Dependence in IV-Related Bytes of RC4 Key Enhances Vulnerabilities in WPA

Sourav Sen Gupta[1(✉)], Subhamoy Maitra[1], Willi Meier[2], Goutam Paul[1], and Santanu Sarkar[3]

[1] Indian Statistical Institute, Kolkata, India
sg.sourav@gmail.com,{subho,goutam.paul}@isical.ac.in
[2] FHNW, Windisch, Switzerland
willi.meier@fhnw.ch
[3] Chennai Mathematical Institute, Chennai, India
sarkar.santanu.bir@gmail.com

Abstract. The first three bytes of the RC4 key in WPA are public as they are derived from the public parameter IV, and this derivation leads to a strong mutual dependence between the first two bytes of the RC4 key. In this paper, we provide a disciplined study of RC4 biases resulting specifically in such a scenario. Motivated by the work of AlFardan et al. (2013), we first prove the interesting sawtooth distribution of the first byte in WPA and the similar nature for the biases in the initial keystream bytes towards zero. As we note, this sawtooth characteristics of these biases surface due to the dependence of the first two bytes of the RC4 key in WPA, both derived from the same byte of the IV. Our result on the nature of the first keystream byte provides a significantly improved distinguisher for RC4 used in WPA than what had been presented by Sepehrdad et al. (2011–2012). Further, we revisit the correlation of initial keystream bytes in WPA to the first three bytes of the RC4 key. As these bytes are known from the IV, one can obtain new as well as significantly improved biases in WPA than the absolute biases exploited earlier by AlFardan et al. or Isobe et al. We notice that the correlations of the keystream bytes with publicly known IV values of WPA potentially strengthen the practical plaintext recovery attack on the protocol.

Keywords: RC4 · WPA · Bias · Key correlation · Plaintext recovery

1 Introduction

The RC4 stream cipher and several modifications thereof (incorporated in various security protocols) have undergone rigorous analysis in cryptographic literature. The importance and timeliness of this topic is evident from the rich history

S. Sen Gupta—Supported by DRDO sponsored project Centre of Excellence in Cryptology (CoEC), under MOC ERIP/ER/1009002/M/01/1319/788/D(R&D) of ER&IPR, DRDO.

C. Cid and C. Rechberger (Eds.): FSE 2014, LNCS 8540, pp. 350–369, 2015.
DOI: 10.1007/978-3-662-46706-0_18

of research in RC4 over the last two decades. Among the several directions of cryptanalytic research in this area, the two most important aspects have been

1. correlation between the keystream bytes with absolute values, and
2. correlation between the keystream bytes with the Key and/or IV.

The results under item 1 have been extensively used in the broadcast attack model, and some important results in this area can be found in [1,6,8,9,11,19]. These biases directly work on the generic RC4 cipher [6] as well when RC4 is used in protocols like WPA and TLS [1]. In particular, the work of [1] received a lot of attention due to its impact on commercial protocols.

The results under item 2 explains how the RC4 keystream bytes may leak information regarding the secret key bytes. While there exist extensive research results in this area [8,13–15,19], no convincing key-recovery attack is yet available on RC4 using these biases. However, these biases work quite well in attacking protocols where some part of the RC4 key is derived from the public IV, as in the case of WEP [3,4,7,21,22,24].

To resist such attacks against WEP, the WPA [5] protocol had been proposed, where an incremental change in the IV results in a convoluted transformation of the remaining portion of the RC4 key. The two most recent and prominent attacks against WPA have been proposed by Sepehrdad et al. [20] and AlFardan et al. [1]. While the attack of [1] is based on the broadcast model for plaintext recovery, the work of [20] exploits certain weaknesses in the WPA key schedule to mount a key recovery attack with complexity 2^{96}, less than the exhaustive key search effort of 2^{104}. Before we proceed further to explain our contributions in this paper, let us describe RC4 and its usage in the WPA protocol.

We omit the mention of TKIP and refer to the WPA/TKIP protocol simply as WPA in this paper. In addition, we abuse the notation to refer to the instantiation of RC4 in this protocol as WPA, in contrast to standalone RC4.

1.1 Description of RC4

The RC4 cipher consists of a Key Scheduling Algorithm (KSA) and a Pseudo-random Generation Algorithm (PRGA). The internal state of RC4 is obtained as a permutation of $N = 256$ bytes, and the KSA produces the initial pseudorandom permutation of RC4 by scrambling an identity permutation using the secret key k. The secret key k of RC4 is of length typically between 5 to 32 bytes, which generates the expanded key K of length $N = 256$ bytes by simple repetition. If the length of the secret key $k = k_0, \ldots, k_{l-1}$ is l bytes (typically $5 \leq l \leq 32$), then the expanded key K is constructed as $K[i] = k_{i \bmod l}$ for $0 \leq i \leq N-1$. The initial permutation produced by the KSA acts as an input to the next procedure PRGA that generates the keystream, as depicted in Fig. 1.

For round $r = 1, 2, \ldots$ of RC4 PRGA, we denote the indices by i_r, j_r, the keystream output byte by Z_r, and the permutations before and after the swap by S_{r-1} and S_r respectively. All additions (subtractions) in context of RC4 are to be considered as 'addition (subtraction) modulo N', and all equalities in context of RC4 are to be considered as 'congruent modulo N'.

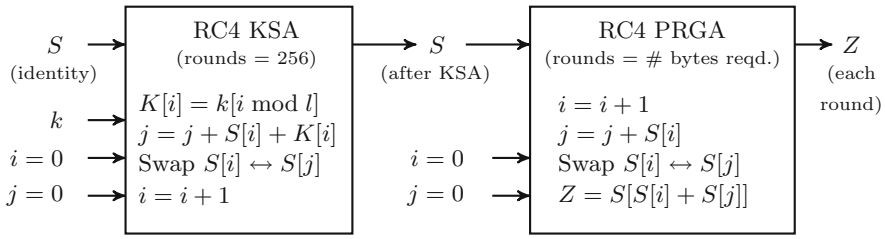

Fig. 1. Description of RC4 stream cipher.

