Improved Differential Attacks on RC5

Lars R. Knudsen*[1]
Willi Meier**[2]

[1] K.U. Leuven,Dept. Elektrotechniek-ESAT, Kard. Mercierlaan 94, B-3001 Heverlee
[2] HTL Brugg-Windisch, CH-5200 Windisch

Abstract. In this paper we investigate the strength of the secret-key algorithm RC5 newly proposed by Ron Rivest. The target version of RC5 works on words of 32 bits, has 12 rounds and a user-selected key of 128 bits. At Crypto'95 Kaliski and Yin estimated the strength of RC5 by differential and linear cryptanalysis. They conjectured that their linear analysis is optimal and that the use of 12 rounds for RC5 is sufficient to make both differential and linear cryptanalysis impractical. In this paper we show that the differential analysis made by Kaliski and Yin is not optimal. We give differential attacks better by up to a factor of 512. Also we show that RC5 has many weak keys with respect to differential attacks. This weakness relies on the structure of the cipher and not on the key schedule.

Keywords. Cryptanalysis. Block Cipher. Differential cryptanalysis. Weak keys.

1 Introduction

RC5 is a secret-key block cipher proposed by Ron Rivest [5]. RC5 has a variable word size, a variable number of rounds and a variable length of the key. The "nominal" choice of parameters is 32 bits words, 12 rounds and a 16 bytes key, referred to as RC5-32/12/16. A novel feature of the algorithm is the use of data-dependent rotations. The security of RC5 relies on the rotation operation and the mixed use of xor and addition of words. Kaliski and Yin evaluated RC5 with respect to differential and linear cryptanalysis [2]. It was shown that linear cryptanalysis is applicable only for versions of RC5 with a small number of rounds. Also, it was conjectured that the linear approximations in the analysis were optimal and that the use of 12 rounds for RC5 is sufficient to make both differential and linear cryptanalysis impractical. In this paper we show that the differential analysis made by Kaliski and Yin is not optimal. In our attacks we exploit the data-dependent rotations to speed up a differential attack. The idea is to choose and find plaintexts so that there are no rotations in the first few rounds. Once these plaintexts have been identified a differential attack can be

* email: knudsen@esat.kuleuven.ac.be
** email: meierw@htl-bw.ch

performed with differentials of higher probability. Our differential attacks are better than the known attacks by up to a factor of 512. Also, by a closer look at the differential attacks of RC5 one finds that there exist keys for which the attacks perform even better. This is somewhat surprising since RC5 has a very complex key schedule, but, as we will see, the existence of weak keys is not due to the key schedule itself.

In the following we use the description of RC5 from [2]. Let (L_0, R_0) denote the left and right halves of the plaintext, respectively, and let S_i be the ith subkey. Then the ciphertext (L_{2r+1}, R_{2r+1}) is defined by

$$
\begin{array}{l}
L_1 = L_0 + S_0 \\
R_1 = R_0 + S_1 \\
\textbf{for } i = 2 \textbf{ to } 2r+1 \textbf{ do} \\
\qquad L_i = R_{i-1} \\
\qquad R_i = ((L_{i-1} \oplus R_{i-1}) << R_{i-1}) + S_i
\end{array}
$$

where $(\alpha << \beta)$ is the rotation of α by $(\beta \bmod w)$ positions to the left. Thus the rotation amount is the value of the $lg(w) \stackrel{\text{def}}{=} \log_2 w$ least significant bits of R_{i-1}. The two equations with L_i and R_i on the left sides will be called a *half-round*. The two initial equations are called the first half-round. For a description of the key schedule we refer to [5]. In the following we will assume that the subkeys produced by the key schedule are uniformly random. This is a reasonable assumption for what we are going to prove which will be illustrated. This paper is organised as follows. In Sect. 2 we first review the attacks by Kaliski and Yin and introduce our improved differential attack applicable for all versions of RC5 in Sect. 3. In Sect. 4 it is shown that RC5 has many weak keys with respect to differential attacks. We conclude and discuss our work in Sect. 5.

2 Differential Attacks

We give first a short description of the differential attack by Kaliski and Yin and refer to [2] for more details.

Definition 1. The difference between two bit-strings X and X^* of equal length is defined to be $\Delta X = X \oplus X^*$, i.e. the exclusive-OR. Also, we define e_s to be the w-bit vector having a one in position s and zeros everywhere else.

The basic idea in the attack is to compute certain bits of L_{2r}, which can then be used to deduce information about the subkey S_{2r+1}. Since $L_{2r} = R_{2r-1}$, knowledge about the rotation amount in the second-last half-round gives the desired information. This knowledge can be obtained by observing which bits are set in the differences of the two ciphertext halves. Once the key S_{2r+1} has been found the intercepted ciphertexts can be decrypted by one half-round and a similar and easier analysis performed on less rounds of RC5.

