* \Rightarrow (K_L = K_R) \wedge (K_R = Rol_{25}(K_L)) \Rightarrow K_R = K_L = Rol_{25}(K_L) \Rightarrow K = 00....00 \vee K = 11....11$, since $gcd(25, 32) = 1$.

Corollary 1 *There exists 2^{36} pairs of keys, K and K^*, such that K and K^* have 16 round keys in common.*

Let $K = K_L || K_R$, $K_L = hh...hh_x$ for some hex digit h and let K_R be any 32 bits. From Theorem 3 we have a key K^* such that K and K^* have 14 round keys in common and we have furthermore $K^*(15) = Rol_{100}(K_L) = K_L = K(1)$ and $K^*(16) = Rol_{113}(K_L) = Rol_{13}(K_L) = K(2)$, i.e. K and K^* have 16 round keys in common. □
Theorem 3 can be used in a chosen plaintext attack to reduce an exhaustive key search by almost a factor 2. It is well known that the complementation property[1] of DES can be used to reduce an exhaustive key search of DES by a factor 2 in an attack that needs two chosen plaintexts [1]. The complementation property holds also for LOKI 91. This property and Theorem 3 can be used to reduce an exhaustive key search by almost a factor 4 in a chosen plaintext attack that needs $2^{33} + 2$ plaintexts.
Algorithm:

1. Pick $P = P_L || P_R$ at random. Get encryptions C, C^* for P, $\overline{P}$.
2. For all $a \in \{0, 1, ..., (2^{32} - 1)\}$:
   Let $P(a)$ be $E_2(P, a)$. More precisely $P(a) = P_L(a) || P_R(a)$, where
   $$P_L(a) = F(P_R, a) \oplus P_L$$
   $$P_R(a) = F(P_L(a), Rol_{13}(a)) \oplus P_R.$$
3. Get encryptions $C(a)$, $C^*(a)$ for $P(a)$, $\overline{P(a)}$ for all a.
4. Let all keys be non discarded.

[1] Let C be the encryption of P using K, then $\overline{C}$ is the encryption of $\overline{P}$ using $\overline{K}$ as the key.

```
5. Exhaustive search for key:
   For every non discarded key $K = K_L||K_R$, do
   (a) Find $C' = E_{16}(P, K)$
   (b) then
         – if $C' = C$ return $K$ and stop
         – if $C' = \overline{C}^*$ return $\overline{K}$ and stop
         – if $E_2(Swap(C'), Rol_{100}(K_L)) = C(K_L)$
             return $(K_R||Rol_{25}(K_L))$ and stop
         – if $E_2(Swap(C'), Rol_{100}(K_L)) = \overline{C^*(K_L)}$
             return $(\overline{K_R}||Rol_{25}(\overline{K_L}))$ and stop
   (c) Discard the four keys in (b).
```

Upon termination we have found either the secret key or a collision for LOKI 91, i.e. $K \neq K^*$, such that $E_{16}(P, K) = E_{16}(P, K^*)$. Note that in step 5, once we have encrypted P using key $K = K_L||K_R$ without success, we do not have to encrypt P using neither $\overline{K}$, $(K_R||Rol_{25}(K_L))$ nor $(\overline{K_R}||Rol_{25}(\overline{K_L}))$. If one of these three keys is the secret key, then the algorithm would have terminated before. At some points in the algorithm some of the four keys in 5(b) are equal, for example the all zero key will appear twice in the same iteration of step 5. Therefore we cannot find an enumeration of the keys in step 5, s.t. the total no. of iterations of step 5 is exactly one quarter of the size of the key space, i.e. 2^{62}. There exists however an enumeration, s.t. the no. of iterations of step 5 is about $2^{62} + 2^{48}$. It is given in Section 4.1 in every glory detail. Table 4 shows the estimates for space, time and number of chosen plaintexts for the attack, where one time unit is a full 16 round encryption and one space unit is 64 bits. The estimate for *Time* is the number of encryptions made in the analysis. In every iteration of step 5 we do one full 16-round encryption in 5(a). For the two last tests in step 5(b) we do at most 2 rounds of encryption. For most iterations however, we need only to do one round of encryption, because we can test for equality of the right halves of $E_2(Swap(C'), Rol_{100}(K_L))$ and $C(K_L)$ (resp. $\overline{C^*(K_L)}$) already after one round of encryption of $Swap(C')$. If the tests fail we need not do the second round of encryption. Therefore for only about one out of 2^{31} iterations we need to do two rounds of encryption in 5(b). The total amount of time therefore is

