Predictive State を用いた RC4 の新しい弱鍵のクラス

長尾 篤 † 大東 俊博 ‡ 五十部 孝典 † 森井 昌克 †

† 神戸大学大学院工学研究科
657-8501 兵庫県神戸市灘区六甲台町 1-1
a.nagao@stu.kobe-u.ac.jp
mmorii@kobe-u.ac.jp

‡ 広島大学情報メディア教育研究センター
739-8511 広島県東広島市鏡山 1-4-2
ohigashi@hiroshima-u.ac.jp

あらまし 2001 年，Mantin と Shamir は RC4 の PRGA において数バイトの内部状態から出力の一部を予測できる Predictive State の存在を発表した．2011 年に寺村らによって Predictive State を導くことのできる弱鍵が発見され，Predictive State を用いて鍵回復攻撃を行えることが示された．しかし寺村らは Predictive State を導く確率が最も高い鍵のみを弱鍵としており，それ以外の弱鍵のクラスについては検討していない．本稿では，寺村らの弱鍵が満たすべき関係式の一部を省略しても弱鍵としての性質を持つことに注目し，新しい弱鍵のクラスを特定する方法を提案する．新たな弱鍵は攻撃に必要な計算量は既存のものより増加するが，より広い鍵空間を弱鍵として定義することができる．特に 5 個の内部状態から成る Predictive State から得られる弱鍵では，その数を既存のものと比べ約 8 倍に拡張することができた．

New Classes of Weak Keys on RC4 using Predictive State

Atsushi Nagao† Toshihiro Ohigashi‡ Takanori Isobe† Masakatu Morii†

†Graduate School of Engineering, Kobe University,
1-1 Rokkodai, Nada, Kobe, 657-8501, Japan
a.nagao@stu.kobe-u.ac.jp
mmorii@kobe-u.ac.jp

‡Information Media Center, Hiroshima University,
1-4-2 Kagamiyama, Higashihiroshima, 739-8511 Japan
ohigashi@hiroshima-u.ac.jp

Abstract In 2001, Mantin and Shamir had indicated the existence of predictive state which predicts a part of keystream from a few bytes of the internal state of PRGA in RC4. In 2011, Teramura et al. generalized classes of weak keys that lead to the predictive state, and exploited it for constructing a key recovery attack. They regard the keys generating predictive state with the highest probability as weak keys, and didn't consider the other class' weak key. In this paper, we propose a new class of weak keys in which a part of equations for weak keys defined by Teramura et al. hold, i.e. we succeed in removing some equations for weak keys. This enables us to increase the number of weak keys, while time complexity for attacks is larger than Teramura et al.'s attack. As a result, the space of our weak keys is about 8 times larger than that of Teramura et al. under the condition that the predictive state which recovers 5 bytes of internal state is used.

1 Introduction

RC4 [1] is a stream cipher designed by Rivest in 1987. It is widely used in security protocols such as Secure Sockets Layer (SSL), Transport Layer Security (TLS) [2], Wired Equivalent Privacy (WEP) [3] and Wi-Fi Protected Access (WPA) [4].

As the attack for RC4, several internal state reconstruction attacks have been proposed [5–10]. Some attacks of these attacks exploit the particular state after key scheduling called predictive state. The predictive state generates unique patterns in the keystream and allows an attacker to recover parts of the internal state from the keystream with a non-negligible probability. In addition, several key recovery attacks on RC4 have been proposed. Roos showed that initial few bytes of a keystream are strongly correlated with the key, and developed a key attack in the weak key setting [11]. After that, Teramura et al. expanded Roos' weak key by utilizing the predictive state and defined it as the generalized class [12]. They showed a property of secret key of secret key that generates a predictive state, and generalized a class of weak keys based on its property. In addition, Teramura et al. constructed a key recovery attack using those weak keys. However, they didn't mention about the space of the weak key deeply.

