Distinguishing Attacks on RC4 and A New Improvement of the Cipher

Jing Lv
Institute of Software
Chinese Academy of Sciences
Beijing, China
Email:lvjing@is.iscas.ac.cn

Bin Zhang
Institute of Information Engineering
Chinese Academy of Sciences
Beijing, China
Email: zhangbin@is.iscas.ac.cn

Dongdai Lin
Institute of Information Engineering
Chinese Academy of Sciences
Beijing, China
Email: ddlin@iie.ac.cn

***Abstract*—RC4, designed by Rivest in 1987, is the most widely deployed stream cipher in practical applications. In this paper, two new class of statistical biases inherent in RC4 are depicted and it is shown that the RC4 keystream is distinguishable from random no matter how many initial bytes have been dumped. RC4A, proposed by Paul and Preneel at FSE 2004 to strengthen the security of RC4, is also found to be vulnerable to similar attacks. Instead, a new pseudorandom bit generator RC4B is proposed, which is believed to provide better immunity against the known attacks.**

I. Introduction

RC4, designed by Ron Rivest in 1987, is the most widely deployed stream cipher in practical applications. Due to its simplicity and extremely fast software performance, RC4 has been integrated into TLS/SSL and WEP applications [10]. RC4 takes an interesting design approach which is quite different from that of LFSR-based stream ciphers. This implies that many of the analysis methods known for such ciphers cannot be applied. The internal state of RC4 consists of a table of 2^n n-bit words and two n-bit pointers, where n is a parameter (for the nominal version, $n = 8$). The table varies slowly in time under the control of itself. When $n = 8$, RC4 has a huge state of $log_2 2^8!$, approximately 1684 bits. It is thus impractical to guess even a small part of this state,or to use standard time/memory/data tradeoff attacks. In addition, the state evolves in a complex non-linear way, and thus it is difficult to combine partial information about states which are far away in time. Consequently, all the techniques developed to attack stream ciphers based on linear feedback shift registers seem to be inapplicable to RC4.

Since its introduction, RC4 has attracted much attention and withstood huge efforts of cryptanalysis, to name but a few [2], [3], [5], [7], [8], [9], [11], [12], [13]. At FSE'2001, Mantin and Shamir showed that the second output byte of RC4 is not random. Later at FSE'2004, Paul and Preneel observed that the first two output bytes are equal with probability significantly less than expected. Based on these results, it is easy to launch distinguishing attacks on RC4. To frustrate such attacks, the first 256 bytes of RC4 are suggested to be dumped. In this paper, we report two new class of statistical biases inherent in each keystream word and every two consecutive keystream words of RC4-N, where N is the length of the internal array, and construct efficient distinguishers accordingly. Further, it is shown that RC4A, proposed by Paul and Preneel at FSE 2004 is also vulnerable to similar attacks. Instead, a new pseudorandom bit generator, RC4B, is proposed, which is believed to provide better immunity against the above attacks.

This paper is organized as follows. In Section 2, we introduce the RC4 cipher and the notations we use throughout this paper. We present our theoretical analysis about the biases in the output in details in Section 3. The corresponding distinguishers are constructed in Section 4. In Section 5, we provide our experimental results to verify our analysis on RC4. Then in Section 6, we analyze the RC4A cipher similarly and show that it fails to strengthen RC4 in this aspect. Instead, we propose the RC4B cipher in Section 7 with its analysis. Finally, some conclusions are presented in Section 8.

II. Description of RC4

RC4 runs in two phases, the key scheduling phase KSA and the output keystream generation phase PRGA. The description is as follows.

```
1   KSA
2   for i ← 0 to N − 1
3       do s[i] ← i
4   j ← 0
5   for i = 0 to N − 1
6       do j ← j + s[i] + k[i mod l]
7          swap (s[i], s[j])
8   PRGA
9   while i ≥ 0
10      do i ← i + 1
11         j ← j + s[i]
12         swap (s[i], s[j])
13         output s[s[i] + s[j]]
```

The KSA swaps N pairs of the array $\{0,1,2 \ldots N-1\}$, depending on the value of the secret key, where l is the word length of the secret key. At the end of KSA, we reach an initial state for PRGA phase, which generates keystream words of $\log_2 N$ bits. Note that the symbol $+$ denotes the addition modular N.