Estimates for	Time	Space	Chosen plaintexts
	1.07×2^{62}	$2^{33} + 2$	$2^{33} + 2$

Table 4. Complexity of the chosen plaintext attack

$$(2^{62} + 2^{48}) \times \frac{17}{16} + (\frac{2^{62} + 2^{48}}{2^{31}}) \times \frac{1}{16} \simeq 1.07 \times 2^{62}.$$

Compared to this the time used in step 2 is negligible. The above attack is a weak attack. First of all, it is not very likely that we can get the encryptions for $2^{33}+2$ chosen plaintexts, furthermore an exhaustive search for 2^{62} keys is computationally infeasible. The LOKI cipher is meant as an alternative to DES, with which it is interface compatible. The so far best known attack on DES was introduced by Biham and Shamir in [3]. The attack is a chosen plaintext attack that needs 2^{47} chosen plaintexts. The time used in the analysis phase is 2^{37}. The time needed for the above attack on LOKI 91 is significantly higher than for Biham and Shamirs attack on DES, however the requirements for ever getting to the analysis phase, i.e. the number of encryptions of chosen plaintexts needed, are much higher for the attack on DES.
The steps 2, 3 and 5 can be carried out in parallel, for instance by letting $K_L = a$ in step 5, in that way we don't have to store the 2^{32} $C(a), C^*(a)$'s in step 3. It seems impossible however to obtain an enumeration that at the same time makes the total no. of iterations of step 5 be close to 2^{62} and enables a parallel run of the algorithm.

4.1 Enumeration of the keys in the chosen plaintext attack

We use the same notation as in the previous Sect. Let $\mathcal{A}$ be a function from $GF(2)^{64}$ to itself

$$\mathcal{A} : K_L||K_R \rightarrow K_R||Rol_{25}(K_L)$$

As stated above, once we have tried the key $K = K_L||K_R$ in step 5 of the algorithm without success, we don't have to try the keys

$$\mathcal{A}(K),\ \overline{K},\ \mathcal{A}(\overline{K})$$

The idea is to use $\mathcal{A}$ to construct a set of keys about half the size of the key space and s.t.

- the biggest block of bits in every key consists of 1's.
- for every key K, $\mathcal{A}(K)$ is also in the set.

Then let the enumeration of the keys be every second key from the above constructed set of keys. Later in this Sect. we show that the enumeration obtained this way makes the total no. of keys tried in the attack be very close to 2^{62}.
Let $\mathcal{A}list(K)$ be the set of 64 keys $\{K, \mathcal{A}(K), \mathcal{A}^2(K), ..., \mathcal{A}^{63}(K)\}$. Note that $\mathcal{A}^{64}(K) = K$. Define for $K = K_L||K_R$

$$\mathcal{M}_K = \cup_{p,q}\{\mathcal{A}list(Rol_p(K_L)||Rol_q(K_R)) \cup \mathcal{A}list(Rol_p(K_R)||Rol_q(K_L))\}$$

for $p = 0, 1, 2, 3$ and $q = 0, 8, 16, 24$.