1.2 Description of WPA

IEEE 802.11 standard protocol for WiFi security used to be Wired Equivalent Privacy (WEP), which was replaced by Wi-Fi Protected Access (WPA) in 2004. Both WEP and WPA use RC4 as their core cipher, and the WPA protocol can be thought of as a wrapper on top of WEP to provide good key mixing features. WPA introduces a key hashing module in the original WEP design to defend against the Fluhrer, Mantin and Shamir attack [4]. It also includes a message integrity feature and a key management scheme to avoid key reuse.

TKIP Key Schedule. WPA uses a 16-byte secret key for RC4 PRNG, the core encryption module of the system. This RC4 secret key RC4KEY is generated through a key schedule procedure known as TKIP [5], which takes as input a 128-bit *temporal key* TK (shared between the parties), transmitter's 48-bit MAC address TA and a 48-bit *initialization vector* IV, and passes those through two phases to obtain the final RC4 secret key.

In Phase 1, a 80-bit key P1K is generated from TK, TA and IV32, the upper 32 bits of the IV, using an unbalanced Feistel cipher with 80-bit block and 128-bit key structure. In Phase 2, the 128-bit RC4KEY is generated from TK, P1K (from Phase 1) and IV16, the lower 16 bits of the IV. In this phase, TK and P1K are mixed (using a temporary key PPK) to construct the last 104 bits (13 bytes) of the RC4KEY, and the first 24 bits (3 bytes) of the RC4KEY are constructed directly from the IV16, as follows [5, Annex H.1].

```
RC4KEY[0] = Hi8(IV16);                    /* top byte of IV16 */
RC4KEY[1] = (Hi8(IV16) | 0x20) & 0x7F;    /* avoid FMS attack */
RC4KEY[2] = Lo8(IV16);                    /* low byte of IV16 */
```

In the above expression, Hi8(IV16) and Lo8(IV16) indicate the top and lower bytes of IV16, respectively. RC4KEY[0] and RC4KEY[2] are simply two parts of the counter IV16, while RC4KEY[1] is purposefully constructed to avoid the known WEP attack by Fluhrer, Mantin and Shamir [4]. Once the 128-byte (16-byte) RC4KEY is prepared, it is directly used for encryption in the RC4 PRNG core of the protocol.

1.3 Contributions of This Paper

There is a growing concern regarding how far we should study the combinatorial nature of RC4 and protocols based on it. However, we can not help but notice the glaring implications of such studies in mounting practical attacks on commercial protocols that still handle a bulk of everyday network traffic. In this backdrop, we present the motivation and contribution of our paper as follows.

Motivation. To the best of our knowledge, the dependence of the first two bytes of the RC4 key, constructed from the public parameter IV during WPA key schedule, has not been studied thoroughly from a combinatorial viewpoint. We draw our motivation from two important questions in this direction.

1. How do the biases of keystream bytes towards absolute values differ for RC4 in WPA compared to those in case of generic RC4?
2. Are there any exploitable correlations between the keystream bytes and the first three key bytes of RC4 derived from the IV in WPA?

Contribution. Our results provide the first disciplined study of keystream non-randomness in RC4 when used in WPA. The study contains explanation of existing biases as well as discovery of new ones. The results have diverse applications in different cryptanalytic results, ranging from the best WPA distinguisher to improved broadcast attack against WPA.

Specific Outcomes of our First Motivation. We provide theoretical justification of some experimental observations on WPA, made by AlFardan et al. [1], to obtain further insight into such observations.

- In Sect. 2.3, we derive the complete sawtooth distribution of the first keystream byte Z_1 when RC4 is executed with IV's as in WPA.
- The biases in Z_1 gives a method to distinguish the keystream of WPA from that of generic RC4, with a packet complexity of approximately 2^{19}. Note that WPA may be considered as a 'mode of operation' for RC4, and our observation shows that this mode statistically deviates from the core cipher, where the deviation is visible with a considerably less number of packets. The previously known distinguisher of [20], first presented in Eurocrypt 2011, achieves a 0.5 probability of success in distinguishing WPA from generic RC4 with time complexity 2^{43} and packet complexity 2^{40}. Later in [18], the distinguisher was improved to achieve 0.5 probability of success in distinguishing WPA with time complexity 2^{42} and packet complexity 2^{42}.
- In Sect. 2.4, we show how the initial keystream bytes $Z_3, \ldots, Z_{255}$ of WPA are biased towards zero following a similar sawtooth pattern, and in Sect. 2.5, we provide a theoretical estimate for $(Z_r = r)$ better than [6].

Specific Outcomes of our Second Motivation. All biases of the keystream bytes in WPA presented in [1] are correlated to absolute values in $[0, 255]$, and the experimental study discovers that they are mostly of the order of $1/N^2$ over the probability of random association $1/N$. Indeed these are not of the same level

as the bias in the event $(Z_2 = 0)$, which is of the order $1/N$ over the probability of random association $1/N$, as proved in [11]. However, it is well known that there are quite a few significant biases of the keystream bytes with the initial key bytes of RC4 [7,8,13–15,17,19]. The first three bytes of the RC4 key in WPA are derived from the public parameter IV, and thus the correlation of keystream bytes with any combination of the first three RC4 key bytes can be successfully exploited in broadcast attack against WPA. Our investigations in this direction reveal the following results.

- There exist high biases in the keystream bytes Z_1, Z_2, Z_3, Z_{256}, Z_{257} towards the first three 'public' (IV-derived) bytes of the RC4 key in WPA. For the first time in the literature, we discover such hugely significant biases, matching the order of the $(Z_2 = 0)$ bias [11], even in the case of WPA.
- In a broadcast setting, we could recover the aforesaid bytes of the plaintext with probability close to 1 using only 2^{21} samples, in contrast with the existing works [1,6] that require 2^{30} samples for the same bytes.
- We explore some new biases in this line and present a detailed study on the correlations of the keystream bytes with different IV combinations in WPA.
- We also discover a new absolute bias at the keystream byte Z_{259}, the farthest known so far among the initial keystream bytes to have a significant bias.

