On Differential and Linear Cryptanalysis of the RC5 Encryption Algorithm

Burton S. Kaliski Jr. and Yiqun Lisa Yin

RSA Laboratories, 100 Marine Parkway, Redwood City, CA 94065
burt@rsa.com, yiqun@rsa.com

Abstract. This paper analyzes the security of the RC5 encryption algorithm against differential and linear cryptanalysis. RC5 is a new block cipher recently designed by Ron Rivest. It has a variable word size, a variable number of rounds, and a variable-length secret key. In RC5, the secret key is used to fill an expanded key table which is then used in encryption. Both our differential and linear attacks on RC5 recover every bit of the expanded key table without any exhaustive search. However, the plaintext requirement is strongly dependent on the number of rounds. For 64-bit block size, our differential attack on nine-round RC5 uses 2^{45} chosen plaintext pairs (about the same as DES), while 2^{62} pairs are needed for 12-round RC5. Similarly, our linear attack on five-round RC5 uses 2^{47} known plaintexts (about the same as DES), and the plaintext requirement is impractical for more than six rounds. We conjecture that the linear approximations used in our linear cryptanalysis are optimal. Thus, we conclude that Rivest's suggested use of 12 rounds is sufficient to make differential and linear cryptanalysis of RC5 impractical.

1 Introduction

The RC5 encryption algorithm [9] is a new block cipher designed by Ron Rivest in 1994. RC5 has a variable word size, a variable number of rounds, and a variable-length secret key. A particular RC5 algorithm is designated as RC5-$w/r/b$, where w is the *word size* in bits ($w = 16, 32, 64$), r is the number of rounds ($0 \leq r \leq 255$), and b is the number of bytes in the secret key ($0 \leq b \leq 255$). RC5 has a two-word ($2w$-bit) input and output block size.

RC5 encryption consists of three primitive operations: (1) addition modulo 2^w denoted by "+," (2) bit-wise exclusive-OR denoted by "$\oplus$," and (3) rotation: the rotation of x left by y is denoted by $x \lll y$ (the $\log_2(w)$ low-order bits of y are used). Before encryption, the secret key is used to fill an expanded key table S with $2r + 2$ words. Let (A, B) denote the two words in both the input and output block. The encryption algorithm is described below.

$A = A + S[0]$
$B = B + S[1]$
for $i = 1$ **to** r **do**
    $A = ((A \oplus B) \lll B) + S[2i]$
    $B = ((B \oplus A) \lll A) + S[2i + 1]$

D. Coppersmith (Ed.): Advances in Cryptology - CRYPTO '95, LNCS 963, pp. 171-184, 1995.

A distinguishing feature of RC5 is its heavy use of *data-dependent rotations*– the rotation amounts are random variables dependent on the input data, and they are not predetermined values. The security of RC5 relies on the rotation operation as well as the mixed use of different operations.

In this paper, we analyze the security of RC5 against differential cryptanalysis [1] and linear cryptanalysis [7]. We first briefly review both types of cryptanalysis. For differential cryptanalysis, the basic idea is that two chosen plaintexts P and P^* with a certain difference $P' = P \oplus P^*$ can encipher to two ciphertexts C and C^* such that $C' = C \oplus C^*$ has a specific value with non-negligible probability, and such a "characteristic" (P', C') is useful in deriving certain bits of the key. For linear cryptanalysis, the basic idea is to find linear approximations (parity relations among certain bits of plaintext, ciphertext, and key) which hold with probability $\neq 1/2$ (i.e., bias $= |p - 1/2| \neq 0$); such approximations can be used to obtain information about the key.

To attack RC5, one can try to find either the original secret key or the expanded key table S; both our differential attack and linear attack use the latter approach, and hence they are independent of the length of the secret key.

Our differential attack is quite effective on RC5 when the number of rounds r is relatively small, and it recovers every bit of the expanded key table S. The number of chosen plaintext pairs needed for RC5-32 is 2^{45} for $r = 9$ (about the same as 16-round DES), and the number of plaintext pairs is 2^{62} for $r = 12$. Hence, the number of plaintext requirement of our attack becomes impractical when the number of rounds is large, which is also the case for other block ciphers. We implemented our attack on for certain choices of w and r, and the actual number of plaintexts used matched our theoretical calculation.

A notable feature of our differential attack is that the type of the characteristics used is quite different from the characteristics used in attacks on other block ciphers, e.g. DES. In particular, for a given plaintext difference P' and ciphertext difference C', there are many possible paths (intermediate differences) from P' to C', each with the same probability. (In contrast, while there may be many paths for other block ciphers [4], generally one path dominates the rest.) These paths are of a similar form, allowing us to treat them uniformly. The feature has a boosting effect on the probability of getting a plaintext pair with specified P' and C'.

In our linear cryptanalysis of RC5, we construct linear approximations for RC5 and use them to compute every bit of the expanded key table S. Our linear attack on five-round RC5-32 uses 2^{47} known plaintexts (about the same as 16-round DES), and the plaintext requirement becomes impractical when $r > 6$. We also try to establish an upper bound on the bias of the best linear approximation. We prove that for each half-round of RC5, the best linear approximation that can be alternated with a trivial linear approximation holds with bias $1/2w$, and we conjecture that by alternating the two approximations, we indeed obtain the best approximation for RC5 for the proposed word sizes.