Denote a differential for one half-round by $\Omega = (\Omega_P, \Omega_T)$, where $\Omega_P = (\Delta L_{i-1}, \Delta R_{i-1})$ and $\Omega_T = (\Delta L_i, \Delta R_i)$. Let p^{Ω} denote the probability of the

Ω	Ω_P	Ω_T	Conditions	Probability
Ω^1	$(0, e_s)$	(e_s, e_s)	$s \geq lg(w)$	$p \geq (1/2w)$
Ω^2	(e_s, e_s)	$(e_s, 0)$	$s \geq lg(w)$	$p = 1$
Ω^3	$(e_s, 0)$	$(0, e_t)$	$s, t \geq lg(w)$	$p \geq (1/2w)$
Ω^4	$(0, e_s)$	(e_s, e_t)	$s, t \geq lg(w),\ t \neq s$	$p \geq (1/2w)$
Ω^5	(e_s, e_t)	$(e_t, e_u \oplus e_v)$	$s, t \geq lg(w),\ t \neq s,\ u > v$ $t - s = \pm(u - v)$	$p \geq (1/4w)$
Ω^6	$(e_t, e_u \oplus e_v)$	$(e_u \oplus e_v, e_x \oplus e_y \oplus e_z)$	$u, v \geq lg(w)$	$p \geq (1/8w)$

Table 1. Useful half-round differentials of RC5 [2].

differential Ω. The half-round differentials of Table 1 are of special interest to us as we will see. The first three half-round differentials can be concatenated to obtain an iterative differential, i.e. a differential over three half-rounds that can be concatenated with itself. The differentials Ω^4 and Ω^5 are suitable for obtaining the desired information about L_{2r} used to deduce the key bits. The differential Ω^6 will be used later in our improved attack.

Table 2 lists the probabilities of differentials to be used in attacks on RC5 with any number of rounds. $\bar{\Omega}$ denotes the concatenation of Ω^1, Ω^2, and Ω^3. The differential $\Omega^{k'}$ in Table 2 is the same as the differential Ω^k, except it is used in the first half-round only and will have probability one. We note that the differentials found by Kaliski and Yin are differentials by the definition of Lai, Massey and Murphy [3]. The original concept of *characteristics* by Biham and Shamir [1] predicts one specific value of the ciphertext difference after each round of a cipher, while in differentials, as used here, the intermediate ciphertext difference can take on several values. Thus, there are many characteristics contained in the differentials of Table 2.

$2r+1$	ΔP	Ω_{2r+1}	$p^{\Omega_{2r+1}}$
$3m$	$(0, e_{w-1})$	$\Omega^{1'}, \bar{\Omega}, ..., \bar{\Omega}, \Omega^4, \Omega^5$	$(\frac{w-lg(w)-1}{w})(\frac{w-lg(w)}{(2w)^2})^{m-1}$
$3m+1$	$(e_{w-1}, 0)$	$\Omega^{3'}, \Omega^3, \bar{\Omega}, ..., \bar{\Omega}, \Omega^4, \Omega^5$	$(w - lg(w) - 1)(\frac{w-lg(w)}{(2w)^2})^m$
$3m+2$	(e_{w-1}, e_{w-1})	$\Omega^{2'}, \Omega^2, \Omega^3, \bar{\Omega}, ..., \bar{\Omega}, \Omega^4, \Omega^5$	$(w - lg(w) - 1)(\frac{w-lg(w)}{(2w)^2})^m$

Table 2. r-round differentials for RC5 and their probabilities [2].

Example 1. Consider 6-round RC5 and the differential $(e_{w-1}, 0)$, see Table 2. In the first half-round the probability is 1. For Ω^3 there are $(w - lg(w))$ possible values for t, thus the probability in the second half-round is $(w - lg(w))/2w$. The next three occurrences of $\bar{\Omega}$ have probability each $(w - lg(w))/(2w)^2$ since there is only one possible value for s in Ω^1 and $(w - lg(w))$ possible values for t in Ω^3. The second-last half-round, using Ω^4, has probability $(w - lg(w) - 1)/2w$ and the last half-round, using Ω^5, has probability 1, since there are w possibilities for u, v and the factor 4 can be eliminated [2].

The number of pairs required for a successful differential attack is estimated to be about $2w \times 1/p^{\Omega}$ for $r \leq 11$ and $8w \times 1/p^{\Omega}$ for $r = 12$ [2].