We define the state $s_t[l]$ the lth element of the array after the swapping at round t in PRGA, z_t denotes the tth output word.

The indices i_t, j_t represent the swapping indices at round t. In this paper, we assume that the permutation is distributed uniformly, and $j_t (t > 0)$ follows the uniform distribution over $[0, N-1]$.

III. Previous Attacks on RC4

There are two approaches in the study of cryptanalysis of RC4: attacks based on the weakness of the KSA and attacks based on the weakness of PRGA. Considering PRGA, Knudsen have attacked versions of RC4 with $n < 8$ by their backtracking algorithm in which adversary guess the internal state and checks if an anomaly occurs in later stage[4]. At FSE'2001, Mantin and Shamir showed that the second output byte of RC4 is not random. Though it is a serious weakness in RC4, it can be avoided by simply dumping the first N bytes of RC4 keystream[7]. Later at FSE'2004, Paul and Preneel observed that the first two output bytes are equal with probability significantly less than expected, they also proposed similar conclusion when $t = 0(\text{mod}N)$, which has a smaller bias[11]. We notice that all the bias proposed in [7] and [11] are at some special rounds, there are no conclusions about bias exists at any round or most of the rounds. In [8], it is proposed to search for special internal states that contains a pattern consisting of two pointer values and some known permutation entries, which believed to significantly reduces the complexity of the algorithm in [4]. In [13], they present a technique to automatically reveal linear correlations in the PRGA phase, and then bind the new bias they found with known KSA weakness to provide key recovery attacks, we notice that the new bias they found are also exit at some special rounds.

IV. The biases in RC4

Several bias about some special words of RC4 are presented in [5], [6], [7], [11], [13]. Almost all of them are about the first N words. To the authors' knowledge, there is no general rules discovered for the RC4 keystream so far. It is really a tough work to distinguish RC4 keystream from random just by some special bytes in the first N words, let alone to recover the inner state. What's more, the first N words are dumped sometimes for the security of the cipher.

In this section, we present our new bias in the RC4 keystream, which gives almost no limit to the time or the index i. We believe the general rules we discovery can be well used to attack the cipher. We present our theoretical analysis about the statistical biases in this section, which will be used in the next section to construct the corresponding distinguishers. Here comes our first theorem.

Theorem 1. *If $s_{t-1}[t+1] = 0, j_{t-1} = 0$ and $s_{t-1}[t] \neq t+1$, then $z_{t+1} = 0$.*

Proof. The proof comes from the execution process of the cipher. First, at round t, we have

$$\begin{aligned} i_t =& t, \\ j_t =& j_{t-1} + s_{t-1}[i_t] = s_{t-1}[t] \neq t+1. \end{aligned}$$

The next step is swap$(s_{t-1}[t], s_{t-1}[j_t])$. At round $t+1$, we get

$$\begin{aligned} i_{t+1} =& t+1, \\ j_{t+1} =& j_t + s_t[t+1]. \end{aligned}$$

Since $i_t, j_t \neq t+1$, so $s_t[t+1] = s_{t-1}[t+1] = 0$, thus $j_{t+1} = j_t$, so next swap$(s_t[t+1], s_t[j_t])$. From above, we obtain

$$\begin{aligned} z_{t+1} &= s_{t+1}[s_{t+1}[t+1] + s_{t+1}[j_t]] \\ &= s_{t+1}[s_t[t+1] + s_t[j_t]] \\ &= s_{t+1}[s_{t-1}[t+1] + s_{t-1}[t]] \\ &= s_{t+1}[j_t] = s_t[t+1] = s_{t-1}[t+1] = 0. \end{aligned}$$

This completes the proof. □

Corollary 1 immediately follows by noting the fact that $j_0 = 0$.

Corollary 1. *If $s_0[2] = 0$ and $s_0[1] \neq 2$, then $z_2 = 0$.*

This is a conclusion in [8], we can see it is just a special case of our Theorem 1.

From Theorem 1, we can compute the bias by using the total probability formula.