In this paper, we extend Teramura et al.'s weak key. Teramura et al.'s method dealt only with the most promising keys in generating a predictive state, namely their method needs a relational equations in the secret key to configure a b-predictive a-state. However in practice, keys which do not included in Teramura et al.'s weak keys can lead to the same predictive state. We have discover that the same predictive state can be derived even if a few equations are omitted from Teramura et al.'s weak key. Although those time complexity is inferior to Teramura et al.'s weak key, the space of weak keys are much larger than Teramura et al.'s weak key. When using the predictive state made by 5 state, the number of weak keys increased from $2^{100.91}$ to $2^{103.78}$. Thus, the space of the secret key should not be used in RC4 has become larger.

2 RC4

In this section, we briefly review the algorithm of the stream cipher RC4.

Algorithm 1 KSA

for $i^* = 0$ to $N-1$ **do**
  $S^*[x] \leftarrow x$
end for
$j^* \leftarrow 0$
for $i^* = 0$ to $N-1$ **do**
  $j^* \leftarrow j^* + S^*[i^*] + K[i^* \mod L]$
  Swap $S^*[i^*]$ and $S^*[j^*]$
end for

Algorithm 2 PRGA

$i \leftarrow 0$
$j \leftarrow 0$
loop
  $i \leftarrow i + 1$
  $j \leftarrow j + S[i]$
  Swap $S[i]$ and $S[j]$
  Output $Z \leftarrow S[S[i] + S[j]]$
end loop

2.1 Description of RC4

The stream cipher RC4 consists of two algorithms: Key Scheduling Algorithm (KSA) and Pseudo-Random number Generation Algorithm (PRGA). The KSA initializes an $(N+2)$-byte internal state by using an L-byte secret key K. In general, N is equal to 256 and L is equal to 16. The PRGA generates a pseudo-random output stream Z from the internal state initialized by the KSA. Details of KSA and PRGA are described in Algorithm 1 and Algorithm 2. In this paper, KSA and PRGA are distinguished by a star (*), which is attached to variables at KSA. RC4's internal state consists of an N-byte permutation array S and two indices i and j. Let $S_i[x]$ be the value of the array S at the index x and $S_i^{-1}[y]$ be the index of the value y in the array S after the i-th round in the PRGA, respectively. Then j_i is the value of j during the i-th round where the rounds are indexed with respect to i. We define Z_i as the output after the i-th round in the PRGA. In RC4, all operations are carried out under modulo N.

2.2 Predictive State

Fluhrer and McGrew have observed stronger correlations between keystreams and the initial states, and introduced the notion as fortuitous state [6]. This is a special class of internal states defined by the values of i_t, j_t, and several state elements of array S at the t-th round, which predict the keystream. Mantin and Shamir expanded and generalized the notion of fortuitous state as predictive state [7]. The predictive state is defined as follows.

Definition 1 An a-state is a partially specified RC4 states, that includes i_t, j_t and a elements of array S at the t-th round $(S_t[x_1], \ldots, S_t[x_a])$.

Definition 2 Let X_a be a set of internal states that includes an a-state and let the internal state in X_a. If every S_a outputs a same sequence of b bytes then a-state is said to be b-predictive.

When $n = 8$, an example of $(b = 5)$-predictive $(a = 5)$-state at $(t = 0)$-th round is given as

$$\begin{cases} S_0[1] = 1, \\ S_0[2] = 2, \\ S_0[3] = 255, \\ S_0[4] = 254, \\ S_0[5] = 3, \\ i_0 = 0, j_0 = 0. \end{cases} \quad (1)$$

If five elements of the internal state satisfy Eq.(1), the internal state necessarily outputs

$$\begin{cases} Z_1 = 2, \\ Z_2 = 1, \\ Z_3 = 2, \\ Z_4 = 1, \\ Z_5 = 3. \end{cases} \quad (2)$$

It is regardless of the values of other 251 elements in S_0. In this paper, unless otherwise noted, we consider only the predictive state at 0-th round.

3 Teramura et al.'s Weak Key

In 2011, Teramura et al. generalized classes of weak keys. Teramura et al.'s weak key was able to lead to the predictive state and outputs the particular keystream which predicted by the predictive state. Teramura et al. exploited it for constructing a key recovery attack using those keys, the attack using the keys which can lead to the 5-predictive 5-state can recover 40 bits of the secret key with probability $2^{-7.1}$.