Lemma 3 *For $K = K_L||K_R$, $\mathcal{M}_K$ is the set of all keys of the forms:*

$$Rol_x(K_L)||Rol_y(K_R)$$

$$Rol_x(K_R)||Rol_y(K_L)$$

for all $x, y \in \{0, 1, ..., 31\}$

Proof: For fixed K there are $2 \times 32 \times 32 = 2^{11}$ keys of the above form. Since $\mathcal{A}list$ produces 64 keys, the total no. of keys in $\mathcal{M}_K$ is $2 \times 16 \times 64 = 2^{11}$. Therefore it suffices to show that the pairs of rotations of the keys in $\mathcal{M}_K$ are distinct, i.e. that $Rol_a(K_L)||Rol_b(K_R)$ does not appear twice for any a, b. It is obvious that $Rol_a(K_L)||Rol_b(K_R)$ does not appear twice in one $\mathcal{A}list$. There are two cases to consider, $Rol_a(K_L)||Rol_b(K_R)$ appears in

1. $\underline{\mathcal{A}list(Rol_p(K_L)||Rol_q(K_R)) \text{ and } \mathcal{A}list(Rol_{p'}(K_L)||Rol_{q'}(K_R))}$
   $Rol_a(K_L)||Rol_b(K_R) = Rol_{p+25n}(K_L)||Rol_{q+25n}(K_R) \quad \wedge$
   $Rol_a(K_L)||Rol_b(K_R) = Rol_{p'+25n}(K_L)||Rol_{q'+25n}(K_R) \quad \Rightarrow$
   $p + 25n = p' + 25n \; mod \, 32 \;\wedge\; q + 25n = q' + 25n \; mod \, 32 \quad \Rightarrow$
   $p - p' = q - q' \; mod \, 32 \;\Rightarrow\; (p, q) = (p', q')$,
   since $p - p' \in \{-3, -2, -1, 0, 1, 2, 3\}$ and $q - q' \in \{0, 8, 16, 24\}$.

2. $\underline{\mathcal{A}list(Rol_p(K_L)||Rol_q(K_R)) \text{ and } \mathcal{A}list(Rol_{p'}(K_R)||Rol_{q'}(K_L))}$
   $Rol_a(K_L)||Rol_b(K_R) = Rol_{p+25n}(K_L)||Rol_{q+25n}(K_R) \quad \wedge$
   $Rol_a(K_L)||Rol_b(K_R) = Rol_{q'+25n}(K_L)||Rol_{p'+25+25n}(K_R) \quad \Rightarrow$
   $p + 25n = q' + 25n \; mod \, 32 \;\wedge\; q + 25n = p' + 25 + 25n \; mod \, 32 \quad \Rightarrow$
   $p + p' + 25 = q + q' \; mod \, 32$
   A contradiction, since $p + p' + 25 \in \{25, 26, ..., 31\}$ and $q + q' \in \{0, 8, 16, 24\}$.

□

Let Ka and Kb be two 32-bit keyhalves, s.t. Ka and Kb are no rotations of each other, i.e. $Rol_x(Ka) \neq Kb$ for any x, $0 < x < 32$. For $K = Ka||Kb$, $\mathcal{M}_K$ is a set of distinct keys except in the cases where $Rol_x(Ka) = Ka$ and/or $Rol_y(Kb) = Kb$.

Lemma 4 *Let H be a 32-bit key and $Rol_n(H)$ any rotation to the left of H, where $0 < n < 32$. Then there are $2^{gcd(n,2)}$ possible values of H, such that $H = Rol_n(H)$.*

From Lemma 4 it follows for $K = K_L||K_R$, where K_L and K_R are no rotations of each other, that

Lemma 5 *There at most 2^{49} keys for which the elements in $\mathcal{M}_K$ are not distinct.*

Proof: Assume we have two equal keys K' and $K*$ from $\mathcal{M}_K$. Then

$$K' = Rol_a(K_L)||Rol_b(K_R), \; K* = Rol_c(K_L)||Rol_d(K_R)$$

Clearly from the proof of Lemma 1 $(a, b) \neq (c, d)$. Then

$$Rol_a(K_L) = Rol_c(K_L) \wedge Rol_b(K_R) = Rol_d(K_R) \Rightarrow$$
$$Rol_{a-c}(K_L) = K_L \wedge Rol_{b-d}(K_R) = K_R$$

If $a = c$ then there are 2^{32} possible values for K_L, but then there are at most 2^{16} possible values for K_R according to Lemma 4, since $(a, b) \neq (c, d)$. If $b = d$

then $a \neq b$ and we get a total no. of $2 \times 2^{32} \times 2^{16} = 2^{49}$ keys. □

The following algorithm makes a list of 32 bit strings, where no two strings are rotations of each other and where the biggest block of bits in every string consists of 1's.