Corollary 2. *The probability of the output word is zero can be approximated by the equation below:*

$$Pr(z_{t+1} = 0) = \begin{cases} \frac{1}{N}(1 + (1 - \frac{1}{N})^2) & \text{if } t = 1 \\ \frac{1}{N}(1 + \frac{1}{N}(1 - \frac{1}{N})^2) & \text{if } t > 1 \end{cases}$$

Proof. Let B_t denote the event $s_{t-1}[t+1] = 0, j_{t-1} = 0$, and $s_{t-1}[t] \neq t+1$. Then

$$Pr(B_t) = \begin{cases} \frac{1}{N}(1 - \frac{1}{N}) & \text{if } t = 1 \\ \frac{1}{N^2}(1 - \frac{1}{N}) & \text{if } t > 1 \end{cases} \quad (1)$$

We prove the result by decomposing the event $z_{t+1} = 0$ into two cases.

$$\begin{aligned} & Pr(z_{t+1} = 0) \\ =& Pr(z_{t+1} = 0|B_t)Pr(B_t) + Pr(z_{t+1} = 0|\bar{B}_t)Pr(\bar{B}_t) \\ =& Pr(B_t) + 1/N(1 - Pr(\bar{B}_t)) \end{aligned}$$

Where $\bar{B}_t$ represents the supplement event of B_t. Substituting the value of $Pr(B_t)$ in (1), we obtain Corollary 2. □

Next, we introduce another bias in RC4 keystream.

Theorem 2. *When $t \neq -2, -1(modN)$, if $j_{t-1} = 0$, $s_{t-1}[i_t] = t+1$, then we have $z_t \neq z_{t+1}$.*[1]

Proof. At round t, we have

$$i_t = t,\ j_t = j_{t-1} + s_{t-1}[t] = 0 + t + 1 = t + 1$$

then we swap $s_{t-1}[t]$ and $s_{t-1}[t+1]$, and output

$$z_t = s_t[s_t[t+1] + s_t[t]] = s_t[s_t[t] + t + 1]$$

[1] in [1], it mentions that if $i_t = j_t$, $s_t[i_{t+1}] = 2$, then $z_t = z_{t+1}$ without prove. Unfortunately, it is not true.

At round $t+1$, we have

$$i_{t+1} = t+1,\ j_{t+1} = j_t + s_t[t+1] = t+1+t+1 = 2t+2,$$

then we swap $s_{t+1}[t+1]$ and $s_{t+1}[2t+2]$, and output

$$z_{t+1} = s_{t+1}[s_{t+1}[t+1]+s_{t+1}[2t+2]] = s_{t+1}[s_t[2t+2]+t+1].$$

So if $z_t = z_{t+1}$, there are only two cases:
1) the index of the two output is equal, and neither of them is the exchange index at round $t+1$, that is to say:

$$s_t[t]+t+1 = s_t[2t+2]+t+1,\ s_t[t]+t+1 \neq t+1, 2t+2.$$

In this case we get $s_t[t]+t+1 = s_t[2t+2]+t+1$, so

$$t = 2t+2, t = -2, s_{-2}[-2] \neq 0, -1.$$

2) the index of the two output are both the exchange indexes, that is to say:

$$s_t[t]+t+1 = t+1, s_t[2t+2]+t+1 = 2t+2$$

or

$$s_t[t]+t+1 = 2t+2, s_t[2t+2]+t+1 = t+1;$$

3) there is in fact no changes happen at the round $t+1$, that is to say:

$$i_{t+1} = j_{t+1}.$$

In the first case, we have

$$\begin{aligned} &s_t[2t+2] = t+1 = s_t[t+1] \\ &t+1 = 2t+2 \\ &t = -1 \end{aligned}$$

then $s_{-1}[0] = s_{-1}[-1] = 0$, which is impossible.

In the second case, we have

$$s_t[t] = t+1 = s_t[t+1] \Rightarrow t = t+1$$

which is also impossible.

In the third case, we have

$$t+1 = 2t+1$$

$\Rightarrow t = -1$. □

The same as Theorem 1, we consider the situation at initial time and then calculate the bias.