An independent work [12] in a similar direction is to appear in FSE 2014.

2 Biases in WPA Resulting from TKIP Key Schedule

The first three bytes of the RC4 key in WPA is derived as in Eq. (1).

$$\begin{aligned} K[0] &= (\mathsf{IV16} >> 8) \,\&\, \mathsf{0xFF} & K[2] &= \mathsf{IV16} \,\&\, \mathsf{0xFF} \\ K[1] &= ((\mathsf{IV16} >> 8) \mid \mathsf{0x20}) \,\&\, \mathsf{0x7F} \end{aligned} \tag{1}$$

Note that a 2-byte IV16 is expanded to the initial 3 bytes of the key (Fig. 2), and the first two key bytes $K[0]$ and $K[1]$ have 6 bits in common, apart from the two fixed bits in $K[1]$. The third key byte $K[2]$ is independent of the first two bytes of the key. Thus, TKIP can generate only 2^{16}, and not 2^{24}, distinct values for the first 3 bytes of the RC4 secret key – a loss in entropy that we believe may result into some non-random behavior in the initial phases of the cipher.

2.1 Bias in $K[0] + K[1]$ for WPA

As $K[0]$ and $K[1]$ share 6 bits from the common source Hi8(IV16), we first take a look at their sum, $K[0] + K[1]$, for potential non-randomness. We notice that

1. $K[0] + K[1]$ must always be *even*, as $K[0]$ and $K[1]$ have the same LSB.
2. $K[1]$ can never exceed 127 as its MSB is 0. It can not even attain all possible values below 127, as its 6-th bit (from LSB side) is fixed at 1.
3. Values of $K[1]$ and $K[0] + K[1]$ strictly depend on the value/range of $K[0]$.

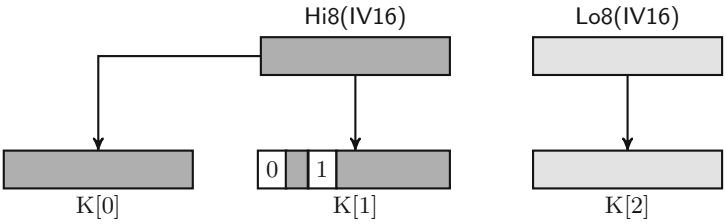

Fig. 2. Expansion of WPA IV16 into the first three bytes of the RC4 key.

These restrictions result in corresponding conditions on the range of $K[1]$ and $K[0]+K[1]$, depending on the range of $K[0]$. The complete set of conditions on the respective ranges is shown in Table 1, which results in a consolidated probability distribution of $K[0] + K[1]$ as described in Theorem 1.

Theorem 1. *The probability distribution of the sum of first two bytes of the RC4 key generated by TKIP key schedule in WPA, i.e., the distribution of* $\Pr(K[0] + K[1] = v)$ *for* $v = 0, 1, \ldots, 255$*, is as in Table 1:*

$$\begin{aligned}
&\Pr(K[0] + K[1] = v) = 0 && \text{if } v \text{ is odd;}\\
&\Pr(K[0] + K[1] = v) = 0 && \text{if } v \text{ is even and } v \in [0, 31] \cup [128, 159];\\
&\Pr(K[0] + K[1] = v) = 2/256 && \text{if } v \text{ is even and}\\
& && v \in [32, 63] \cup [96, 127] \cup [160, 191] \cup [224, 255];\\
&\Pr(K[0] + K[1] = v) = 4/256 && \text{if } v \text{ is even and } v \in [64, 95] \cup [192, 223].
\end{aligned}$$

Proof. The value of $K[0] + K[1]$ is always even, as discussed earlier. The value and range of $K[1]$, and hence that of $K[0]+K[1]$, depends on the range of $K[0]$;

Table 1. Probability distribution of $K[0] + K[1]$ resulting due to TKIP key schedule.

$K[0]$	$K[1]$ (depends on $K[0]$)		$K[0] + K[1]$ (only even)	
Range	Value	Range	Value	Range
0 – 31	$K[0] + 32$	32 – 63	$2K[0] + 32$	32 – 95
32 – 63	$K[0]$	32 – 63	$2K[0]$	64 – 127
64 – 95	$K[0] + 32$	96 – 127	$2K[0] + 32$	160 – 223
96 – 127	$K[0]$	96 – 127	$2K[0]$	192 – 255
128 – 159	$K[0] - 96$	32 – 63	$2K[0] - 96$	160 – 233
160 – 191	$K[0] - 128$	32 – 63	$2K[0] - 128$	192 – 255
192 – 223	$K[0] - 96$	96 – 127	$2K[0] - 96$	32 – 95
224 – 255	$K[0] - 128$	96 – 127	$2K[0] - 128$	64 – 127

$K[0] + K[1]$ (only even)	Prob. (0 if odd)
0 – 31	0
32 – 63	2/256
64 – 95	4/256
96 – 127	2/256
128 – 159	0
160 – 191	2/256
192 – 223	4/256
224 – 255	2/256

shown in Table 1. The probability distribution of $K[0]+K[1]$ may be calculated directly from this dependence pattern; also shown in Table 1. One may check

$$\underbrace{(128\times 0)}_{\text{odd values}}+\Bigg(16\times 0+16\times\frac{2}{256}+16\times\frac{4}{256} \\ +16\times\frac{2}{256}+16\times 0+16\times\frac{2}{256}+16\times\frac{4}{256}+16\times\frac{2}{256}\Bigg)=1,$$

to validate the consistency of the probability distribution of $K[0]+K[1]$. □

2.2 Bias in RC4 PRGA Initial Permutation S_0 for WPA

In 2007, Paul and Maitra [13] proved the famous Roos' biases [15], which state that the initial bytes of the permutation S_0 are biased towards the secret key bytes. $S_0[0]$ is biased towards $K[0]$, which is uniformly distributed, identical to the lower half of the counter IV16. For $S_0[1]$ however, we get the following result.