3.1 Outline

Teramura et al.'s weak key which may lead to b-predictive a-state is expressed in a equations configured by $K[0], \ldots, K[a]$. The a equations in the secret key becomes b-predictive a-state at t-th round through the three steps in the KSA and PRGA.

Step 1 Convert a equations in secret key to a condition of $S^*_{x_1}[x_1], \ldots, S^*_{x_a}[x_a]$ in KSA. Its success probability is denoted by $P[E_{S_1}]$.

Step 2 Convert a condition in KSA to a condition of $S_0[x_1], \ldots, S_0[x_a]$ in first round of PRGA. Its success probability is denoted by $P[E_{S_2}]$.

Step 3 Convert a condition in first round of PRGA to a condition of $S_t[x_1], \ldots, S_t[x_a]$ in t-th round of PRGA. Its success probability is denoted by $P[E_{S_3}]$. If $t = 0$, we need not do this step.

When $K[0], \ldots, K[a]$ satisfy the equations, the weak key can be obtained at $P[leak] = P[E_{S_1}] \cdot P[E_{S_2}] \cdot P[E_{S_3}]$ of b-predictive a-state at t-th round.

3.2 Notation of Weak Key

Teramura et al. generalized the relational equation as follows:

$$\sum_{x=0}^{L-1} w_{x,y} K[x] = W_y, \quad (3)$$

where $w_{x,y} \in \{0, 1\}$, $W_y \in \{0, 1, \ldots, N-1\}$ and $y = \{0, 1, \ldots, a\}$ in b-predictive a-state.

3.3 Step 1

In this section, we discuss how to derive the relational equations that assign an arbitrary value X_i to $S^*_{x_i+1}[x_i]$. From the operation of the KSA described in Algorithm 1, we represent the value of $j^*_{x_i+1}$ as follows:

$$\begin{aligned} j^*_{x_i+1} &= j^*_{x_i} + S^*_{x_i}[x_i] + K[x_i] \\ &= \sum_{t=0}^{x_i} K[t] + \sum_{t=0}^{x_i} S^*_t[t]. \end{aligned} \tag{4}$$

Remember that the value of $S^*_{x_i}[j^*_{x_i+1}]$ is assigned to $S^*_{x_i+1}[x_i]$. Thus, $X_i = S^*_{x_i+1}[x_i]$ is satisfied if the following equation holds.

$$S^{*\,-1}_{x_i}[X_i] = j^*_{x_i+1} = \sum_{t=0}^{x_i} K[t] + \sum_{t=0}^{x_i} S^*_t[t], \tag{5}$$

where X_i is the value stored in $S_t[x_i]$ at predictive state with finally, and $S^{*\,-1}_{x_i}[X_i]$ is the index of the value X_i.

From Eq.(5), we derive the relational equation:

$$\sum_{t=0}^{x_i} K[t] = S^{*\,-1}_{x_i}[X_i] - \sum_{t=0}^{x_i} S^*_t[t]. \tag{6}$$

Here X_i is an arbitrary value and x_i is an index satisfying $0 \le x_i \le L-1$. Therefore, the condition to be satisfied by $w_{x,y}$ and W_y is denoted as follows in i-th equation.

$$w_{x,i} \begin{cases} 1 & (0 \le x \le x_i) \\ 0 & (x_i + 1 \le x \le L-1) \end{cases} \tag{7}$$

$$W_i = S^{*\,-1}_{x_i}[X_i] - \sum_{t=0}^{x_i} S^*_t[t] \tag{8}$$

If a key has met the above condition, full control of the early rounds of the KSA. Therefore, the success probability of Step 1 is $P[E_{S_1}] = 1$.