Corollary 3. *If $s_0[1] = 2$, then the first two output words of RC4 are always different.*

It is easy to get this conclusion by noticing the fact that $j_0 = 0$. Corollary 3 is proved in [11], it is only a special case of our Theorem 2.

Corollary 4. *When $t \neq -2, -1(mod N)$,the probability*

$$Pr(z_t = z_{t+1}) = \begin{cases} \frac{1}{N}(1-\frac{1}{N^2}) & \text{if } t > 1 \\ \frac{1}{N}(1-\frac{1}{N}) & \text{if } t = 1 \end{cases}$$

Proof. Let A_t note the event $j_{t-1} = 0$ and $s_{t-1}[t] = t+1$. Then

$$Pr(A_t) = \begin{cases} \frac{1}{N} & \text{if } t = 1 \\ \frac{1}{N^2} & \text{if } t > 1 \end{cases} \quad (1)$$

when $t > 1$, we have

$$\begin{aligned} &Pr(z_t = z_{t+1}) \\ =&Pr(z_t = z_{t+1}|A_t)Pr(A_t) + Pr(z_t = z_{t+1}|\bar{A}_t)Pr(\bar{A}_t) \\ =&0 * Pr(A_t) + 1/N(1 - Pr(A_t)) \\ =&1/N(1 - Pr(A_t)) \end{aligned}$$

Substituting the value of $Pr(A_t)$ in (1) to the equation above ,we obtain Corollary 4. □

V. The distinguishers

Theorem 1 and Theorem 2 immediately give a class of distinguishers. In this section, we construct our distinguisher sequences $\{\mathscr{A}_t\}_{t>1}$ and $\{\mathscr{B}_t\}_{t>0}$ using the biases. In order to calculate how many samples the distinguishers need, we quote the theorem in [8]. One can find the proof in [8].

Lemma 1. *If event e occurs in a distribution X with probability p and in Y with probability $p(1+q)$. Then, for small p and q, $O(1/pq^2)$ samples are required to distinguish X from Y with non-negligible probability of success.*

At first, we construct our distinguisher $\{\mathscr{A}_t\}_{t>1}$. Since the probability that $z_t = 0$ in the RC4 keystream exceeds $1/N$. So when the number of 0 is non-ignorable higher, we think it is from RC4 keystream. But there is a question, how to define non-ignorable? In [2], it points out that when the distinct of the means exceeds the standard deviation, the two distribution are distinguishable.

Distinguisher $\mathscr{A}_t$.

For the N_1 given keystreams $\{z^i\}_{i=1}^{N_1}$
Processing:
1: Compute
$c_t = 0$, $\sigma = \sqrt{N_1 \frac{1}{N}(1-\frac{1}{N})}$, $\mu = N_1/N$.
2: For every keystream, if $z_t^i = 0$, then $c_t = c_t + 1$.
3: If $c_t > \mu + \sigma$, then outputs 1, else outputs 0.

By applying Lemma 1 to the distinguisher, we get the data complexity for $\mathscr{A}_t$.

Theorem 3. *$\mathscr{A}_t$ needs $N_1 = O(N^3)$ keystreams to distinguish RC4 from Random when $t > 2$, and $N_1 = O(N)$ keystreams when $t = 2$.*

Proof. Since $Pr(z_t = 0) = \frac{1}{N}(1 + \frac{1}{N}(1-\frac{1}{N})^2)$ when $t > 2$, by Lemma 1 the samples needed is

$$N_1 = \frac{1}{\frac{1}{N}(\frac{1}{N})^2(1-\frac{1}{N})^2} \doteq O(N^3).$$

Similar, when $t = 2$,

$$N_1 = O(\frac{1}{\frac{1}{N}(1-\frac{1}{N})^2}) \doteq O(N).$$

This completes the proof. □

By corollary 4, we get our second distinguisher sequence $\{\mathscr{B}_t\}_{t>0}$.

The distinguisher $\mathscr{B}_t$

For the N_1 given keystreams.

Processing:

1: Compute

$c_t = 0, \sigma = \sqrt{N_1 \frac{1}{N}(1-\frac{1}{N})}$, $\mu = N_1/N$.