Theorem 2. *In case of WPA, the probability distribution of* $(S_0[1]=v)$ *for* $v=0,1,\ldots,N-1$, *after the completion of KSA, is given as*

$$\begin{aligned}\Pr(S_0[1]=v)=\ &\alpha\cdot\Pr(K[0]+K[1]=v-1)\\ &+(1-\alpha)\cdot(1-\Pr(K[0]+K[1]=v-1))\cdot\Pr(S_0[1]=v)_{RC4}\\ &+\frac{(1-\alpha)}{N-1}\cdot\sum_{x\neq v}\Pr(K[0]+K[1]=x-1)\cdot\Pr(S_0[1]=x)_{RC4},\end{aligned}$$

where $\alpha=1/N+(1-1/N)^{N+2}$, *and the probability terms* $\Pr(S_0[1]=v)_{RC4}$ *and* $\Pr(S_0[1]=x)_{RC4}$ *refer to the corresponding values in generic RC4.*

Proof. From the proof of Roos' biases in [13], we know that the initial permutation byte $S_0[1]$ is biased towards $K[0]+K[1]+1$ with a probability $\Pr(S_0[1]=K[0]+K[1]+1)\approx 1/N+(1-1/N)^{N+2}=\alpha$, say. Thus we write the probability distribution of $S_0[1]=v$ in case of WPA as follows.

$$\begin{aligned}\Pr(S_0[1]=v)=\ &\Pr(S_0[1]=v\ \wedge\ K[0]+K[1]+1=v)\\ &+\sum_{x\neq v}\Pr(S_0[1]=v\ \wedge\ K[0]+K[1]+1=x)\end{aligned}$$

The first event $(S_0[1]=v\ \wedge\ K[0]+K[1]+1=v)$ occurs if and only if the independent events $(S_0[1]=K[0]+K[1]+1)$ and $(K[0]+K[1]=v-1)$ occur simultaneously. This happens with probability $\alpha\cdot\Pr(K[0]+K[1]=v-1)$ where α is due to Roos' bias, and the second term is obtained from Theorem 1.

On the other hand, the event $(S_0[1]=v\ \wedge\ K[0]+K[1]+1=x)$ for $x\neq v$ may be further decomposed as follows

$$\begin{aligned}&\Pr(S_0[1]=v\ \wedge\ S_0[1]=K[0]+K[1]+1\ \wedge\ K[0]+K[1]+1=x)\\ &\quad+\Pr(S_0[1]=v\ \wedge\ S_0[1]\neq K[0]+K[1]+1\ \wedge\ K[0]+K[1]+1=x).\end{aligned}$$

The first term denotes an impossible condition (probability 0), and the second term can be computed as $\Pr(K[0]+K[1]=x-1)\cdot\Pr(S_0[1]\neq K[0]+K[1]+1)\cdot\Pr(S_0[1]=v \mid S_0[1]\neq x)$, that is, as

$$(1-\alpha)\cdot\Pr(K[0]+K[1]=x-1)\cdot(\Pr(S_0[1]=v)_{RC4}+\Pr(S_0[1]=x)_{RC4}/(N-1)),$$

where we assume that $(S_0[1]=v)$ and $(S_0[1]=x)$ occur exactly as in generic RC4 when $S_0[1]\neq K[0]+K[1]+1$, with appropriate probability normalization. We get the result after due simplification of the summation over $x\neq v$. □

For $N=256$, as in WPA and RC4, we get $\alpha\approx 0.368$ in Theorem 2. The probabilities $\Pr(K[0]+K[1]=v-1)$ and $\Pr(K[0]+K[1]=x-1)$ are taken from Theorem 1, and the probabilities $\Pr(S_0[1]=v)_{RC4}$ and $\Pr(S_0[1]=x)_{RC4}$ are taken from Proposition 1, derived in [10, Theorem 6.2.1].

Proposition 1 (from [10]). *After RC4 KSA, for* $0\leq u\leq N-1$, $0\leq v\leq N-1$,

$$\Pr(S_0[u]=v)=\begin{cases}\frac{1}{N}\left(\left(\frac{N-1}{N}\right)^v+\left(1-\left(\frac{N-1}{N}\right)^v\right)\left(\frac{N-1}{N}\right)^{N-u-1}\right), & \text{if } v\leq u;\\ \frac{1}{N}\left(\left(\frac{N-1}{N}\right)^{N-u-1}+\left(\frac{N-1}{N}\right)^v\right), & \text{if } v>u.\end{cases}$$

The theoretical distribution of $S_0[1]$ in WPA, thus produced from Theorem 2, is shown in Fig. 3. This distribution closely matches our experimental data, and differs significantly from the one for generic RC4 (as derived in [10]).

2.3 Bias in the First Keystream Byte Z_1 of WPA

Recall that in the first round of RC4 PRGA, the initial permutation entry $S_0[1]$ serves as $j_1=S_0[i_1]=S_0[1]$, and plays an important role in determining the

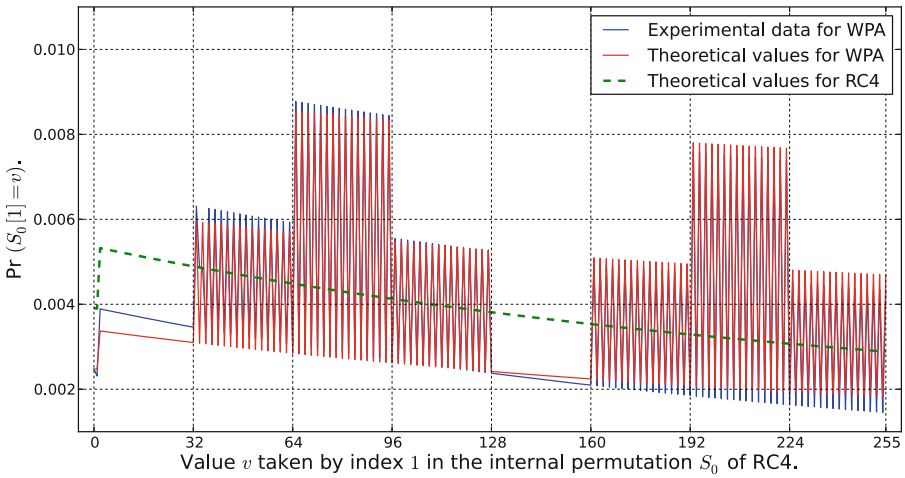

Fig. 3. Theoretical plot for $\Pr(S_0[1]=v)$ for RC4 and WPA, where $v=0,\ldots,255$.