Both our differential and linear attacks on RC5 are very effective in determining key bits. After enough plaintext/ciphertext pairs are generated, the entire S

table is computed bit by bit without *any* exhaustive search. For examples, in the implementation of our differential attack, all the running time was used for generating plaintext/ciphertext pairs, and the time for key search (i.e., computing S) was negligible—less than a second on a Sun4.

We conclude that the nominal choice $r = 12$ for RC5-32 proposed by Rivest [9] provides good security against differential and linear cryptanalysis. Of course, the possibility remains that there are other ways to attack RC5, and further study is needed to fully determine the security of RC5 for any particular parameter values.

The remainder of the paper is organized into sections as follows. In §2, we introduce some useful notation. In §3, we present our differential attack on RC5, and in §4, we describe our linear attack on RC5. In §5, we summarize our recent progress in analyzing the security of RC5 with focus on the use of data-dependent rotations. In §6, we conclude with some future research directions.

2 Notation

In Rivest's description of RC5 [9], a round consists of two equations, and in each equation, one of A or B is modified while the other remains unchanged. We will refer to each equation as a *half-round*. So one half-round of RC5 is similar to a full-round in a Feistel cipher. For ease of discussion, we adopt the common notation for Feistel ciphers and rewrite RC5 as follows.

$L_1 = L_0 + S_0$
$R_1 = R_0 + S_1$
for $i = 2$ **to** $2r + 1$ **do**
$\quad L_i = R_{i-1}$
$\quad R_i = ((L_{i-1} \oplus R_{i-1}) \lll R_{i-1}) + S_i$

We will use the above description of RC5 throughout the paper. We will refer to the two equations which involve (L_{i-1}, R_{i-1}) and (L_i, R_i) as the i^{th} half-round of RC5. Hence, the two initial equations ($L_1 = L_0 + S_0$ and $R_1 = R_0 + S_1$) together are considered as the *first* half-round, and RC5 contains $2r+1$ half-rounds in total. The input block is (L_0, R_0) and the output block is (L_{2r+1}, R_{2r+1}). For ease of use, we have changed $S[i]$ to S_i.

For a binary vector x of length w, we label the bit positions from the most significant bit to the least significant bit as $w-1, \ldots, 1, 0$. We use $x[s]$ to denote the s^{th} bit of x and $x[s..t]$ $(s \geq t)$ to denote the s^{th} through t^{th} bits of x. We use $\lg(w)$ to denote $\log_2(w)$. Note that $x \bmod w = x[\lg(w) - 1..0]$ are the bits of x which are used to determine a rotation count. We use $x[s, t, \ldots, u]$ to denote $x[s] \oplus x[t] \oplus \cdots \oplus x[u]$.

3 Differential cryptanalysis of RC5

3.1 Structure of the differential attack

In this subsection, we first describe a general idea for attacking RC5 by analyzing the structure of the cipher. We then introduce the form of the chosen

plaintext/ciphertext pairs that are used in our differential attack and outline the high-level structure of the attack.

We first observe that RC5 has an iterative structure. Such a structure allows us to reduce the problem of computing the entire S table to the problem of computing the last entry S_{2r+1}. We now consider the last half-round of RC5 which involves L_{2r}, R_{2r}, R_{2r+1}, and S_{2r+1}. Suppose an algorithm $\mathcal{B}$ can compute $L_{2r}[b]$ for some $b \in \{0, \ldots, w-1\}$ given plaintext/ciphertext pairs. Since R_{2r+1} and $R_{2r}(= L_{2r+1})$ are known, bit $L_{2r}[b]$ gives information about some bit of S_{2r+1} depending on the rotation amount $R_{2r} \bmod w$. (For instance, given $L_{2r}[0]$, we can compute $S_{2r+1}[0]$ if $R_{2r} \bmod w = 0$.) For $s = 0, \ldots, w-1$, the following pseudocode computes $S_{2r+1}[s]$ using $\mathcal{B}$ when $S_{2r+1}[s-1..0]$ has been obtained.

Select a plaintext/ciphertext pair $(L_0, R_0)/(L_{2r+1}, R_{2r+1})$
  such that $(b + L_{2r+1}) \bmod w = s$
Use algorithm $\mathcal{B}$ **to calculate** $L_{2r}[b]$
If $s \geq 1$
  $x[s-1..0] = R_{2r+1}[s-1..0] - S_{2r+1}[s-1..0]$
$x[s] = L_{2r}[b] \oplus L_{2r+1}[b]$
$S_{2r+1}[s..0] = R_{2r+1}[s..0] - x[s..0]$

Therefore, we reduce the problem of computing S table to computing some bit of L_{2r}. We remark that the basic idea described so far will be used in both our differential and linear cryptanalysis.