2: For every keystream

If $z_t^i = z_{t+1}^i$, then $c_t = c_t + 1$.

3: If $c_t < \mu - \sigma$, then outputs 1, else outputs 0.

Using the same method we get the data complexity of $\mathscr{B}_t$.

Theorem 4. *$\mathscr{B}_t$ needs $N_1 = O(N^5)$ keystreams to distinguish RC4 from Random when $t > 1$, and $N_1 = O(N^3)$ keystreams when $t = 1$.*

Proof. Since $Pr(z_t = z_{t+1}) = \frac{1}{N}(1 - \frac{1}{N^2})$ when $t > 2$, the samples needed is

$$N_1 = O(N(N^2)^2) = O(N^5)$$

by Lemma 1. Similar, when $t = 1$,

$$N_1 = O(\frac{1}{\frac{1}{N}(\frac{1}{N})^2}) = O(N^3)$$

□

VI. THE EXPERIMENT RESULT

In [2], it indicates that when $|\mu - \mu_0| > \sigma_0$, the two streams are distinguishable, where μ, μ_0 represents the mean value and σ_0 the standard deviation. When $t \geq 2$, let X_t denotes the event $z_t = 0$ when the keystream is true. Y_t denotes the event $z_t = 0$ when the keystream is random. In other words,

$$X(Y)_t = \begin{cases} 1 & z_t = 0 \\ 0 & \text{otherwise} \end{cases}$$

We run N_1 keystreams for $\mathscr{A}_t$, by Theorem3, let

$$N_1 = \begin{cases} N & \text{if } t = 2 \\ N^3 & \text{if } t \geq 3 \end{cases}$$

So we get in the random case:

$$\mu_0 = EY_t = \begin{cases} N \cdot 1/N = 1 & t = 2 \\ N^3 \cdot 1/N = N^2 & t \geq 3 \end{cases}$$

$$\sigma_0 = \sqrt{DY_t} = \begin{cases} \sqrt{N \cdot 1/N(1-1/N)} & t = 2 \\ \sqrt{N^3 \cdot 1/N(1-1/N)} & t \geq 3 \end{cases}$$

and in the true keystream case:

$$\mu_1 = EX_t = N_1 \cdot p_1 = N_1 \cdot n_t/N_1 = n_t$$

where n_t is the number of $X_t = 1$ in the N_1 true keystreams. Since $Pr(z_t = 0) > 1/N$, when

$$n_t > \mu_0 + \sigma_0$$

the two keystreams are distinguishable. We do similar things for $\mathscr{B}_t$. Let

$$X(Y)'_t = \begin{cases} 1 & z_t = z_{t+1} \\ 0 & \text{otherwise} \end{cases}$$

We run N'_1 keystreams for $\mathscr{B}_t$,

$$N'_1 = \begin{cases} N^3 & \text{if } t = 1 \\ N^5 & \text{if } t \geq 2 \end{cases}$$

So,

$$\mu_0 = EY'_t = \begin{cases} N^3 \cdot 1/N = N^2 & t = 2 \\ N^5 \cdot 1/N = N^4 & t \geq 3 \end{cases}$$

$$\sigma_0 = \sqrt{DY'_t} = \begin{cases} \sqrt{N^3 \cdot 1/N(1-1/N)} & t = 1 \\ \sqrt{N^5 \cdot 1/N(1-1/N)} & t \geq 2 \end{cases}$$

and for the true keystream

$$\mu'_1 = EX'_t = N'_1 \cdot p'_1 = N'_1 \cdot n'_t/N'_1 = n'_t$$

where n'_t is the number of $X'_t = 1$ in the N'_1 true keystreams. Since when the keystream is true, $Pr(z_t = z_{t+1}) < 1/N$. When

$$n'_t < \mu'_0 + \sigma'_0$$

the two keystreams are distinguishable.

For $N = 256$, we run N keystreams for $\mathscr{A}_2$ 2^{12} times, there are 2422 $\mathscr{A}_2$ outputs 1, so the success probability is 2422/4096=59.1%. Next we run N^3 keystreams for $\mathscr{A}_3$ to $\mathscr{A}_N$, there are 82 $\mathscr{A}_t$ outputs 1, so the success probability is $82/(N-2) = 32.3\%$.