ALGORITHM - No-rotations-of-keys (NRK)

For all positive $k \leq 32$, list all k-tuples $(a_1, a_2, ..., a_k)$, s.t.

1. $\sum_{i=1}^{k} a_i = 32$
2. $a_i \geq 1$ for $0 < i \leq k$
3. $\sum_{i=1}^{k} a_i \times 32^{k-i} \geq \sum_{i=1}^{k} a_{i+n\, mod\, (k+1)} \times 32^{k-i}$, for all $n \leq k$.

Method: For every k-tuple $(a_1, ..., a_k)$ output the 32-bit key, where the a_1 MSB are 1-bits, the next a_2 bits are 0-bits and so on.

Lemma 6 *No two keys in the output from (NRK) are rotations of each other.*

Proof: Because of the inequality in 3. above if $k > 1$, then k is even. Therefore for $k > 1$ the a_k LSB are 0-bits and furthermore $a_1 \geq a_i$ for $i \leq k$.

Let A and A' be two 32 bit keys from (NRK), s.t. $A = Rol_x(A')$ for some fixed x. Write A and A' as tuples $(a_1, ..., a_m)$ and $(a'_1, ..., a'_l)$ according to the method in (NRK). Clearly $l = m$ otherwise A cannot be a rotation of A'. Because $A = Rol_x(A')$ we have for some i

$$a_{1+n} = a'_{i+n\, mod\, (m+1)}, \quad 0 < n \leq m$$

Especially we have $a_1 = a'_i$ and $a'_1 = a_{m-i+2}$. Because of the inequality in 3. above we have

$$a'_i \leq a'_1 \ \wedge \ a_1 \geq a_{m-i+2}$$

Therefore $a'_i = a'_1 \Rightarrow a_1 = a'_1$. Now $a_1 = a_{m-i+2} \Rightarrow a_2 \geq a_{m-i+3}$. Similar as before

$$a_2 = a'_{i+1} \leq a'_2 = a_{m-i+3} \Rightarrow a'_2 = a_2$$

By induction we obtain $A = A'$ □

ALGORITHM - Enumeration (EN)

```
1. $i = 1$
2. Let $K_L$ be the $i$'th output from (NRK).
3. For $j = 1$ to $i$ do
   (a) Let $K_R$ be the $j$'th output from (NRK)
   (b) For $K = K_L || K_R$ output the first and then every second key from all
       $\mathcal{A}lists$ in $\mathcal{M}_K$
   (c) For $K = \overline{K_L} || K_R$ do as in 3b
4. $i = i + 1$, goto 1.
```

We are left to check whether the set

$$KS = \cup_{Ki}\{Ki, \mathcal{A}(Ki), \overline{Ki}, \overline{\mathcal{A}(Ki)}\}$$

where the $Ki's$ are the keys output from (EN), contains the entire keyspace.
Let $K^* = K^*_L || K^*_R$ be an arbitrary key. Rotate K^*_L and K^*_R such that the biggest blocks of bits (0's or 1's) are the MSB. Let $K(j) = K'_L || K'_R$ be that key.
If the MSB in both K'_L and K'_R are 1's then they are both output from (NRK). Then at some point $K(j)$ or $K(l) = K'_R || K'_L$, say $K(j)$, are the key considered in step 3(b) of (EN). Let $K(n)$, $0 < n \leq 2^{10}$ be all keys output in step 3(b) when $K = K(j)$. Then $L = \{K(n), \mathcal{A}(K(n))\}$, $0 < n \leq 2^{10}$ are all rotations of the key halves in $K(j)$ according to Lemma 3. Therefore $K^* \in L \in KS$.
If MSB in both K'_L and K'_R are 0's, then at some point either $\overline{K(j)}$ or $\overline{K(l)}$ are the key considered in step 3(b). Let $K(n)$ be as before, when $K = \overline{K(j)}$. Then $L = \{K(n), \mathcal{A}(K(n))\}$, $0 < n \leq 2^{10}$ are all rotations of the key halves in $K(j)$ according to Lemma 3. Therefore $\overline{K^*} \in L \in KS \Rightarrow K^* \in \overline{L} \in KS$.
If the MSB in K'_L and K'_R are 1's and 0's resp. or vice versa similar arguments hold for step 3(c).
We have implemented (NRK) on a SUN-Sparc workstation. The number of key halves output from (NRK) is $2^{26}+2068$. It means that the number of keys output from (NRK) in 2. and 3(a) above is about $2^{51} + 2^{37}$. Every second key from $\mathcal{M}_K$ gives 2^{10} keys for each K. The total number of keys in the enumeration therefore is about