We will use the following type of the plaintext/ciphertext pairs for computing S_i in our differential attack. Let e_s denote the w-bit binary vector which is 1 in bit s and 0 everywhere else. A *good pair for* S_i consists of two plaintexts $P = (L_0, R_0), P^* = (L_0^*, R_0^*)$ and their ciphertexts $C = (L_i, R_i), C^* = (L_i^*, R_i^*)$ satisfying the following conditions:

$$P' = P \oplus P^* = \begin{cases} (0, e_{w-1}) & \text{if } i \bmod 3 = 0, \\ (e_{w-1}, 0) & \text{if } i \bmod 3 = 1, \\ (e_{w-1}, e_{w-1}) & \text{if } i \bmod 3 = 2, \end{cases} \tag{1}$$

$$C' = C \oplus C^* = (e_t, e_u \oplus e_v), \quad t \geq \lg(w),\ u > v. \tag{2}$$

Note that when computing S_i, we can verify if the conditions hold since $S_{i+1}, \ldots, S_{2r+1}$ are already known. We will show later that with high probability, a good pair for S_i allows us to recover $L_{i-1} \bmod w$, which yields $L_{i-1}[b]$ for $0 \leq b \leq \lg(w) - 1$. Based on the discussions above, a good pair for S_i is *useful for predicting* $S_i[s]$ if

$$\exists b,\ 0 \leq b \leq \lg(w) - 1,\ \ s.t.\ (b + L_i) \bmod w = s.$$

The structure of our differential attack is given in the following pseudocode. At a high level, we compute the expanded key table S entry by entry[1] in reverse order and compute each entry S_i from the least significant bit to the most significant bit.

[1] The first three entries S_0, S_1, and S_2 cannot be computed in the uniform way as described in the pseudocode. Nevertheless, they can be easily computed by a simple algorithm when $S_3, \ldots, S_{2r+1}$ are known.

```
For i = 2r + 1 down to 3
  Obtain a set G_i of good pairs for S_i
  For s = 0 to w − 1
    Select a pair in G_i that is useful for predicting S_i[s]
    Compute S_i[s]
```

3.2 Characteristics of RC5

In this subsection, we first describe some characteristics for a half-round of RC5. Then we show how to join them together to form characteristics that are useful to compute the expanded key table S.

A characteristic for a half-round is denoted by $\Omega = (\Omega_P, \Omega_T)$, where $\Omega_P = (L'_{i-1}, R'_{i-1})$ and $\Omega_T = (L'_i, R'_i)$. Generally speaking, if a pair of inputs to a half-round have different rotation amounts, then the pair of outputs from the half-round will differ in many bits. Consequently, we will focus on characteristics for which the pair of inputs have the same rotation amounts. In most of our characteristics, each half of Ω_P and Ω_T is either zero or e_s where $s \geq \lg(w)$, implying that the rotation amounts are the same.

We will calculate the probability associated with a half-round characteristic by averaging over both the pair of inputs and round key S_i for simplicity; there may be keys for which the probability is higher and others for which it is lower. However, assuming the key expansion of RC5 is good, each round key will be essentially independent, and hence the overall probability of a characteristic for $(2r+1)$ half-rounds will be closed to what we would expect for nearly all keys. Our implementation results (see §3.4) also confirm this.

The following table lists five half-round characteristics that will be used in our attack. When analyzing the probabilities, we use the fact that for random inputs x and y such that $x \oplus y = e_s$ and random key S_i, the probability that $(x + S_i) \oplus (y + S_i) = e_s$ is at least $1/2$.

Ω	Ω_P	Ω_T	conditions	probability
Ω^1	$(0, e_s)$	(e_s, e_s)	$s \geq \lg(w)$	$p \geq (1/w) \cdot (1/2)$
Ω^2	(e_s, e_s)	$(e_s, 0)$	$s \geq \lg(w)$	$p = 1$
Ω^3	$(e_s, 0)$	$(0, e_t)$	$s, t \geq \lg(w)$	$p \geq (1/w) \cdot (1/2)$
Ω^4	$(0, e_s)$	(e_s, e_t)	$s, t \geq \lg(w), t \neq s$	$p \geq (1/w) \cdot (1/2)$
Ω^5	(e_s, e_t)	$(e_t, e_u \oplus e_v)$	$s, t \geq \lg(w), t \neq s, u > v$ $t - s = \pm(u - v) \bmod w$	$p \geq (1/w) \cdot (1/2) \cdot (1/2)$

We note that for characteristics Ω^3, Ω^4, and Ω^5, there are many possible output differences Ω_T for each input difference Ω_P. In particular, there are $(w - \lg(w))$ choices of parameter t for Ω^3, $(w - \lg(w) - 1)$ choices of parameter t for Ω^4, and w choices of parameters (u, v) for Ω^5 for each choice of Ω_P.

Two characteristics can be joined together if Ω_T from the first one and Ω_P from the second have the same value. For example, Ω^3 with parameters (s_1, t_1) can be joined with Ω^1 with parameter s_2 if $t_1 = s_2$. Therefore, the possible ways

to join the above five characteristics are Ω^1-Ω^2, Ω^2-Ω^3, Ω^3-Ω^1, Ω^3-Ω^4, and Ω^4-Ω^5. In particular, $\bar{\Omega} = \Omega^1$-Ω^2-Ω^3 is a useful characteristic for three half-rounds since it can be repeatedly joined with itself.