For $N = 64$, we run N^3 keystreams for $\mathscr{B}_1$ 2^{12} times, there are 2485 $\mathscr{B}_1$ outputs 1, so the success probability is 60.7% . Next we run N^5 keystreams for $\mathscr{B}_2$ to $\mathscr{B}_{N-1}$, there are 37 $\mathscr{B}_t$ outputs 1, so the success probability is 37/($N-1$)=62.0% . For N=128, 80 of $\mathscr{B}_2$ to $\mathscr{B}_{N-1}$ outputs 1, so the success probability is 63.0%.

VII. DETECTION OF STATES IN THE KEYSTREAM

Definition 1. *An a-state is a partially specified RC4 state, that includes i, j and a states elements of the RC4 state array s.*

The internal a-state can be regarded as an internal event with probability

$$Pr[E_{int}] = N^{-a-1}.$$

When the internal event occurs, there is an external event E_{ext} observed in the keystream, which is associated with the interval event, i.e $Pr[E_{ext}|E_{int}] = 1$. In[8], there is a search algorithm consisting in the sequential search through the values of internal state components that are consisting with a given keystream segment, with backtracking in case of found contradictions. It means one should calculate the

probability $Pr[E_{int}|E_{ext}]$. Applying Bayes' law we can derive the probability

$$\begin{aligned}&Pr[E_{int}|E_{ext}]\\=&\frac{Pr[E_{ext}|E_{int}]Pr[E_{int}]}{Pr[E_{ext}]}\\=&\frac{Pr[E_{int}]}{Pr[E_{ext}]}.\end{aligned}$$

Theorem 5. *If $z_{t+1} = 0$, then $Pr(s_{t-1}[t+1] = 0, j_{t-1} = 0) \geqslant \frac{1}{N} - \frac{1}{N^2} \approx \frac{1}{N}$.*

Proof. By Theorem 1, we have

$$Pr[z_{t+1} = 0|s_{t-1}[t+1] = 0, j_{t-1} = 0, s_{t-1}[t] \neq t+1] = 1,$$

so

$$\begin{aligned}&Pr[s_{t-1}[t+1] = 0, j_{t-1} = 0|z_{t+1} = 0]\\\geqslant&Pr[s_{t-1}[t+1] = 0, j_{t-1} = 0, s_{t-1}[t] \neq t+1|z_{t+1} = 0]\\=&\frac{Pr[s_{t-1}[t+1] = 0, j_{t-1} = 0, s_{t-1}[t] \neq t+1]}{Pr[z_{t+1} = 0]}\\=&\frac{\frac{1}{N^2}(1-\frac{1}{N})}{\frac{1}{N}}\\=&\frac{1}{N}(1-\frac{1}{N})\end{aligned}$$

□

Theorem 6. *When $t \neq -1, -2(modN)$, if $z_t \neq z_{t+1}$, then $Pr[s_{t-1}[i_t] = t+1, j_{t-1} = 0] = \frac{1}{N}$.*

Proof. When $t \neq -1, -2(modN)$, by Theorem 2, we have

$$Pr[z_t \neg z_{t+1}|s_{t-1}[i_t] = t+1, j_{t-1} = 0] = 1,$$

so

$$\begin{aligned}&Pr[s_{t-1}[i_t] = t+1, j_{t-1} = 0|z_t \neq z_{t+1}]\\=&\frac{Pr[s_{t-1}[i_t] = t+1, j_{t-1} = 0]}{Pr[z_t \neq z_{t+1}]}\\=&\frac{1/N^2}{1/N} = \frac{1}{N}\end{aligned}$$

□

In [8], the algorithm search special inner states from the keystream segment. As we known, when $t = 0$, i, j is initial with zero, so there are totally $N!$ of initial state, including i, j, and the state array $s[N]$. Since the algorithm of RC4 is invertible, in every step t, there are $(N-1)N!$ impossible states. In [1], there is an example about an impossible cycle. So when searching for a patten in [8], wether it is a possible state should be take into consideration. Our conclusion present in the above two theorem meet the experiment result well.