first keystream byte $Z_1 = S_1[S_1[i_1] + S_1[j_1]] = S_1[S_0[S_0[1]] + S_0[1]]$. In fact, we know that $S_0[1]$ is prominent in the distribution of Z_1 proved by Sen Gupta et al. in [17, Theorem 13]. We reproduce the distribution as follows.

Proposition 2 (from [17]). *The probability distribution of the first output byte of RC4 keystream is as follows, where* $v \in \{0, \ldots, N-1\}$, $\mathcal{L}_v = \{0, 1, \ldots, N-1\} \setminus \{1, v\}$ *and* $\mathcal{T}_{v,X} = \{0, 1, \ldots, N-1\} \setminus \{0, X, 1-X, v\}$.

$$\Pr(Z_1 = v) = Q_v + \sum_{X \in \mathcal{L}_v} \sum_{Y \in \mathcal{T}_{v,X}} \Pr(S_0[1] = X \wedge S_0[X] = Y \wedge S_0[X+Y] = v);$$

$$Q_v = \begin{cases} \Pr(S_0[1] = 1 \ \wedge \ S_0[2] = 0), & \text{if } v = 0; \\ \Pr(S_0[1] = 0 \ \wedge \ S_0[0] = 1), & \text{if } v = 1; \\ \Pr(S_0[1] = 1 \ \wedge \ S_0[2] = v) + \Pr(S_0[1] = v \ \wedge \ S_0[v] = 0) & \\ \quad + \Pr(S_0[1] = 1 - v \ \wedge \ S_0[1-v] = v), & \text{otherwise.} \end{cases}$$

We consider two cases while computing the numeric values of $\Pr(Z_1 = v)$. If the initial permutation S_0 of RC4 PRGA is constructed from the regular KSA with random key, the probabilities $\Pr(S_0[u] = v)$ closely follow the distribution proved by Mantin in [10, Theorem 6.2.1]. However, if the initial permutation S_0 originates from RC4 KSA using TKIP-generated keys, as in the case with WPA, then $\Pr(S_0[1] = v)$ must be computed using Theorem 2, including its idiosyncratic biases for WPA shown in Fig. 3.

We compute the exact probabilities $\Pr(Z_1 = v)$ for RC4 and WPA using the estimation strategy of joint probabilities proposed in [17]; particularly estimating the joint probabilities $\Pr(S_0[X] = A \ \wedge \ S_0[Y] = B)$ as $\Pr(S_0[X] = A) \cdot (\Pr(S_0[Y] = B) + \Pr(S_0[Y] = A)/(N-1))$. The distribution of $S_0[1] = v$ is considered independently in each case. This results in two different distributions of Z_1; one for generic RC4 (same as [17]) and the other for RC4 in WPA.

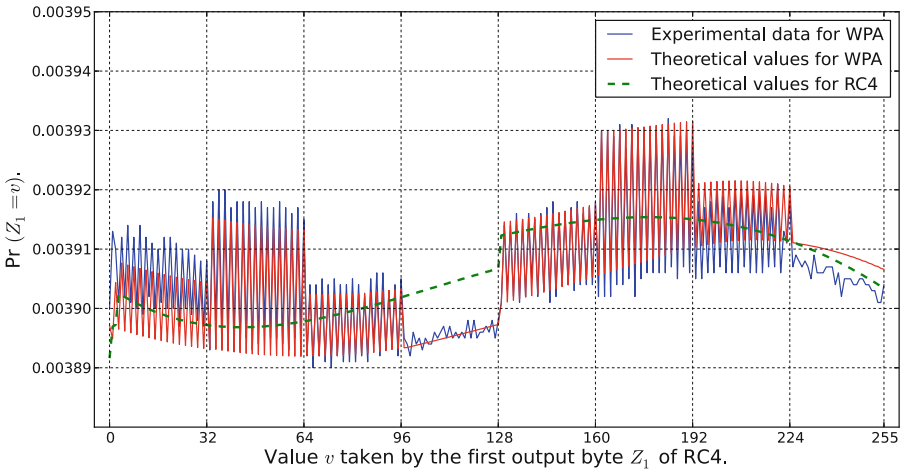

Fig. 4. Theoretical plot for $\Pr(Z_1 = v)$ for RC4 and WPA, where $v = 0, \ldots, 255$.

Figure 4 displays the two distributions, clearly pointing out the bias resulting in the PRGA as a result of TKIP key schedule, and shows that the theoretical distribution for WPA closely matches our experimental data.

Note that the patterns of these two theoretical distributions closely match the recent experimental observations of AlFardan et al. [1] (Fig. 10(a) in the full online version of the paper). The only difference is that there exist keylength dependent spikes at $Z_1 = 129$ for the observations in [1], as the experiments were done using 16-byte keys; whereas in our theoretical analysis, we disregard the keylength dependence altogether, and prove a general distribution of Z_1.

In fact, if WPA had employed RC4 with full-length 256-byte secret keys, where the first three bytes of the key $K[0], K[1], K[2]$ were constructed from the IV using TKIP key schedule principle (as in Eq. (1)), the pattern of the bias in Z_1 for WPA would have been the same. We have independently verified our theoretical results through experiments involving secret keys of various lengths.