For the first half-round, there are three characteristics that hold with probability 1:

$\Omega^{1'} : \Omega_P = \Omega_T = (0, e_{w-1})$, which may be joined with Ω^1,

$\Omega^{2'} : \Omega_P = \Omega_T = (e_{w-1}, e_{w-1})$, which may be joined with Ω^2, and

$\Omega^{3'} : \Omega_P = \Omega_T = (e_{w-1}, 0)$ which may be joined with Ω^3.

In what follows, we construct characteristics for $2r+1$ half-rounds of RC5, which we denote by Ω_{2r+1}. Characteristic Ω_{2r+1} consists of a sequence of half-round characteristics. Since there are many possible values for the parameters of some of the half-round characteristics, there are many possible paths (i.e., intermediate differences (L'_i, R'_i) for $1 \leq i \leq 2r$) from P' to C' for Ω_{2r+1}, all of which have the same probability p. Let N denote the total number of possible paths for Ω_{2r+1}. We define the probability associated with Ω_{2r+1} as $p^{\Omega_{2r+1}} = Np$.

For different values of r, the following table lists the plaintext difference P', the sequence of half-round characteristics in Ω_{2r+1}, and the probability[2] associated with $p^{\Omega_{2r+1}}$.

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

A *right pair* with respect to Ω_{2r+1} consists of two plaintexts P, P^* and their ciphertexts C, C^* such that for all $0 \leq i \leq 2r+1$, the corresponding difference (L'_i, R'_i) has the proper form specified by sequence of the half-round characteristics for Ω_{2r+1}. We remark that for $i \leq 2r$, characteristic Ω_i, its associated probability p^{Ω_i}, and a right pair with respect to Ω_i can be defined in a similar way.

Recall the definition of a good pair for S_i in §3.1. We see that a right pair with respect to Ω_i is a good pair for S_i. Hence, the expected number of chosen plaintext pairs to get a good pair for S_i is at most the expected number of chosen plaintext pairs to get a right pair with respect to Ω_i which is at most $\frac{1}{p^{\Omega_i}}$.

[2] (1) The factor $\frac{1}{4}$ in Ω^5 can be mostly eliminated by taking the carry effect into account when analyzing output differences. Hence the factor does not appear in $p^{\Omega_{2r+1}}$ in the table. (2) When $2r+1 = 3m$, the probability associated with the first occurrence of the half-round characteristic Ω^1 is $\frac{1}{w}$ instead of $\frac{1}{2w}$ since the parameter $s = w-1$.

3.3 Using a right pair to compute L_{2r} mod w

In this subsection, we show how to compute L_{2r} mod w using a right pair with respect to Ω_{2r+1}. Similarly, we can compute L_{i-1} mod w using a right pair with respect to Ω_i. Let Ω^4 and Ω^5 be the characteristics for the $(2r)^{th}$ and $(2r+1)^{th}$ half-rounds, respectively. Recall that the parameters in Ω^4 are s and t and the parameters in Ω^5 are s, t, u, and v.

We consider the $(2r)^{th}$ half-round and obtain the following formula:

$$L_{2r} \bmod w = R_{2r-1} \bmod w = (t - s) \bmod w.$$

Given the ciphertext difference (L'_{2r+1}, R'_{2r+1}), the values of t, u, and v are easily obtained. So we need only compute s in order to get L_{2r} mod w. In the $(2r+1)^{th}$ half-round, the rotation amount L_{2r+1} mod w $(= R_{2r} \bmod w)$ is equal to either $(u-t)$ mod w or $(v-t)$ mod w. Since u, v, t, and L_{2r+1} are known, it is obvious which case holds. In the first case $s = (v - L_{2r+1}) \bmod w$ and in the second case $s = (u - L_{2r+1}) \bmod w$, and the value of L_{2r} mod w follows.

3.4 Implementation of the differential attack

In this subsection, we estimate the expected number of chosen plaintext pairs required to mount our differential attack, and then we present some experimental results.

We first recall the structure of our attack described in §3.1. The expected number of plaintext pairs required for computing S_i is the product of (1) the number of good pairs required for S_i (i.e., $|G_i|$) and (2) the expected number of plaintext pairs to get a single good pair $(\leq \frac{1}{p^{\Omega_i}})$. From the form of the good pairs, we can see that the chosen plaintext pairs for computing S_i can be reused to compute S_j if $j = i \bmod 3$. Hence, the expected number of plaintext pairs required for our attack is determined by

$$\sum_{i=2r-1}^{2r+1} |G_i| \times \frac{1}{p^{\Omega_i}}. \tag{3}$$

We now consider how many good pairs are needed, and we focus our discussions on RC5-32. We can show that $|G_i| = 2w$ good pairs are enough to guarantee a high success rate for our attack when $r \leq 11$. For $r = 12$, $8w$ good pairs are needed due to random noise. (A detailed discussion is included in the appendix.) For RC5-32, the number of chosen plaintext pairs determined by Equation 3 are listed for increasing r in the following table.