We close this section by commenting on the modified version of RC5, where all additions are changed to exclusive-or, considered by Kaliski and Yin [2]. First note that the parity bit of the plaintext exclusive-or'ed to the parity bit of all subkeys equals the parity bit of the ciphertext. So given one plaintext-ciphertext pair we get one bit of information about the subkeys and thereby one bit of information about the plaintexts from all further intercepted ciphertexts. This version of RC5 is therefore weak.

3 Our Differential Attacks

The first observation in our improvement of the differential attack is, that if $R_1 = 0 \bmod w$, hereafter denoted $R_1 =_w 0$, there will be no rotation in the second half-round. Consider Example 1 again. If there is no rotation in the second half-round the probability is 1, since it holds that $x \oplus y = e_{w-1} \Rightarrow (x+S_2) \oplus (y+S_2) = e_{w-1}$. In a similar manner, if $R_2 =_w 0$ there will be no rotation in the third half-round. More precisely, if we choose

$$L_0 =_w w - S_0 - S_2 \tag{1}$$

$$R_0 =_w w - S_1 \tag{2}$$

then we get

$$\begin{aligned} L_1 &=_w L_0 + S_0 =_w w - S_2 \\ R_1 &=_w R_0 + S_1 =_w 0 \\ R_2 &=_w (L_1 \oplus R_1) << R_1) + S_2 \\ &=_w (w - S_2) << 0) + S_2 =_w 0. \end{aligned}$$

In this way there will be no rotations in the second and third half-rounds. For the differential in Example 1 this means that if (1) and (2) holds then the probability of the first four half-rounds is one. Since the keys S_0, S_1, S_2 are unknown to an attacker, he does not know the solution to equations (1) and (2). However, he can construct differentials for all $w \times w$ possible values of the $lg(w)$ least significant bits of both L_0 and R_0 in turn and observe the probabilities for each value. The idea is that for the values satisfying equations (1) and (2) the probability of the differential will be higher than for other values. At a first glance it may seem that we will need more pairs than for the differential attack by Kaliski-Yin. But there are two advantages in our approach. For the values satisfying (1) and (2)

- the differential Ω will have a higher probability, and
- we will need fewer than $2w/p^{\Omega}$ pairs for success.

The plan for our extended differential attack is as follows.

1. **Subkey detection.** For all values of the $lg(w)$ least significant bits of both plaintext halves, construct differentials and observe their probabilities. Determine the values of L_0 and R_0 satisfying (1) and (2), i.e. determine $2 \times lg(w)$ key bits.
2. **Improved differential attack.** Perform the differential attack by Kaliski-Yin [2] with increased performance.

From this it is obvious that our differential attack is improved only if the total amount of pairs needed in the key detection part is less than the amount of pairs needed in the attack by Kaliski and Yin.

3.1 A Basic Key Detection Algorithm

We split the key detection algorithm into two parts. In the first part we will determine the values of the right halves of the plaintexts satisfying equation (2). In the second part we will determine the values of the left halves of the plaintexts satisfying equation (1).

For the first part the difference in the plaintexts will be $(0, e_{w-1})$, thus the texts to be rotated in the second half-rounds have difference e_{w-1}. If there is no rotation, the difference after the second half-round is (e_{w-1}, e_{w-1}). In the third half-round the texts to be rotated have difference zero, thence here the probability of the differential is one, whether or not there is a rotation. We will need to create differentials for the w different values of the $lg(w)$ least significant bits of the right halves of the plaintexts. On the other hand, for the right value of the plaintexts the probability of the differential is improved by a factor of w compared to the estimate[3] in Table 2. Furthermore, for pairs of plaintexts not satisfying equation (2) there will be a rotation in the second half-round, which means that the amounts to be rotated in the third half-round for the pair are not equal, which again means that the pair is a *wrong* pair, i.e. it does not follow the expected values in the differential. Therefore we need only about one right pair for success instead of $2w$ pairs in the differential attack. If there are right pairs for more than one of the w values of the right halves of the plaintexts, further pairs are generated to detect the correct values.

In Table 3 we list the differentials used in the first part of the key detection algorithm and their probabilities for the plaintexts satisfying equation (2). For $2r + 1 = 3m$ the probability of the differential is a factor of w higher than for the full differential attack in [2]. We generate pairs for w different groups of plaintexts, but need only about one right pair. Totally this part of the key detection algorithm needs about a factor of $w/w \times 2w = 2w$ less pairs compared to the estimates for the differential attack. For $2r + 1 = 3m + 1$ this factor is $2(w - lg(w))$ and for $2r + 1 = 3m + 2$ the factor is $\frac{4w}{(w-lg(w)-1)}$, which can easily be seen by comparing the probabilities of Tables 2 and 3. For $w = 32$ these factors are 64, 54, and 5, respectively. The improvement is highest for

[3] Note that the first occurrence of the differential Ω^1 has probability $1/w$ as noted in [2].