Distinguishing WPA. We attempt to combine the values of Z_1 in suitable subsets of the support interval $\{0, 1, \ldots, 255\}$ to construct a distinguisher between WPA and generic RC4. The structure of the event considered for distinguishing WPA from RC4 in this case is '$e_S : (Z_1 \in S)$ where $S \subseteq \{0, 1, \ldots, 255\}$'. The subset S may be quite large, and thus the base probability $p = \Pr(e_S)$ in either distribution is not essentially small. In such a case, the distinguisher complexity may be estimated as $O(\frac{1-p}{pq^2})$.

Now we may define a suitable set S for the target distinguishing event. As most of higher biases are for *even* values of the first byte, we assume that the distributions of WPA and RC4 differ the most in cases when Z_1 takes an even value. Based on this intuition, we pick the set S as the set of all even values $\{0, 2, 4, \ldots, 254\}$ within the range; thus defining the distinguishing event as

$$e_S \ : \ (Z_1 = 2k \text{ for } k = 0, 1, \ldots, 127).$$

From our theoretical results on the distribution of Z_1 in WPA and RC4, as proved in Sect. 2.3, we estimate the following probabilities:

$$\left.\begin{aligned} p = \Pr(e_S) \text{ in RC4 } \& \approx 0.4999946 \\ p(1+q) = \Pr(e_S) \text{ in WPA } \& \approx 0.5007041 \end{aligned}\right\} \Rightarrow \quad q \approx 0.001419 \approx 0.363/N.$$

For $N = 256$, we require an estimated $8N^2 = 2^{19}$ keystream packets to distinguish WPA from generic RC4 with more than 70 % probability of success. This is the best distinguisher of WPA to date, improving the previous distinguishers of packet complexity more than 2^{40}, identified by Sepehrdad et al. [18,20].

It may be noted that some distinguishers of RC4 (compared to uniform random generators) remain equally effective in its WPA 'mode of operation', like the distinguisher based on $(Z_2 = 0)$. However, the sawtooth pattern of Z_1 is unique to WPA, and is not present in original RC4.

2.4 Bias Towards Zero in Bytes $Z_3, \ldots, Z_{255}$ of WPA

We extend the effect of the bias in S_0 of WPA to the biases in the initial keystream bytes towards zero. Maitra et al. [9] proved the biases of the initial keystream bytes $Z_3, \ldots, Z_{255}$ towards zero, and we reproduce their result from [17, Theorem14] in Proposition 3, as follows.

Proposition 3 (from [17]). *For RC4 PRGA rounds $3 \leq r \leq N-1$, the probability that $Z_r = 0$ is given by:*

$$\Pr(Z_r = 0) \approx \frac{1}{N} + \frac{c_r}{N^2}, \quad \textit{where}$$

$$c_r = \begin{cases} \frac{N}{N-1}\left(N \cdot \Pr(S_{r-1}[r] = r) - 1\right) - \frac{N-2}{N-1}, & \textit{for } r = 3; \\ \frac{N}{N-1}\left(N \cdot \Pr(S_{r-1}[r] = r) - 1\right), & \textit{otherwise.} \end{cases}$$

In [17], the computation of $\Pr(Z_r = 0)$ depended on the computation of $\Pr(S_{r-1}[r] = r)$, which in turn required the distributions of initial permutations S_0 and S_1 of RC4 PRGA (details in [17, Corollary 2] and [17, Lemma 1]).

We consider two cases – one in which the initial permutation S_0 is generated by generic RC4 KSA using random keys, and the other where S_0 is biased (Fig. 3) for using RC4 with keys originating from TKIP. These two cases produce two different distributions of $\Pr(Z_r = 0)$ for $r = 3, \ldots, 255$. The patterns closely match the experimental observations of AlFardan et al. [1] (Fig. 11 in the full online version of the paper) as well as our experimental data, as shown in Fig. 5.

2.5 Bias in $(Z_r = R)$ for WPA

Significant biases in the event $(Z_r = r)$ for $r = 3, \ldots, 255$ have recently surfaced in the context of plaintext recovery attack on RC4 [1,6], and these biases are

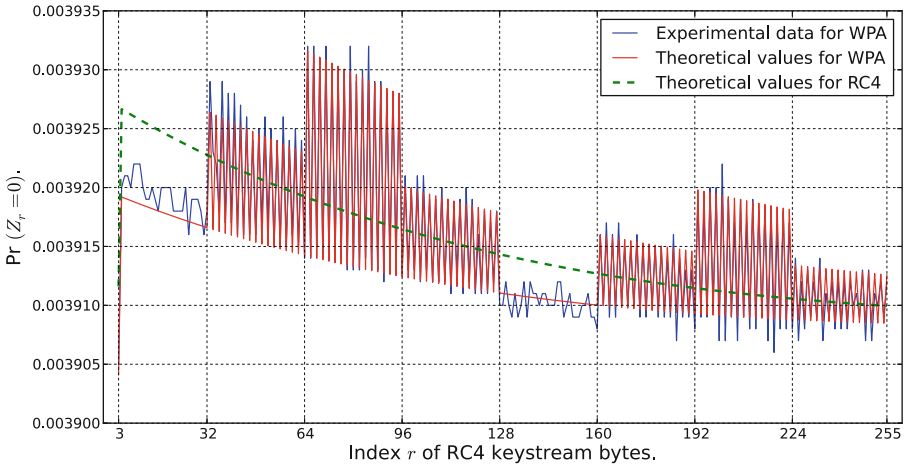

Fig. 5. Theoretical plot for $\Pr(Z_r = 0)$ for RC4 and WPA, where $r = 3, \ldots, 255$.

found to be more prominent than the previous ones for certain values of r. Isobe et al. identified these biases and attempted a proof in [6, Theorem 8], but the estimates did not 'exactly coincide with the experimental values'. Considering the significance of these biases in cryptanalysis of RC4, we explore an alternative avenue to estimate them, as detailed in Theorem 3.