r	plaintext pairs	r	plaintext pairs	r	plaintext pairs	r	plaintext pairs
1	2^7	4	2^{21}	7	2^{36}	10	2^{50}
2	2^{10}	5	2^{25}	8	2^{39}	11	2^{54}
3	2^{16}	6	2^{31}	9	2^{45}	12	2^{62}

We have implemented our full attack for $w = 32, r \leq 6$ on a Sun4 workstation. The actual number of plaintexts used matched our theoretical calculation, and the success rate was very high. Each run took about 10 minutes for five rounds and about 12 hours for six rounds. Note that for each S_i, only 64 plaintext/ciphertext pairs were actually used for computing the key, and all other pairs were discarded right after they were generated. In addition, no exhaustive search is needed in our attack. Therefore, in our implementation, the time used for computing the S table was negligible (less than a second on the Sun4) after enough good pairs were generated.

4 Linear cryptanalysis of RC5

4.1 Linear approximations for a half-round

In this subsection, we study linear approximations for a half-round of RC5. We decompose the equation $R_i = ((L_{i-1} \oplus R_{i-1}) \lll R_{i-1}) + S_i$ into the following three equations, each of which involves only a single primitive operation, and we consider possible linear approximations for each of the equations. We will say that a linear approximation is *perfect* if it holds with bias 1/2 (probability 1 or 0).

$$X = L_{i-1} \oplus R_{i-1}, \quad Y = X \lll R_{i-1}, \quad R_i = Y + S_i.$$

I. $X = L_{i-1} \oplus R_{i-1}$

The equation has numerous perfect linear approximations. In particular, all approximations involving the same bits of X, L_{i-1}, and R_{i-1} are perfect. All other approximations have bias zero.

II. $Y = X \lll R_{i-1}$

The linear approximations for this equation can be divided into two types depending on whether bits of R_{i-1} are involved. We first consider approximations in which no bits of R_{i-1} are involved. Any such approximation involving just one bit of X and Y holds with probability $1/2+1/2w$, since for one rotation amount, the bits are guaranteed to be equal and for the other $w-1$ amounts, the bits are equal with probability 1/2. In general, for $t = 0, \ldots, \lg(w)$, an approximation involving 2^t bits of X and 2^t bits of Y (in proper positions) holds with probability $1/2 + 2^t/w$.

We next consider approximations in which some bits of R_{i-1} are involved. Some of these approximations have a non-zero bias. For example,

$$Y[0] = X[0] \oplus R_{i-1}[0] \tag{4}$$

holds with probability $1/2+1/2w$, since when the rotation amount is 0, $R_{i-1}[0] = 0$ and $Y[0] = X[0]$, and when the amount is otherwise, the equation holds with probability 1/2. We remark that an approximation will have bias zero if any bit $R_{i-1}[s]$ where $s \geq \lg(w)$ is involved.

III. $R_i = Y + S_i$

The best linear approximation for this equation is

$$R_i[0] = Y[0] + S_i[0], \tag{5}$$

which holds with probability 1. All other approximations are not perfect, and their biases are dependent on the key. For example, the bias of the approximation $R_i[s] = Y[s] + S_i[s]$ ranges from 0 to 1/2 and is averaged at 1/4 for $s \geq 1$.

For the first half-round which uses only the + operation, both approximations

$$L_1[0] = L_0[0] \oplus S_0[0] \quad \text{and} \quad R_1[0] = R_0[0] \oplus S_1[0]$$

hold with probability 1. We will denote them as `C` and `D`, respectively.

IV. Joining the linear approximations

Based on the above discussion, we can construct many possible linear approximations for a half-round of RC5 by joining the approximations for the three operations. For example, by joining $X[0] = L_{i-1}[0] \oplus R_{i-1}[0]$, approximation (4), and approximation (5), we obtain the following approximation for a half-round:

$$R_i[0] = L_{i-1}[0] \oplus S_i[0].$$

This holds with probability $1/2 + 1/2w$. We will denote it as `E`.

Since $L_i = R_{i-1}$ in a half-round of RC5, there are many trivial approximations which involve the same bits of L_i and R_{i-1} and hold with probability 1. The following trivial approximation

$$L_i[0] = R_{i-1}[0]$$

can be alternated with approximation `E`, and we will denote it as `-`.

4.2 The linear cryptanalytic attack

In this subsection, we show how to use half-round linear approximations `C`, `D`, `E`, and `-` to compute the expanded key table S. Based on our discussions in §3.1, we need only show how to compute the last entry S_{2r+1}. Similarly as in our differential attack, obtaining information about bits of L_{2r} is also very useful for computing S_{2r+1} in our linear attack.