VIII. The RC4A

In [1], they construct a new cipher RC4A in order to avoid the weakness in Corollary 3. The RC4A contains of two arrays, s^1 and s^2, using the key k_1 and k_2 respective in the KSA. And the new PRGA is defined as follow:

```
1  i ← 0, j¹ ← 0, j² ← 0
2  while i ≥ 0
3        do i++
4           j¹ ← j¹ + s¹[i]
5           swap (s¹[i], s¹[j¹])
6           output z = s²[s¹[i] + s¹[j¹]]
7           j² ← j² + s²[i]
8           swap (s²[i], s²[j²])
9           output s¹[s²[i] + s²[j²]]
```

We can see from the algorithm what different from RC4 cipher is that the index produced by one array is output through the other array. The way of exchange the array is totally the same as RC4, in other words, the array is update just by itself, it has nothing to do with another array. This lead to the same problem as RC4.

Theorem 7. *Assuming that the arrays s^1, s^2 of RC4A are distributed uniformly , then $Pr(z_1 = z_3) = \frac{1}{N}(1 - (\frac{1}{N})^2)$.*

Proof. Let C denotes the event $s_0^1[1] = 2, s_0^2[1] = 1$. We consider the situation when C happens. Let

$$X = s_0^1[2], Z = s_0^1[4].$$

At round 1,

$$i = 1, j_1^1 = s_0^1[1] = 2,$$

so we swap $(s_0^1[1], s_0^1[2])$, and output $z_1 = s_0^2[X+2]$.

$$j^2 = s_0^2[1] = 1 = i,$$

so $s_0^2 = s_1^2$, and output z_2.
At round 2,

$$i = 2, j^1 = 2 + s_1^1[2] = 4,$$

so we swap $(s_1^1[2], s_1^1[4])$, since $s^1[4]$ hasn't been changed in the swap, $s_1^1[4] = s_0^1[4] = Z$, so we output

$$z_3 = s_1^2[2+Z] = s_0^2[2+Z].$$

Since $X \neq Z$ so $z_1 \neq z_3$. From above we compute the probability as follows

$$\begin{aligned}&Pr(z_1 = z_3)\\=&Pr(z_1 = z_3|C)Pr(C) + Pr(z_1 = z_3|\bar{C})Pr(\bar{C})\\=&0 * Pr(C) + \frac{1}{N}(1 - (\frac{1}{N})^2)\\=&\frac{1}{N}(1 - (\frac{1}{N})^2)\end{aligned}$$

This completes the proof. □

Considering the situation when $j_t^2 = t$, $s_{t-1}^1[t] = t+1$ and $j_{t-1}^1 = 0$, we can get Theorem 6 in a similar way.

Theorem 8. *Assuming that the arrays s^1, s^2 of RC4A are distributed uniformly , then*

$$Pr(z_{2t-1} = z_{2t+1}) = \frac{1}{N}(1 - (\frac{1}{N})^3), \text{ for all } t > 1.$$

We conclude from Theorem 5 and Theorem 6 that RC4A cipher is better than RC4 cipher in resisting the distinguish attack. But we can still construct a distinguish attack using O(N^5) keystreams when we know the first output word, and O(N^7) keystreams when we know other output word. The reason is that RC4A cipher doesn't alter RC4's shuffle model, it means that the state array will be changed in the same way the RC4 cipher.

IX. THE RC4B

From the analysis about RC4A, we conclude that in order to avoid the weakness we mentioned in the previous section, we have to change the way of the array's update. In this section, we introduce a new cipher RC4B, which also based on the RC4's exchange shuffle model. Like RC4A, it also consists of two arrays s^1, s^2, what's more, the KSA of RC4B is the same as RC4A. But different from RC4A, RC4B mixes the two arrays' state. The algorithm of RC4B is as follows:

$i \leftarrow 0, j^1 \leftarrow 0, j^2 \leftarrow 0$
while $i \geq 0$
  do $i++$
    $j^1 \leftarrow j^1 + s^2[i]$
    $swap\ (s^1[i], s^1[j^1])$
    $output\ z \leftarrow s^2[s^1[i] + s^1[j^1]]$
    $j^2 \leftarrow j^2 + s^1[i]$
    $swap\ (s^2[i], s^2[j^2])$
    $output\ s^1[s^2[i] + s^2[j^2]]$

RC4B generates keystream faster than RC4. To produce two successive output word. The i pointer stays the same, while the j pointer increment two times the same as RC4.