Theorem 3. *For RC4 PRGA rounds $3 \le r \le N-1$, the probability that $Z_r = r$ is approximately*

$$\Pr(Z_r = r) = \frac{1}{N} + \Pr(S_0[1] = r) \cdot \frac{1}{N}\left(1 - \frac{1}{N}\right)\left(1 - \frac{r-2}{N}\right)\left(1 - \frac{2}{N}\right)^{r-3}.$$

Proof. The major path leading to the target event is as follows.

- Suppose that $S_0[1] = r$, i.e., $j_1 = r$ and $j_2 = r + S_1[2]$. This ensures that $S_1[r] = r$ after the first round of PRGA, and $S_2[j_2] = S_1[2]$ after the second.
- Suppose that $j_2 \neq 3, \ldots, r$, which occurs with probability $\left(1 - \frac{r-2}{N}\right)$. This ensures that i_r does not touch either of the locations r or j_2 till round $r-1$.
- Suppose that none of the indices $j_3, \ldots, j_{r-1}$ touches either of the locations r or j_2. This happens with probability $\left(1 - \frac{2}{N}\right)^{r-3}$ as $j_2 \neq r$, and ensures that after round $r-1$, we have $S_{r-1}[r] = r$ and $S_{r-1}[j_2] = S_1[2]$.
- Finally, suppose that $j_r = j_2$, which holds with probability $1/N$. This ensures that after round r, we have $S_r[r] = S_1[2]$ and $S_r[j_2] = r$.
- The final state results in $Z_r = S_r[r + S_1[2]] = S_r[j_2] = r$ with probability 1.

Considering the above events to be independent, the probability that the main path holds is given by $\alpha = \Pr(S_0[1] = r) \cdot \frac{1}{N}\left(1 - \frac{r-2}{N}\right)\left(1 - \frac{2}{N}\right)^{r-3}$. If the above path does not occur, then we assume that the event $(Z_r = r)$ happens due to random association, with probability $1/N$. Thus we can compute the target probability as $\Pr(Z_r = r) \approx \alpha + (1-\alpha)\frac{1}{N}$, and get the result. □

Figure 6 (upper plot) displays our theoretical result in comparison with that of Isobe et al. [6], where the experimental data for RC4 has been obtained from the authors of [1], and the values of $\Pr(S_0[1] = r)$ are obtained from Mantin's distribution [10] for S_0. It is evident that our theoretical values match the experimental data better than that of [6]. Note that the experimental values are for RC4 with 16-byte secret keys, and hence the data is non-smooth (with small spikes) at certain points. In contrast, the theoretical values of our result is for general RC4 with full-length secret keys, thus making the curve smooth.

It is interesting to note that RC4 in WPA exhibits enhanced non-random behavior in the events $(Z_r = r)$, as shown in Fig. 6 (lower plot) for 2^{32} runs of WPA. However, substituting the distribution of S_0 for WPA in our theoretical result (or that of [6]) does not match the experimental observations, and we believe that further investigation in this direction is necessary to settle the issue.

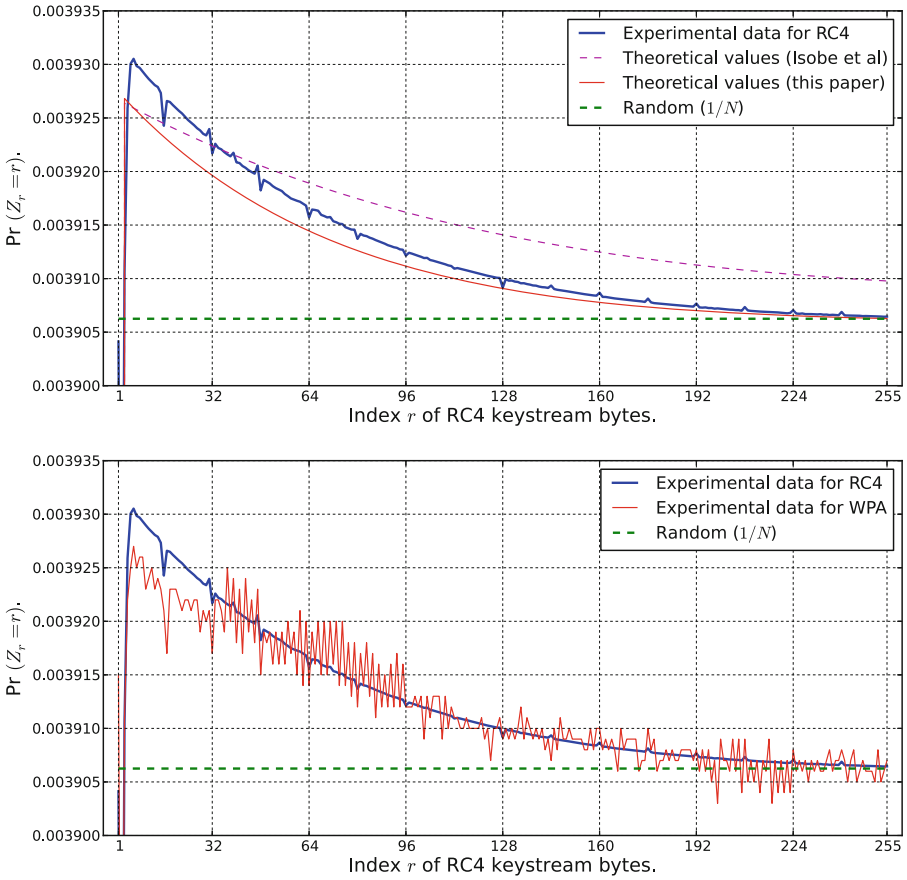

Fig. 6. Plot of $\Pr(Z_r = r)$ for RC4 and WPA, where $r = 3, \ldots, 255$. The experimental data for RC4 in these plots are obtained from the authors of [1].