We first note that `D-E-E-...E-` is a linear approximation for $2r$ half-rounds with the following form:

$$R_0[0] \oplus L_{2r}[0] = S_1[0] \oplus S_3[0] \oplus \cdots \oplus S_{2r-1}[0].$$

Since `E` appears exactly $r-1$ times, by Matsui's "piling-up" lemma [7], the above approximation holds with probability $\frac{1}{2} + \frac{1}{2w^{r-1}}$. Let T denote the value $S_1[0] \oplus S_3[0] \oplus \cdots \oplus S_{2r-1}[0]$. Then T is fixed for a given expanded key table S, and the value of $R_0[0] \oplus L_{2r}[0]$ is always biased toward T.

In our linear attack on RC5, we will compute S_{2r+1} in three steps: (1) compute $S_{2r+1}[0]$, (2) compute T given $S_{2r+1}[0]$, and (3) for $s = 1, \ldots, w-1$, compute $S_{2r+1}[s]$ given T and $S_{2r+1}[s-1..0]$. (For $i \leq 2r$, we can compute S_i using similar techniques with the linear approximation `D-E-E-...E-` if i is even and the linear approximation `CE-E-...E-` if i is odd.)

Step 1: Compute $S_{2r+1}[0]$. We observe that for plaintext/ciphertext pairs such that $L_{2r+1} \bmod w = 1$, one of the following two approximations is perfect:

$$L_{2r}[0] = L_{2r+1}[0] \oplus R_{2r+1}[1] \qquad \text{if } S_{2r+1}[0] = 0$$
$$L_{2r}[0] = L_{2r+1}[0] \oplus R_{2r+1}[1] \oplus R_{2r+1}[0] \text{ if } S_{2r+1}[0] = 1$$

(These can be seen by observing the effect of the carry out from the least significant bit of the addition.) Moreover, the first approximation has zero bias if $S_{2r+1}[0] = 1$ and the second approximation has zero bias if $S_{2r+1}[0] = 0$. To compute $S_{2r+1}[0]$, we obtain N known plaintext/ciphertext pairs such that $L_{2r+1} \bmod w = 1$ and consider the two quantities

$$R_0[0] \oplus (L_{2r+1}[0] \oplus R_{2r+1}[1]) \quad \text{and} \quad R_0[0] \oplus (L_{2r+1}[0] \oplus R_{2r+1}[1] \oplus R_{2r+1}[0]).$$

Let U_0 be the number of plaintexts such that the first quantity is zero and U_1 be the number of plaintexts such that the second quantity is zero. If $|U_0 - N/2| \geq |U_1 - N/2|$, we predict $S_{2r+1}[0] = 0$; otherwise, we predict $S_{2r+1}[0] = 1$.

Step 2: Compute T given $S_{2r+1}[0]$. We observe that for plaintext/ciphertext pairs such that $L_{2r+1} \bmod w = 0$, the approximation

$$L_{2r}[0] = L_{2r+1}[0] \oplus R_{2r+1}[0] \oplus S_{2r+1}[0]$$

holds with probability 1 and the right-hand side is known. To compute T, we obtain N known plaintext/ciphertext pairs such that $L_{2r+1} \bmod w = 0$. Let U be the number of plaintexts such that

$$R_0[0] \oplus (L_{2r+1}[0] \oplus R_{2r+1}[0] \oplus S_{2r+1}[0])$$

is zero. If $U \geq N/2$, we predict $T = 0$; otherwise, we predict $T = 1$.

Step 3: For $s = 1, \ldots, w-1$, compute $S_{2r+1}[s]$ given T and $S_{2r+1}[s-1..0]$. For a given plaintext/ciphertext pair, let $y = R_{2r+1} - S_{2r+1}$ and let $carry(s)$ denote the carry out from $y[s-1..0] + S_{2r+1}[s-1..0]$. We observe that for plaintext/ciphertext pairs such that $L_{2r+1} \bmod w = s$, the approximation

$$L_{2r}[0] = L_{2r+1}[0] \oplus R_{2r+1}[s] \oplus S_{2r+1}[s] \oplus carry(s)$$

holds with probability 1. To compute $S_{2r+1}[s]$, we obtain N known plaintext/ciphertext pairs such that $L_{2r+1} \bmod w = s$. Let U be the number of plaintexts such that

$$(R_0[0] \oplus T) \oplus (L_{2r+1}[0] \oplus R_{2r+1}[s] \oplus carry(s))$$

is zero. If $U \geq N/2$, we predict $S_{2r+1}[s] = 0$; otherwise, we predict $S_{2r+1}[s] = 1$.

The total number of known plaintexts expected is determined by the number of plaintexts expected for computing S_{2r+1} since the plaintexts can be reused for each S_i. Note that the bias of the $2r$ half-round approximation `D-E-E-...E-` is $\frac{1}{2w^{r-1}}$. Standard techniques for the type of linear cryptanalysis in our attack require an amount of plaintexts approximately equal to the inverse square of the bias. Therefore, $N = 4w^{2(r-1)}$ known plaintexts are required for computing each bit $S_{2r+1}[s]$ for $s = 0, 1, \ldots, w-1$, which leads to $w \times 4w^{2(r-1)}$ known plaintexts for r-round RC5. For RC5-32, the number of plaintexts is 2^{37} for $r = 4$, 2^{47} for $r = 5$, and 2^{57} for $r = 6$.