In RC4 and RC4A, the arrays are changed by themselves, that is to say, the exchange index i, j is increment by the same array. We can see from our analysis that this is an important reason for the above attack. In RC4B, which elements to swap at each step is determined by the other array. If we apply the proof of Theorem 5 to RC4B, then at round 2, we have no idea about the value of j^1, the reason is that the value of s_1^2 is unknown. We believe it can resist the attack above. RC4B mix the two arrays more sufficient than RC4A, the advantage of two arrays instead of one is markedly. The number of different states is $N! * N! * N^3$, this is approximately 2^{3388} when $N = 256$. This is a very huge state space. More analysis for RC4B will be present in the full vision.

X. CONCLUSIONS

In this paper, we have depicted two new classes of biases in the RC4 keystream and built distinguishers accordingly. Our results indicate that the RC4 keystream is far from random even if the initial keystream bytes have been dumped. This is an very important weakness in RC4. Similar weakness in RC4A is also proposed and a new pseudorandom bit generator RC4B is proposed, which we believe will be much better than RC4 and RC4A in security.

REFERENCES

[1] Hal Finney "an RC4 cycle that can't happen."*sci.crypt 1994*.

[2] E.Biham, Y.Carmeli "Efficient reconstruction of RC4 keys from Internal states" *Fast Software Encryption-FSE'2008* LNCS vol.5086, pp. 270-288, Springer-Verlag, 2008.

[3] A.Klein "Attacks on the RC4 stream cipher" *Designs, Codes and Cryptography* vol.48, issue 3, pp. 269-286, September 2008.

[4] Knudsen,L.R.,Meier"Analysis methods for (alleged) RC4." *ASIACRYPT 1998* LNCS, Vol. 1514, pp. 327-341. Springer, Heidelberg 1998.

[5] S.Maitra, G.Paul "New form of permutation bias and secret key leakage in keystream bytes of RC4" *Fast Software Encryption-FSE'2008* LNCS vol.5086, pp. 253-269, Springer-Verlag, 2008.

[6] S.Maitra, G.Paul "Attack on Broadcast RC4 Revisted" *Fast Software Encryption-FSE'2011* LNCS vol.6733, pp. 199-217, Springer-Verlag, 2011.

[7] I.Mantin, A.Shamir "A practical attack on broadcast RC4" *Fast Software Encryption-FSE'2001* LNCS vol. 2355, pp. 152-164, 2002.

[8] A. Maximov and D.Khovratovich "New state recovery attack on RC4" *Advances in Cryptology-Crypto'2008*, LNCS vol. 5157, pp.297-316, Springer-Verlag, 2008.

[9] I.Mironov "Not so random shuffle of RC4" *Advances in Cryptology-Crypto'2002* LNCS vol.2442, pp. 304-319, Springer-Verlag, 2002.

[10] R. Rivest, "RSA Security response to weaknesses in key scheduling algorithm of RC4" *Technical note available from RSA Security, Inc. site*, http://www.rsasecurity.com/rsalabs/technotes/wep.html, 2001.

[11] S.Paul, B.Preneel "A New weakness in the RC4 keystream generator and an approach to improve the security of the cipher," *Fast Software Encryption-FSE'2004* LNCS vol. 3017, pp. 245-259, Springer-Verlag, 2004.

[12] P.Sepehrdad, S.Vaudenay, and M.Vuagnoux "Statistical attack on RC4 Distinguishing WPA" *Advances in Cryptology-Eurocrypt'2011*, LNCS vol. 6632, pp. 343-363, Springer-Verlag, 2011.

[13] P.Sepehrdad, S.Vaudenay and M.Vuagnoux "Discovery and exploitation of new bias in RC4" *Selected Areas in Cryptography-SAC 2010*, LNCS vol. 6544, pp. 74-91, Springer-Verlag, 2011