3.4 Step 2

Step 2 inherits the value of $S^*_{x_i}[x_i]$ in KSA to $S_0[x_i]$ in PRGA. If $j^*_t \neq x_i$ for all $t \in \{x_i+1, \ldots, N-1\}$, the value of $S^*_{x_i+1}[x_i]$ is unchanged. In order to meet this, pointer j^* must not pointed to x_i from x_i-th round to $(N-1)$-th round. Therefore, this probability is $(\frac{N-1}{N})^{N-x_i-1}$. The probability be satisfied for all x_i is

$$\begin{aligned} P[E_{S_2}] &\simeq \prod_{i=1}^{a} \left(\frac{N-1}{N}\right)^{N-x_i-1} \\ &= \left(\frac{N-1}{N}\right)^{a(N-1)-\sum_{i=1}^{a} x_i}. \end{aligned} \tag{9}$$

3.5 Step 3

Step 3 converts S_0 to S_t. However the contents of Step 3 is not directly related to new weak key, so omit the details. Briefly, Step 3 is dependent on the value of t, especially when $t = 0$ is $P[E_{S_3}] = 1$, because there is no need to manipulate. When $t \ge 1$, it is necessary that the pointer j does not point to $S_i[x_1], \ldots, S_i[x_a]$ for all $i \in \{0, \ldots, t\}$.

4 New Weak Key

We find new weak keys around Teramura et al.'s weak keys. Those keys may lead to same predictive states with Teramura et al.'s weak keys. The time complexity for attacks using new weak keys is inferior to that of Teramura et al.'s attack. However, the space of new weak keys are much larger than Teramura et al.'s weak key.

4.1 Idea

Teramura et al.'s method have to take over the condition of KSA to that of PRGA at Step 2. Then, it was important that pointer j^* does not point to the elements of S^* storing $x_1, \ldots, x_a$. On the contrary, we think that the key may lead to the same predictive state in the case that j^* work well by accident even if a few equations in Teramura et al.'s weak key are not satisfied. As a result, we find deriving weak keys by performing some processes in Step 1 and Step 2 in Teramura et al.'s weak keys. Figure 1 shows the conceptual diagram.

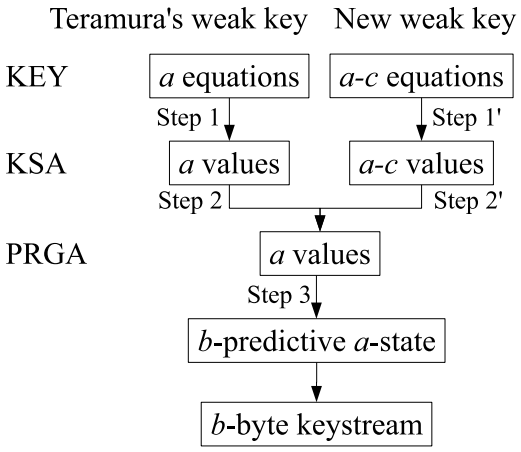

Figure 1: Conceptual diagram

4.2 Step 1'

In Teramura et al.'s weak key, a equations were required between $K[0], \ldots, K[a]$. Here, we consider that the case of c equations are not satisfied. Equations are omitted from last one that contains $K[a]$. In other words, a new weak key have $(a-c)$ equations using $K[0], \ldots, K[a-c]$, otherwise the remaining equations work incorrect. Because, if the value of j^* has changed, the later value of j^* couldn't be predicted and the value of W_i would change. Therefore, we don't consider omitting the middle equation at first. Then, in the Step 1', the remaining $(a-c)$ equations convert to condition of $S^*_{x_0}[x_1], \ldots, S^*_{x_{a-c}}[x_{a-c}]$ as with Teramura et al.'s weak key. The probability that Step 1' holds is $P[E_{S'_1}] = 1$.

4.3 Step 2'

In Step 2', we inherit the values of $S^*_{x_0}[x_1], \ldots, S^*_{x_{a-c-1}}[x_{a-c}]$ in KSA to $S_0[x_1], \ldots, S_0[x_{a-c}]$ in PRGA. At the same time, $S_0[x_{a-c+1}], \ldots, S_0[x_a]$ must take the desired value by fortuity. Thus, the new weak key joins with Teramura et al.'s weak key at the end of Step 2'. The new weak key gets the predictive state in the same Step 3 as Teramura et al.'s weak key. Then, we describe Step 2' in detail. First, $S_0[x_0], \ldots, S_0[x_{a-c}]$ is the same as Teramura et al.'s weak key, this probability is

$$\left(\frac{N-1}{N}\right)^{(a-c)(N-1)-\sum_{i=1}^{a-c} x_i}. \quad (10)$$

On the other hand, $S_0[x_{a-c+1}], \ldots, S_0[x_a]$ are assumed to be random values, the probability of obtaining a set of the correct value is $\left(\frac{1}{N}\right)^c$. Therefore, the probability that Step 2' holds is

$$P[E_{S'_2}] = \left(\frac{N-1}{N}\right)^{(a-c)(N-1)-\sum_{i=1}^{a-c} x_i} \left(\frac{1}{N}\right)^c. \quad (11)$$

4.4 Time Complexity

In this section, we evaluate the time complexity of the new weak key.