4.3 A conjecture on the bias of the best linear approximations

We conjecture that for the proposed word sizes $w = 16, 32, 64$, the linear approximation for $2r$ half-rounds used in our linear attack in §4.2 is the best linear approximation for RC5. If the conjecture is correct, we would then be able to conclude that standard linear cryptanalysis is only effective for RC5 with a very small number of rounds.

We have strong evidence for the correctness of our conjecture. In particular, we show that E is a best half-round approximation that can be alternated with a trivial approximation.

Lemma 1. *Let set M contain all half-round approximations in which neither bits of R_{i-1} nor bits of L_i are involved. Then* E *is a best approximation among all approximations in M.*

Proof. Let F be an arbitrary approximation in M. Then F can be decomposed into three approximations, one for each operation. There may be many possible decompositions, and we consider the constraints on the three approximations for a given decomposition. The approximation for $Y = X \lll R_{i-1}$ cannot involve $R_{i-1}[s]$ with $s \geq \lg(w)$ since F has bias zero otherwise. Hence, the approximation for $X = L_{i-1} \oplus R_{i-1}$ cannot involve $X[s]$ with $s \geq \lg(w)$; otherwise, either F involves bits of R_{i-1} or it has bias zero. Any approximation for $Y = X \lll R_{i-1}$ involving only $X[s]$ with $s \leq \lg(w) - 1$ holds with bias at most $1/2w$ since there is only one rotation amount that can match the bit positions of X and Y. Therefore, F has bias at most $1/2w$. Since E holds with bias $1/2w$, it is a best approximation among all approximations in M. □

We remark that the $\lll$ and $+$ operations are incompatible when constructing linear approximations for a half-round of RC5. It is clear that the bias gets larger for $\lll$ if more bits are involved in an approximation, and the bias gets smaller for $+$ if more bits are involved. Hence, the mixed use of the two operations provides good security against linear cryptanalysis.

5 Work in progress

In this section, we summarize some of our research progress [3] in analyzing the security of RC5. In particular, we focus our analysis on the use of data-dependent rotations.

We have studied how the use of data-dependent rotations helps prevent differential cryptanalysis. More specifically, we have analyzed the number of possible output differences of a half-round when the pair of inputs have different rotation amounts. We have proved that, for instance, if $\Omega_P = (e_s, e_s)$ for $s < \lg(w)$, then Ω_T is uniformly distributed in a large set containing at least $2^{w/2}$ distinct values. (Recall that for $s \geq \lg(w)$, Ω_T only has one possible value $(e_s, 0)$.) The results show that data-dependent rotations spread out bit differences in a pair of inputs in a drastic way when the differences affect the rotation amounts. Clearly, the

more input bits that differ, the higher chance that one of them will affect the rotation amounts in the pair of outputs. So a good characteristic for RC5 should always keep the number of input bit differences in each half-round as small as possible. From this viewpoint, the characteristics used in our differential attack are the best possible since there is at most one bit difference in L'_i and at most one in R'_i (except in the last two half-rounds).

The notion of a "Markov cipher" was introduced by Lai, Massey, and Murphy [4] as a criterion for an iterative cipher to be resistant to differential cryptanalysis. Loosely speaking, an iterative cipher is *Markov* if there is a way of defining differences such that the probability of an output difference of the round function depends on only the input difference and is independent of the values of the inputs. We have shown that RC5 is not a Markov cipher with respect to the difference measures $\oplus$ and $-$. This fact, however, does not imply that RC5 is vulnerable to differential attacks, since the essential property that makes a Markov cipher secure against differential attacks is that every output difference will be roughly equally likely after sufficiently many rounds. For RC5, the output difference of a half-round ranges over a large set of possible values if the input differences affect the rotation amounts, and the probability that this will happen goes to one as the number of rounds increases. Hence, even though it may not be the case that *every* output difference will occur after many rounds of RC5, the large number of possible output differences would make a differential attack impossible.

We have also considered the impact of certain simple modifications to RC5 in an effort to appreciate which operations are essential for security. For instance, if all additions were change to exclusive-or, our differential attack would be more successful since the change increases the probability of some half-round characteristics by a factor of two.

6 Conclusions

In this paper, we have studied the security of RC5 using standard techniques from differential and linear cryptanalysis. We conclude that the choice $r = 12$ for RC5-32 proposed by Rivest provides good security against both types of attacks. As a next step, we will analyze RC5 based on more special techniques such as differential cryptanalysis with partial differentials and high-order differentials, linear cryptanalysis with multiple approximations [2], and differential-linear cryptanalysis [5].

The heavy use of data-dependent rotations is a distinguishing feature of RC5, which provides certain security against differential and linear cryptanalysis. We have also seen that, however, the rotation operation "helps" an attacker in the sense that information about a bit of L_{2r} can be spread by the rotation in the last half-round to give information about every bit of the key S_{2r+1}. This feature of RC5 may also be useful in other types of attacks on RC5.

One of the design goals of RC5 was an exceptional simplicity, with the objective of making analysis easier. In contrast with other block ciphers, all of our

characteristics and linear approximations for RC5 were obtained analytically without any aid of computer experiments. The simple design of RC5 will help fully determine its security in a rather rapid way.