$$(2^{51} + 2^{37}) \times 2 \times 2^{10} = 2^{62} + 2^{48}.$$

We have given an enumeration of the keys, s.t. the total no. of iterations of step 5 in the algorithm of the chosen plaintext attack is close to 2^{62}. The time used in the enumeration (EN) is negligible compared to the 1.07×2^{62} full 16 rounds of encryption of LOKI 91, since it runs only one time per every 2×2^{10} runs of step 5 in the algorithm of the chosen plaintext attack.

5 Conclusion and open problems

We have shown that we cannot find a characteristic for LOKI 91 good enough to do a successful differential attack on LOKI 91. Still it is not enough to conclude that LOKI 91 is secure against this kind of attack. To do that we need an efficient way of calculating the probabilities of differentials.
We have shown that the size of the image of the F-function in LOKI 91 is only $\frac{8}{13}$ of the size of the image of the F-function in DES. We have found no way of exploiting this fact in an attack on LOKI 91. Whether it represents a weakness of the algorithm is left as an open question.
Finally we introduced a chosen plaintext attack on LOKI 91 that reduces an exhaustive key search by almost a factor 4. The attack exploits a weakness in the key schedule of LOKI 91. It might also be possible to use this weakness, the common round key property, to find collisions for LOKI 91 when used as a hash function. This is left as an open question.

6 Acknowledgements

We wish to thank Prof. Ivan Damgård for valuable discussions on the enumeration of the keys in the chosen plaintext attack.

References

1. E. Biham, A. Shamir. *Differential Cryptanalysis of DES-like Cryptosystems.* Journal of Cryptology, Vol. 4 No. 1 1991.
2. E. Biham, A. Shamir. *Differential Cryptanalysis of Snefru, Khafre, REDOC-II, LOKI and Lucifer.* Extended abstract appears in Advances in Cryptology, proceedings of CRYPTO 91.
3. E. Biham, A. Shamir. *Differential Cryptanalysis of the full 16-round DES.* Technical Report # 708, Technion - Israel Institute of Technology.
4. L. Brown, J. Pieprzyk, J. Seberry. *LOKI - A Cryptographic Primitive for Authentication and Secrecy Applications.* Advances in Cryptology - AUSCRYPT '90. Springer Verlag, Lecture Notes 453, pp. 229-236, 1990.
5. L. Brown, M. Kwan, J. Pieprzyk, J. Seberry. *Improving Resistance to Differential Cryptanalysis and the Redesign of LOKI.* Abstracts from ASIACRYPT'91.
6. X. Lai, J. L. Massey, S. Murphy. *Markov Ciphers and Differential Cryptanalysis.* Advances in Cryptology - Eurocrypt '91. Lecture Notes in Computer Science 547, Springer Verlag.
7. K. Nyberg, L. Ramkilde Knudsen. *Provable Security Against Differential Cryptanalysis.* Presented at the rump session of CRYPTO'92. To appear in the proceedings of CRYPTO'92.
8. L. Ramkilde Knudsen. *Cryptanalysis of LOKI.* Abstracts from ASIACRYPT'91.
9. L. Ramkilde Knudsen. *Iterative Characteristics of DES and s^2-DES.* To appear in the proceedings from CRYPTO'92.

This article was processed using the LaTeX macro package with LLNCS style