Initially, we explain the method of the key recovery attack using new weak keys. As a prerequisite, it is necessary to obtain a sufficient number of pairs keystreams and secret keys. And we extract the keystreams which match a predictive state's keystream. We assume that those keystreams were extracted from the weak keys, and we do the exhaustive search to the parts which are outsides of weak key's condition. The time complexity of the exhaustive search is $N^{L-(a-c)}$. Since the probability of the keystreams were extracted from the weak keys is defined $P[E_{key}|E_{out}]$, the conclusive time complexity is $N^{L-(a-c)}/P[E_{key}|E_{out}]$. The probability that the desired keystream obtained from a random key is

$$P[E_{out}] \simeq \left(\frac{1}{N}\right)^b. \quad (12)$$

And the probability of obtaining a new weak key consisting of the $(a-c)$ equations is

$$P[E_{key}] = \left(\frac{1}{N}\right)^{a-c}. \quad (13)$$

The probability that a given keystream from a predictive state be obtained is

$$\begin{aligned} P[E_{out}|E_{key}] &= P[E_{S'_2}] + P[rand] \\ &\simeq \left(\frac{N-1}{N}\right)^{(a-c)(N-1)-\sum_{i=1}^{a-c} x_i} \left(\frac{1}{N}\right)^c. \end{aligned} \quad (14)$$

In Eq.(14), we provide that the probability of obtaining the same keystream from non-predictive state $P[rand] \simeq P[E_{out}]$ is sufficiently lower than $P[E_{S_2'}]$ is required. From Bayes' theorem, the probability that a specified keystream derived from a new weak key is

$$P[E_{key}|E_{out}] = \frac{P[E_{key}]}{P[E_{out}]} \cdot P[E_{out}|E_{key}] \quad (15)$$

$$\simeq \left(\frac{1}{N}\right)^{a-b-c} \cdot P[E_{S_2'}]$$

$$\simeq \left(\frac{N-1}{N}\right)^{(a-c)(N-1)-\sum_{i=1}^{a-c} x_i} \left(\frac{1}{N}\right)^{a-b}. \quad (16)$$

Therefore, when we try to attack using a new weak key which omits c equations, the time complexity is

$$\frac{N^{L-(a-c)}}{P[E_{key}|E_{out}]}$$

$$\simeq \left(\frac{N}{N-1}\right)^{(a-c)(N-1)-\sum_{i=1}^{a-c} x_i} \cdot N^{L-b+c}. \quad (17)$$

When $N = 256, L = 16$, which is common parameters in RC4, the time complexity is

$$\left(\frac{255}{256}\right)^{255(a-c)-\sum_{i=1}^{a-c} x_i} 256^{16-b+c}. \quad (18)$$

Table 1 shows the time complexity of the new weak key for b-predictive a-state without c equations. When $c = 0$, this is Teramura et al.'s weak key. It is told that new weak key which omitted c equations has the same performance as Teramura et al.'s weak key using $(b-c)$-predictive $(a-c)$-state. When 5-predictive 5-state is used and 1 equation is omitted, the weak keys that apply to this condition can recover secret key at $2^{101.70}$.

4.5 Number of Weak Keys

In this section, we discuss the number of weak keys which certainly lead to the specified keystreams. The upper limit of the number of strict weak keys to lead the b-predictive a-state is given by $256^{16-a} = 2^{128-8a}$ when $N = 256, L = 16$ is used. Incidentally, the number of strict weak keys does not have a relation directly, because the key recovery attack uses the predictive state, but not each weak key.