Acknowledgement

We would like to thank Paul Kocher, Ron Rivest, and Matt Robshaw for helpful discussions.

References

1. E. Biham and A. Shamir. *Differential Cryptanalysis of the Data Encryption Standard.* Springer-Verlag, New York, 1993.
2. B. S. Kaliski Jr. and M. J. B. Robshaw. Linear cryptanalysis using multiple approximations. In Y. G. Desmedt, editor, *Advances in Cryptology — Crypto '94*, pages 26–39, Springer Verlag, New York, 1994.
3. B. S. Kaliski and Y. L. Yin. *On the Security of the RC5 Encryption Algorithm.* Technical Report, RSA Laboratories. In preparation.
4. X. Lai, J. L. Massey, and S. Murphy. Markov ciphers and differential cryptanalysis. In D. W. Davies, editor, *Advances in Cryptology — Eurocrypt '91*, pages 17–38, Springer Verlag, Berlin, 1991.
5. S. K. Lanford and M. E. Hellman. Differential-linear cryptanalysis In Y. G. Desmedt, editor, *Advances in Cryptology — Crypto '94*, pages 17–25, Springer-Verlag, New York, 1994.
6. M. Matsui. The first experimental cryptanalysis of the Data Encryption Standard. In Y. G. Desmedt, editor, *Advances in Cryptology — Crypto '94*, pages 1–11, Springer-Verlag, New York, 1994.
7. M. Matsui. Linear cryptanalysis method for DES cipher. In T. Helleseth, editor, *Advances in Cryptology — Eurocrypt '93*, pages 386–397, Springer-Verlag, Berlin, 1994.
8. National Institute of Standards and Technology (NIST). *FIPS Publication 46-2: Data Encryption Standard.* December 30, 1993.
9. R. L. Rivest. The RC5 encryption algorithm. In *Proceedings of the Workshop on Cryptographic Algorithms*, K. U. Leuven, December 1994. To appear.

Appendix

In this appendix, we consider the number of good pairs needed to guarantee a high success rate in our differential attack on RC5-32. We will focus on $|G_{2r+1}|$, and the discussions also apply to any other $|G_i|$.

There are two issues concerning $|G_{2r+1}|$. The first issue is the signal/noise ratio. For a pair of randomly chosen plaintexts, the probability that the pair of ciphertexts have the difference C' defined in Equation 2 is

$$p^{noise} = \frac{(w - \lg w) \cdot w(w-1)/2}{2^{2w}}.$$

Such a random noise is much smaller than $p^{G_{2r+1}}$ when $r \leq 11$ and can be ignored in the analysis.

However, since a good pair has a fixed plaintext difference P' satisfying Equation 1, there is a non-negligible probability that it is not a right pair due to the special difference P'. To see how this can happen, we recall the characteristics for the last five half-rounds in a right pair. The number of non-zero bits in (L'_i, R'_i) for $i = 2r-3, \ldots, 2r+1$ are the following:

$$(1,1),(1,0),(0,1),(1,1),(1,2).$$

A pair of plaintexts with difference P' may follow the correct intermediate differences until the $(2r-4)^{th}$ half-round and then have the following number of non-zero bits in the last five half-rounds:

$$(1,1),(1,2),(2,1),(1,1),(1,2).$$

This happens for a fraction of the good pairs, and yields good pairs which are not right pairs. Implementation results and preliminary analytical results show that the fraction is no more than 10% for $w = 32$ and $r \leq 11$. Therefore, if we generate enough good pairs, we can expect to get many right pairs.

The second issue is how many right pairs are needed. In order to compute each $S_{2r+1}[s]$, G_{2r+1} must contain a right pair that is useful for predicting $S_{2r+1}[s]$. In other words, G_{2r+1} must have the following property:

For all s in $\{0, \ldots, w-1\}$, there exists a right pair in G_{2r+1} and b in $\{0, \ldots, \lg(w)-1\}$, such that $b + L_{2r+1} \bmod w = s$.

We know that a right pair recovers $\lg(w)$ bits of S_{2r+1} and assuming that the S table contains random values, the rotation amount $L_{2r+1} \bmod w$ is uniformly distributed in $\{0, \ldots, w-1\}$ for a random plaintext pair with the proper P'. Hence, if we generate $2w$ good pairs, then on average, each bit $S_{2r+1}[s]$ gets $2w\lg(w)/w = 2\lg(w)$ good pairs that are useful for predicting its value. With high probability, more than half of the good pairs are right pairs, so a majority vote will yield the correct value of $S_{2r+1}[s]$. Therefore, $2w$ good pairs are enough to guarantee a high success rate when p^{noise} is relatively small.

We remark that as r gets larger, $p^{\Omega_{2r+1}}$ will eventually become smaller than p^{noise} and more good pairs will be needed in our attack. For RC5-32, $r = 12$ is the starting point at which $p^{\Omega_{2r+1}}$ becomes smaller than p^{noise}. Simple calculations show that $8w$ pairs are enough to guarantee a high success rate for $r = 12$.