$2r+1=3m$, since the differential we use in the key detection is optimal for the differential attack, whereas in the other cases other differentials are optimal in the differential attack. E.g. for $2r+1=3m+2$ the differential used in the differential attack optimizes the use of half-rounds with zero differences, i.e. with probability one.

Let us explain in more detail the differential for key detection in the case of $2r+1=3m+1$. The probabilities in the second, third, and fourth half-rounds are one, one and $(w-lg(w))/2w$, respectively. Hereafter follow $m-2$ occurrences of $\bar{\Omega}$ each of probability $\frac{w-lg(w)}{(2w)^2}$ and one occurrence of Ω^4 with probability $\frac{w-lg(w)-1}{2w}$. In the second-last half-round, using Ω^5, the inputs to be rotated have difference $e_s \oplus e_t$, where $s,t \geq lg(w)$. We require that $u,v \geq lg(w)$, such that there are equal rotations in the last half-round. This happens with estimated probability $\binom{w-lg(w)}{2}/\binom{w}{2}$, which together with the additional factor of 4 give the desired result. Note that in the last half-round there will be w possibilities for x,y,z and that the factor of 8 can be mostly eliminated in the same way as the factor of 4 in the differential attack of [2].

$2r+1$	$\tilde{\Omega}_{2r+1}$	$p^{\tilde{\Omega}_{2r+1}}$
$3m$	$\Omega^{1'}, \bar{\Omega}, ..., \bar{\Omega}, \Omega^4, \Omega^5$	$(w-lg(w)-1)\cdot(\frac{w-lg(w)}{(2w)^2})^{m-1}$
$3m+1$	$\Omega^{1'}, \bar{\Omega}, ..., \bar{\Omega}, \Omega^4, \Omega^5, \Omega^6$	$(\frac{w-lg(w)}{(2w)^2})^{m-1}\cdot\frac{(w-lg(w))^2}{2w}\cdot\frac{w-lg(w)-1}{2(w-1)}$
$3m+2$	$\Omega^{1'}, \bar{\Omega}, ..., \bar{\Omega}, \Omega^4$	$(\frac{w-lg(w)}{2w})\cdot(\frac{w-lg(w)}{(2w)^2})^{m-1}$

Table 3. The differentials with $\Delta P=(0,e_{w-1})$ for key detection.

For the second part of our algorithm we use the differential for $3m+1$ in Table 2. The difference in the plaintexts is $(e_{w-1},0)$, so the texts to be rotated in the second half-round have difference e_{w-1}. In the first part of the algorithm we found the value of the right halves of the plaintexts, so that there is no rotation here. In the third half-round the texts to be rotated therefore also have difference e_{w-1}. If L_0 satisfies equation (1) there will be no rotation and the difference to be rotated in the following half-round have difference zero. Thus, for the plaintext values satisfying equations (1) and (2) the first four half-rounds of the differential are always satisfied. Therefore the complexity of this part of the algorithm will be lower than for the first part of the key detection algorithm.

For $w=32$, as proposed by Rivest, the estimated number of pairs needed in the key detection algorithm in order to determine the values of the plaintexts satisfying equations (1) and (2) are given in the second column of Table 4. In the following we show how to decrease the complexity of this algorithm.

3.2 Extensions of the key detection algorithm

In this subsection we extend the key detection algorithm and give experimental evidence. In order to detect the right values of the $2\times lg(w)$ subkey bits more efficiently, we shall consider more general output differences than those in the differential attacks of [2]. In addition, considering these more general differences, we can experimentally detect the right values of the subkey bits for up to nine rounds of RC5. This is motivated by reasonings which also give some insight into the interaction of the three basic operations $+$, $\oplus$ and $<<$ used in the design of RC5.

Our first observation concerns a relation between bit differences and integer addition. Recall that the (constant) key words S_i enter each half-round by integer addition. Integer addition of a constant word S to words A and B which only differ in few bits does not necessarily lead to an increase of bit differences in the sums $A+S$ and $B+S$. This may be illustrated by the following special case:

Suppose the words A and B only differ in the i-th bit, $i < w-1$. It is shown in [2] that with probability $\frac{1}{2}$, $A+S$ and $B+S$ also differ in only the i-th bit. If we use the binary representation of words, i.e. $A = a_{w-1}2^{w-1} + \cdots + a_1 2 + a_0$, and similarly for B and S, the binary representation of the sum $Z = A+S$ may be obtained by the formulae

$$z_j = a_j + s_j + \sigma_{j-1} \text{ and } \sigma_j = a_j s_j + a_j \sigma_{j-1} + s_j \sigma_{j-1}, \tag{3}$$

where σ_{j-1} denotes the carry bit and $\sigma_{-1} = 0$ (cf. [4]). Using these formulae one sees that $A+S$ and $B+S$ with probability $\frac{1}{4}$ differ in exactly two (consecutive) bits.