Table 1: Time complexity

a	b	$c=0$	$c=1$	$c=2$	$c=3$	$c=4$
2	1	$2^{122.86}$	–	–	–	–
	2	$2^{114.86}$	$2^{121.43}$	–	–	–
3	2	$2^{116.29}$	$2^{122.86}$	–	–	–
	3	$2^{108.29}$	$2^{114.86}$	$2^{121.43}$	–	–
4	2	$2^{117.70}$	$2^{124.29}$	–	–	–
	3	$2^{109.70}$	$2^{116.29}$	$2^{122.86}$	–	–
	4	$2^{101.70}$	$2^{108.29}$	$2^{114.86}$	$2^{121.43}$	–
5	3	$2^{111.11}$	$2^{117.70}$	$2^{124.29}$	–	–
	4	$2^{103.11}$	$2^{109.70}$	$2^{116.29}$	$2^{122.86}$	–
	5	$2^{95.11}$	$2^{101.70}$	$2^{108.29}$	$2^{114.86}$	$2^{121.43}$

Table 2: Number of weak keys from one predictive state

a	$c=0$	$c=1$	$c=2$	$c=3$	$c=4$
2	$2^{109.14}$	$2^{110.57}$	–	–	–
3	$2^{99.71}$	$2^{101.14}$	$2^{102.57}$	–	–
4	$2^{90.30}$	$2^{91.71}$	$2^{93.14}$	$2^{94.57}$	–
5	$2^{80.89}$	$2^{82.30}$	$2^{83.71}$	$2^{85.14}$	$2^{86.57}$

We describe the number of weak keys obtained from one predictive state to the first, and then we describe the total number of weak keys from all predictive states that have been discovered so far. First, about the number of weak keys obtained from one predictive state. The number of keys that may lead to the b-predictive a-state is 256^{16-a}. By multiplying $P[E_{out}|E_{key}]$ by there, the number of strict weak keys be obtained. Table 2 is summarize the results. As c increases, the number of weak keys increases. However, we must note that the weak keys of a large c contain those of a small c. Second, we consider the number of weak keys that integrate all predictive states. The total number of b-predictive a-state in $a \leq 5, b \leq 5$ is given by Teramura et al. [12]. If there are more than one predictive state of the same a, the weak keys from those predictive state are independent of each other. For example, 5-predictive 5-state has

Table 3: Number of weak keys from all predictive states

a	b	Teramura et al.	New weak key
2	1	$2^{117.28}$	$2^{117.28}$
	2	$2^{109.14}$	$2^{110.57}$
	total	$2^{117.29}$	$2^{117.30}$
3	2	$2^{105.82}$	$2^{107.25}$
	3	0	0
	total	$2^{105.82}$	$2^{107.25}$
4	2	$2^{107.45}$	$2^{108.86}$
	3	$2^{100.54}$	$2^{103.38}$
	4	$2^{93.30}$	$2^{97.57}$
	total	$2^{107.46}$	$2^{108.90}$
5	3	$2^{100.89}$	$2^{103.72}$
	4	$2^{95.07}$	$2^{99.32}$
	5	$2^{87.04}$	$2^{92.72}$
	total	$2^{100.91}$	$2^{103.78}$

been found 71 types, each strict weak key corresponding to 5-predictive 5-state are all different. However, there is a possibility that the predictive state which has large a is extended from the one which has small a. Thus, it is impossible to sum values between different a, it is necessary to consider the relationship of each predictive state to determine the total number of weak keys. This has not been investigated currently, it is a challenge for the future. Therefore, the total number of a-state weak keys is derived by multiplying the number of predictive states by the value of Table 3. Since the 3-predictive 3-state is not exist, the number of weak keys is 0. When $a = 5$, the total number of weak keys increased from $2^{100.91}$ to $2^{103.78}$.

5 Experiment

We are verified probabilities using 2-predictive 2-state. the predictive state used in the experiment is shown in

$$\begin{cases} S_0[1] = 1, \\ S_0[2] = 0, \\ i_0 = 0, j_0 = 0. \end{cases} \tag{19}$$

In this case, the expected keystream is

$$\begin{cases} Z_1 = 0, \\ Z_2 = 0. \end{cases} \tag{20}$$

Table 4: Comparison of probability

	Theoretical	Experimental
$P[E_{key}]$	2^{-8}	2^{-8}
$P[E_{out}]$	2^{-16}	$2^{-14.44}$
$P[E_{out}\|E_{key}]$	$2^{-9.43}$	$2^{-9.40}$
$P[E_{key}\|E_{out}]$	$2^{-1.43}$	$2^{-2.97}$

Teramura et al.'s weak key which corresponds to this predictive state is

$$\boldsymbol{w} = \begin{pmatrix} 1 & 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 1 & 0 & \dots & 0 \end{pmatrix}, \tag{21}$$

$$\boldsymbol{W} = \begin{pmatrix} S_1^{*-1}[1] - \sum_{t=0}^{1} S_t^*[t] \\ S_2^{*-1}[0] - \sum_{t=0}^{2} S_t^*[t] \end{pmatrix}. \tag{22}$$

Then, we omit the second equation as a new weak key. As a result, the key condition is only

$$K[0] + K[1] = S_1^{*-1}[1] - \sum_{t=0}^{1} S_t^*[t]. \tag{23}$$

Then, we derive the probability by experiment. The result is shown in Table 4. For $P[E_{out}|E_{key}]$, we find that the theoretical value was a good approximation to the experimental value. However, the error occurs in $P[E_{out}]$, and this error would have been propagated to $P[E_{key}|E_{out}]$. Since this error is caused by the bias in RC4, we can get a value closes to the theoretical value at $P[E_{key}|E_{out}]$ when $P[E_{out}]$ has no bias.

6 Conclusion

In this paper, we have identified the new class of weak key on RC4. We have discovered that the same predictive state can be derived even if a few equations are omitted from Teramura et al.'s weak key. When using the predictive state made by 5-state, the number of weak keys increased from $2^{100.91}$ to $2^{103.78}$. Thus, the space of the secret key should not be used in the RC4 became larger.

Acknowledgements

This work was supported in part by Grant-in-Aid for Scientific Research (C) (KAKENHI

23560455) for Japan Society for the Promotion of Science.

References

[1] B. Schneier and P. Sutherland. *"Applied cryptography: protocols, algorithms, and source code in C"*. John Wiley & Sons, Inc., 1995.

[2] A. Frier, P. Karlton, and P. Kocher. "The SSL 3.0 protocol". *Netscape Communications Corp*, Vol. 18, p. 2780, 1996.

[3] IEEE Computer Society LAN MAN Standards Committee, et al. "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications", 1997.

[4] W.F. Alliance. "Wi-Fi protected access". *http://www. weca. net/ opensection/ protected_access. asp*, 2003.

[5] L. Knudsen, W. Meier, B. Preneel, V. Rijmen, and S. Verdoolaege. "Analysis methods for (alleged) RC4". Advances in Cryptology-ASIACRYPT'98, pp. 327–341. Springer, 1998.

[6] S. Fluhrer and D. McGrew. "Statistical analysis of the alleged RC4 keystream generator". In *Fast Software Encryption*, pp. 66–71. Springer, 2001.

[7] I. Mantin and A. Shamir. "A practical attack on broadcast RC4". In *Fast Software Encryption*, pp. 87–104. Springer, 2002.

[8] S. Paul and B. Preneel. "Analysis of non-fortuitous predictive states of the RC4 keystream generator". *Progress in Cryptology-INDOCRYPT 2003*, pp. 67–70, 2003.

[9] I. Mantin. "Predicting and distinguishing attacks on RC4 keystream generator". *Advances in Cryptology–Eurocrypt 2005*, pp. 551–551, 2005.

[10] A. Maximov and D. Khovratovich. "New state recovery attack on RC4". *Advances in Cryptology–CRYPTO 2008*, pp. 297–316, 2008.

[11] A. Roos. "A class of weak keys in the RC4 stream cipher", 1995.

[12] R. Teramura, T. Ohigashi, H. Kuwakado, and M. Morii. "Generalized classes of weak keys on RC4 using predictive state". *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. 94, No. 1, pp. 10–18, 2011.