Suppose now the words A and B already differ in exactly two consecutive bits. Then again using the formulae (3) one can see that with probability $\frac{1}{4}$, $A+S$ and $B+S$ differ in exactly one bit and that with probability $\frac{3}{8}$, $A+S$ and $B+S$ differ in exactly two (not necessarily consecutive) bits. Thus with probability $\frac{5}{8}$ the words $A+S$ and $B+S$ differ again in at most two bits if A and B differ in two consecutive bits. Using the formulae (3) one could discuss relations between integer addition and bit differences in a more general setting. However this special case suggests that addition of the key words in each half-round can only moderately contribute to an avalanche effect of bit differences.

Our second comment concerns a relationship between the rotation $<<$ and bit differences in RC5. The avalanche of bit differences in a half-round is expected to be strongest if bits differ in the last $lg(w)$ positions of R_{i-1}, i.e., if different R_{i-1}'s cause different rotations. All the differentials considered in [2] (see also Table 1) refer to differences which escape this (full) rotation effect. If the words differ in only one bit, the probability for this to happen is $\frac{w-lg(w)}{w}$. The more bits are different, the more this probability is reduced. However even for a bit difference of up to eight bits this probability for $w = 32$ is evaluated to be at least 0.21. Thus, differences with up to eight bits different per word escape the full rotation effect with non negligible probability.

These reasonings have motivated to consider output differences with Hamming weights larger than one or two, thus extending the differentials Ω^4 and Ω^5. An estimate for probabilities of such sequences of differences is no longer obvious. But starting with the differences Ω^1 or Ω^3 we may expect a non negligible fraction of sequences of half-rounds for which the initial bit difference propagates in a way such that the carry effect caused by addition of key words is only moderate and where all intermediate differences escape the full rotation effect. Referring to the description of a half-round, in such a situation the Hamming weights of the differences per word propagate roughly like a Fibonacci sequence, i.e., the subsequent Hamming weights of differences in a half-round may be estimated by the sequence 0, 1, 1, 2, 3, 5, 8, 13,...

Thus for consecutive numbers m, n in this sequence we may consider output pairs (L_i, R_i), (L_i^*, R_i^*) whose differences have Hamming weight at most m in the left, and at most n in the right word. Moreover it turns out to be essential only to use output pairs where the $lg(w)$ least significant bits of L_i and L_i^* agree, as otherwise the Hamming weight of the difference in the right words tends to be random as affected by different rotation amounts. We denote such a difference by $\Omega^{m,n}$, $m > 1$, and we expect that the probability for such an output difference is higher than for the output difference determined by Ω^5.

For the first part of the subkey detection the difference in plaintexts is $(0, e_{w-1})$. The strategy is to create differentials for w different values of the right halves of the plaintexts. Our hypothesis is that for the correct value of the $lg(w)$ least significant bits of the right halves of the plaintexts the probability of the output difference $\Omega^{m,n}$ is maximized. For the second part of the subkey detection the difference in plaintexts is $(e_{w-1}, 0)$. The strategy is to use the correct values of the right halves of the plaintexts found in the first part of the algorithm and create differentials for w different values of the left halves of the plaintexts. We subsume our experimental results as follows ($w = 32$). We implemented the tests searching for the correct values in both the left and right halves of the plaintexts for versions with $r < 8$ and we chose as output differences $\Omega^{3,5}$, $\Omega^{5,9}$, and $\Omega^{8,15}$ (thus allowing for one resp. two carry bits in the right words for the second and third differences). For versions with 8 and 9 rounds we searched only for the correct values of the right halves of the plaintexts, i.e. doing only the first part of the above test. Table 4 lists the number of plaintexts required to obtain a 90% success rate for the extended key detection algorithm for versions of RC5 up to 9 rounds. From these numbers we estimated the complexities of RC5 with 10, 11, and 12 rounds. As can be seen from the numbers in Table 4, the extended key detection algorithm is substantially better than the basic algorithm.

3.3 Improved Differential Attack

Once we have detected the right values of the $2 \times lg(w)$ subkey bits we will perform the differential attack described by Kaliski and Yin [2]. The types of differentials used in the attacks depend on the number of rounds of RC5 considered. There are three different differentials depending on the value $2r + 1 \bmod 3$ when r-round RC5 is attacked, as noted in Table 2. This stems from the fact that using Ω^4 and Ω^5 in the last two half-rounds enables us to determine the key of the last half-round. In the following we will use the same types of differentials as used by Kaliski-Yin and determine the factors we save in the number of pairs needed for a successful differential attack. If $2r + 1 = 3m + 1$ the differential has nonzero differences in the second and third half-rounds. With the key detections

Rounds	Basic	Extended
4	2^{16}	2^{12} (*)
5	2^{22}	2^{17} (*)
6	2^{26}	2^{22} (*)
7	2^{31}	2^{27} (*)
8	2^{37}	2^{32} (*)
9	2^{40}	2^{37} (*)
10	2^{45}	2^{42} (**)
11	2^{51}	2^{47} (**)
12	$2^{54.5}$	2^{53} (**)

Table 4. Number of chosen plaintexts needed for the basic and the extended key detection algorithms for $w = 32$. (*) Confirmed by experiments. (**) Estimated.

the probabilities in these half-rounds will be one, and it is straightforward to see that the saving factor is $2w \times 2w/(w - lg(w))$. If $2r+1 = 3m$ we save a factor of w in the second half-round, but nothing in the third half-round, since the texts to be rotated are equal anyway. But if the subkey $S_3 =_w 0$, there will be no rotation in the fourth half-round. This follows from $R_3 = ((R_1 \oplus R_2) << R_2) + S_3$, since it holds that $R_1 =_w R_2 =_w 0$. Therefore, for one out of w keys we save an additional factor of $2w/(w - lg(w))$. If $2r + 1 = 3m + 2$ the texts to be rotated in the second half-round have difference zero, so there is no immediate improvement here, but in the third half-round we will save a factor of $2w/(w - lg(w))$. If $S_3 =_w 0$ we save an additional factor of $2w$, for reasons similar as in the previous case. Table 5 shows the improvement factors of a differential attack for various numbers of rounds after the application of the key detection algorithm. We can

$2r+1$	All keys	1 in w keys
$3m$	w	$2w^2/(w - lg(w))$
$3m+1$	$4w^2/(w - lg(w))$	$-$
$3m+2$	$2w/(w - lg(w))$	$4w^2/(w - lg(w))$

Table 5. Improvement factors of the differential attacks.

now estimate the full complexity of our differential attacks on RC5. Table 6 lists the results of Kaliski-Yin [2] and the complexities of our improved differential attacks. The overall complexity of our attack is the sum of the complexities of the extended key detection and the ensuing differential attack. Except for 12-round RC5 the complexity of the key detection algorithm is much less than for the differential attack. For 12-round RC5 the complexities of both algorithms are about 2^{53}, yielding the overall complexity of 2^{54}. Kaliski-Yin estimated that for 12-round RC5 $8w$ right pairs are needed for a successful differential attack due to random noise [2]. However, since the differential in our attack has a much higher probability we estimate that $2w$ pairs suffice for this attack also.

r	Kaliski-Yin	Our differential attacks	
		All keys	1 in 32 keys
4	2^{22}	2^{17}	2^{15}
5	2^{26}	2^{22}	2^{16}
6	2^{32}	2^{24}	
7	2^{37}	2^{32}	2^{30}
8	2^{40}	2^{38}	2^{32}
9	2^{46}	2^{39}	
10	2^{51}	2^{46} (*)	2^{45}
11	2^{55}	2^{52} (*)	2^{46}
12	2^{63}	2^{54} (*)	

Table 6. Number of chosen plaintexts for the differential attacks on r-round RC5 with 32-bit words. (*) Assuming a successful key detection algorithm.

3.4 RC5 with 64 bit words

Rivest also suggested to use 16 rounds for RC5-64, a 64 bit version of RC5 [5], i.e. a 128 bit block cipher with keys of variable length. Table 7 lists the estimates of our improved differential attack on RC5-64. Although an attack requiring 2^{83} chosen plaintexts is highly unrealistic, our results show that from a theoretical point of view 16 rounds are not sufficient for RC5-64. If resistance against differential attacks is required, a 24 round version of RC5-64 appears to be preferable.

r	Plaintexts
12	2^{58}
14	2^{74}
16	2^{83}
18	2^{91}
20	2^{106}
22	2^{115}
24	2^{123}

Table 7. Number of chosen plaintexts for the differential attacks on r-round RC5 with 64-bit words, assuming a successful key detection algorithm.

4 Differentially Weak Keys

In the following we will show that despite the high complexity of the key schedule in RC5 there exist keys that are weaker than others, in the sense that a differential attack is more efficient than in the average case. We have already seen examples of this in the previous section, but we go on to show that there are more such weak keys.

The subkeys of r-round RC5 are S_i, for $i = 0, .., 2r+1$. We consider triples of subkeys with certain values in the $lg(w)$ least significant bits. Assume that $\{S_i, S_{i+1}, S_{i+2}\} =_w \{z_1, z_2, w - z_1\}$ and that $R_{i-2} =_w R_{i-1} =_w 0$. Then

$$\begin{aligned}
R_i &=_w ((R_{i-2} \oplus R_{i-1}) << R_{i-1}) + S_i =_w z_1, \quad \text{always} \\
R_{i+1} &=_w ((R_{i-1} \oplus R_i) << R_i) + S_{i+1} \\
&=_w ((0 \oplus z_1) << z_1) + z_2 =_w 0, \quad \text{with prob. } p_z \\
R_{i+2} &=_w ((z_1 \oplus 0) << 0) - z_1 =_w 0, \quad \text{always}
\end{aligned}$$

In the case where $z_1 = z_2 = 0$, $p_z = 1$. For $lg(w) \leq z_1 \leq (w - lg(w))$ we can assume that $p_z = 1/w$, if R_{i-2} and R_{i-1} are uniformly random. In that case the amount added to z_2 in the $i{+}1$st half-round will be a random value. If $z_1 < lg(w)$ or $z_1 > (w - lg(w))$ the values of z_1 and z_2 are dependent. E.g. if $z_1 = 1$ then $((z_1 << z_1) \bmod w) \in \{2, 3\}$. Thus, for the above to hold, $(z_2 \bmod w) \in \{(w-2), (w-3)\}$. These triples of keys will be called *differentially weak* keys. Consider the three half-round differential $\bar{\Omega}$. If the keys and plaintexts for the three half-rounds are as above, $\bar{\Omega}$ has probability $1/w$ as opposed to $\frac{w-lg(w)}{(2w)^2}$ in the general case. This is an improvement of a factor of about 4.7 for $w = 32$. Note that the texts to be rotated in half-round $i+1$ have difference zero. And furthermore, since $R_{i+1} =_w R_{i+2} =_w 0$ the above phenomenon can be iterated if also the next triple of keys are differentially weak, i.e. $\{S_{i+3}, S_{i+4}, S_{i+5}\} =_w \{y_1, y_2, -y_1\}$ for values of y_1, y_2 satisfying similar conditions as z_1, z_2 above, and so on for every weak triple of keys.

In the sequel we consider the version RC5-32/12/16, that is, $w = 32$, $r = 12$, with a 128 bit key. A similar analysis can be made for all other parameters of RC5. For this version a simple count of all triples of keys for which the above holds reveals 795 such keys. If the subkeys are uniformly random, such a triple of keys occurs with probability $795/2^{15} \simeq 2^{-5.37}$. The subkeys in RC5 are not random, so we implemented tests to validate this estimate. For random keys we tested whether the triples $\{\{S_3, .., S_5\}, \{S_6, .., S_8\}, \{S_9, .., S_{11}\}\}$ were all differentially weak. For ease of implementation we tested only for triples where $5 \leq z_1 \leq 27$ for z_1 as above. We evaluated the key schedule for RC5-32/12/16 for 10 million random keys. If the subkeys were really random one would expect the three triples to be weak for 113 of these keys. Our implementation found 116 keys to be weak thus confirming the estimate.

Consider keys for which the set $\{\{S_3, S_4, S_5\}, ..., \{S_{3+3k}, S_{4+3k}, S_{5+3k}\}\}$ for $k = 0, 1, ...$ are differentially weak triples of subkeys. We will use the key detection algorithm with the differential $(e_{w-1}, 0)$ to detect the values of the plaintexts yielding no rotations in the second and third half-rounds. We cannot split our algorithm into two parts as in Sect. 3.1, since that requires the use of two different differentials, and as can be seen, a triple of differentially weak keys is weak relatively to one specific differential. Thus in the considered case, where $2r + 1 = 3m + 1$, we have to look for the correct values of the two plaintext halves simultaneously.

We cannot test this version of RC5, since it requires the computation of too many ciphertext pairs. However, we can simulate the basic key detection algorithm. We choose a key detection algorithm with small Hamming weights, $\Omega^{2,3}$. This may not be the optimal choice, but it enables us to estimate the number of wrong pairs, which will be small with the chosen Hamming weights. We count the pairs for which the weights in the left and right halves of the ciphertext pairs are 2 and 3, respectively, and for which the 5 least significants of the difference in the left halves is zero. Otherwise there will be different rotations in the pair of the last half-round and the weights in the right halves will be random. If we assume that for the pairs not satisfying (1) and (2) the resultant difference in the ciphertexts will look random, wrong pairs will be accepted as a right pair with probability $\frac{\binom{27}{2}\times\binom{32}{3}}{2^{64}} \simeq 2^{-43}$.

As an example, consider the set of weak triples of keys where $k = 1$, i.e. there are 2 consecutive triples of weak keys. For the pairs satisfying (1) and (2) we need $w^2 = 2^{10}$ pairs to get a right pair after the 8th half-round. We implemented tests to estimate how many right pairs are needed after the 8th half-round to get a right pair after the $2r + 1 = 25$th half-round. We chose random keys and set the 5 least significant bits of $S_3, ..., S_8$ to zeros. By using plaintexts yielding zero rotations in the second and third half-rounds we could simulate a right pair after the 8th half-round using only one pair of plaintexts. Using 2^{29} pairs in 20 tests we obtained at least one right pair after the 25th half-round in 70% of the cases. In practice one would need to do the tests for all 2^{10} possible values of the 5 least significant bits of the plaintext halves. For each of these values we will need $2^{10} \times 2^{29} = 2^{39}$ pairs to get a right pair, totally 2^{49} pairs. For a pair of values of the plaintexts not satisfying (1) and (2) we will get about $\frac{2^{39}}{2^{43}} \simeq 2^{-4}$ wrong pairs with the right Hamming weights. By repeating this test about 8 times with a high probability unique values are suggested in the key detection algorithm using a total of 2^{53} plaintexts. Subsequently the differential attack with increased efficiency is performed. In Table 8 we list the complexities of the key detection algorithm and of the differential attacks for various groups of weak keys with up to six triples of differentially weak keys. For the keys with one triple of weak keys, the complexity of this attack will be higher than for the attack outlined in the previous section, so we did not implement that test. The estimated plaintexts needed to get at least one right pair for the plaintexts satisfying (1) and (2) in the key detection was 2^{50}, 2^{49}, 2^{47}, 2^{46}, and 2^{44} plaintexts for the $2^{-10.7}$, $2^{-16.0}$, $2^{-21.5}$, $2^{-26.8}$, and $2^{-32.2}$ fractions of the keys, respectively. By repeating the key detection algorithm a small number of times we expect all wrong pairs to be eliminated. Finally we note that estimated complexity of the key detection for the $2^{-10.7}$ fractions of the keys is the same as for the estimated complexity of the key detection for all keys from Sect. 3.2. This stems from the fact that the key detection here cannot be split into two parts. However, for these keys the ensuing differential attack has a lower complexity than in the general case. We note that similar weak keys will occur in all versions of RC5. For RC5 with 15 rounds, the complexity of a differential attack, assuming a successful key detection algorithm, is estimated to 2^{60} plaintexts for one in every $2^{32.2}$ keys. For RC5 with 18 rounds the numbers are 2^{65} plaintexts for one in every 2^{53} keys.

Fraction of keys	Key detection	Differential attack
all	2^{53}	2^{53}
$2^{-5.4}$	unknown	2^{51}
$2^{-10.7}$	2^{53} (*)	2^{49}
$2^{-16.0}$	2^{51} (*)	2^{46}
$2^{-21.5}$	2^{49} (*)	2^{44}
$2^{-26.8}$	2^{48} (*)	2^{42}
$2^{-32.2}$	2^{45} (*)	2^{40}
...	...	...

Table 8. Number of plaintexts for the key detection and the differential attack on RC5 with 12 rounds depending on the key. (*) Estimated by experiments.

5 Concluding remarks

We have shown that the known differential attacks on RC5 are not optimal. By exploiting the data-dependent rotations in RC5 in the first few rounds, we were able to improve the known attacks by a factor up to 512. Also, we showed that there are many weak keys for RC5, for which the differential attacks can be further improved. The first part of our improved attack finds the values of the plaintexts for which the differentials have a higher probability than for other values of the plaintexts. Due to a comparatively small avalanche effect per half-round in RC5, we were able to detect these plaintexts by measuring the Hamming weights in ciphertext differences. A similar approach may be applicable also in other iterated ciphers, provided there is only a small avalanche effect of bit differences in each round.

References

1. E. Biham and A. Shamir. *Differential Cryptanalysis of the Data Encryption Standard.* Springer Verlag, 1993.
2. B. Kaliski and Y.L. Yin. On differential and linear cryptanalysis of the RC5 encryption algorithm. In D. Coppersmith, editor, *Advances in Cryptology - CRYPTO'95, LNCS 963*, pages 171–184. Springer Verlag, 1995.
3. X. Lai, J.L. Massey, and S. Murphy. Markov ciphers and differential cryptanalysis. In D.W. Davies, editor, *Advances in Cryptology - Proc. Eurocrypt'91, LNCS 547*, pages 17–38. Springer Verlag, 1992.
4. R.A.Rueppel. *Analysis and Design of Stream Ciphers.* Springer Verlag, 1986.
5. R. Rivest. The RC5 encryption algorithm. In B. Preneel, editor, *Fast Software Encryption - Second International Workshop, Leuven, Belgium, LNCS 1008*, pages 86–96. Springer Verlag, 1995.