3 Correlation of the Keystream Bytes with IV in WPA

The weaknesses in the WPA key schedule have recently been exploited twice in the literature of RC4 cryptanalysis – first by Sepehrdad et al. [18,20] and then by AlFardan et al. [1]. While Sepehrdad et al. [18,20] attacked the inner workings of the WPA key schedule to devise a key recovery attack with complexity 2^{96}, the recent work of AlFardan et al. [1] mounted a plaintext recovery attack on WPA by exploiting the biases of the keystream bytes towards absolute values. It was shown that the WPA key schedule, designed to prevent key recovery attacks, unintentionally made the plaintext recovery attack on RC4 even simpler. In this section, we target a third direction of attack – exploiting correlations of the keystream bytes towards the IV to perform plaintext recovery of WPA.

In WPA, the first three bytes of the RC4 key, $K[0], K[1], K[2]$, are derived from the IV. For the u-th recipient in a broadcast setting, let us denote these

bytes by $K_u[i]$ where $i = 0, 1, 2$ and $1 \leq u \leq n$. Note that the values of $K_u[i]$ are publicly known, and hence these could be exploited towards plaintext recovery attacks in case they have prominent correlations with the keystream bytes.

In this section, we will investigate for significant correlations between the keystream output bytes Z_r of WPA and certain linear combinations of the bytes $\{K[0], K[1], K[2]\}$. Let us assume that the number of such correlations with probability significantly different from $1/N = 1/256$ for the keystream byte Z_r is bounded above by Q_r. In this setting, we shall denote the corresponding linear combinations as $L_{r,q}(K[0], K[1], K[2])$, where $q = 1, 2, \ldots, Q_r$.

In Table 2, we list the most significant correlations of this kind for RC4 keystream bytes. Some of these are already known in the literature, and the ones identified by us will be pointed out clearly in the course of this discussion. The references for most of these biases can be found in [18, Figure 4.9]. We know that the bytes $K[0]$ and $K[1]$ are dependent in WPA, and hence the biases observed in RC4 may vary in case of WPA. Thus we first present detailed experimental data in Table 2 to explain the scenario. 'WPA (part)' denotes WPA with the first 3 key bytes constructed from the IV and next 13 key bytes chosen randomly, and 'WPA (full)' denotes WPA with first 3 key bytes constructed from the IV and next 13 key bytes generated by TKIP. We notice that 'WPA (part)' models 'WPA (full)' quite well, also observed earlier by [1].

In Table 2, note that there are certain cases where the biases in RC4 and WPA are not the same. In fact, there are some cases where the biases are in the opposite direction. There are also a few situations where there exist prominent biases in some cases but none in the others. Let us explain a couple of cases.

Table 2. Linear correlations observed between the keystream bytes and key of RC4 and WPA with 2^{32} samples. Probability of random association is $1/256 \approx 0.003906$.

Byte	Linear combinations	[18, Fig. 4.9]	Our Experiments		
			RC4	WPA (part)	WPA (full)
Z_1	$L_{1,1} = -K[0] - K[1]$	0.005304	0.005264	0.005334	0.005338
	$L_{1,2} = K[0]$	0.004367	0.004325	0.004179	0.004179
	$L_{1,3} = K[0] + K[1] + K[2] + 3$	0.005214	0.005220	0.004684	0.004633
	$L_{1,4} = K[0] + K[1] + 1$	0.004072	0.004025	0.003761	0.003760
	$L_{1,5} = K[0] - K[1] - 1$	0.004100	0.004083	0.003905	0.003905
	$L_{1,6} = K[2] + 3$	0.004461	0.004428	0.003904	0.003902
	$L_{1,7} = -K[0] - K[1] + K[2] + 3$	0.004458	0.004424	0.003903	0.003903
Z_2	$L_{2,1} = -1 - K[0] - K[1] - K[2]$	0.005316	0.005298	0.005304	0.005303
	$L_{2,2} = -K[1] - K[2] - 3$	0.005348	0.005303	0.005313	0.005314
	$L_{2,3} = K[1] + K[2] + 3$	0.005341	0.005304	0.005315	0.005315
	$L_{2,4} = K[0] + K[1] + K[2] + 3$	0.002512	0.002507	0.002505	0.002503
Z_3	$L_{3,1} = K[0] + K[1] + K[2] + 3$	0.004436	0.004401	0.004406	0.004405
Z_{256}	$L_{256,1} = -K[0]$	0.004450	0.004427	0.004430	0.004429
	$L_{256,2} = -K[1]$	–	0.003907	0.004037	0.004036
Z_{257}	$L_{257,1} = -K[0] - K[1]$	0.004115	0.004096	0.004095	0.004094

Correlation of $L_{1,4}$: Let us first point out the contrast when the correlation of Z_1 is studied with $K[0]+K[1]+1$. This is positive for RC4, but negative for WPA. In [19], it has been observed experimentally that Z_1 has a positive bias towards $K[0]+K[1]+1$. This bias has been explained in [16] by considering two paths. The first path considers the scenario when Z_1 is always equal to $K[0]+K[1]+1$, which requires the condition $K[1] = N-1$ to be satisfied. However in WPA, the most significant bit of $K[1]$ is zero, and thus $K[1]$ cannot be equal to $N-1$. So this path does not contribute to the event $Z_1 = K[0]+K[1]+1$ in WPA. For the other path, it is assumed in [16] that $\Pr(Z_1 = K[0]+K[1]+1) = 1/N$ when $K[1] \neq N-1$. The experimental results in [16] show that this value is actually $(1/N - 4/N^2)$ for RC4. However in WPA, this value is even lower, close to $(1/N - 9.5/N^2)$. Hence the contrast in the biases between RC4 and WPA.

Correlation of $L_{256,2}$: Consider the case where Z_{256} has no bias towards $(-K[1])$ in RC4, but it is biased in WPA. The reason is that $\Pr(K[1] = K[0]) = 1/4$ in WPA, and thus we can write $P(Z_{256} = -K[1]) \approx 0.25 \times P(Z_{256} = -K[0]) + 0.75 \times 1/N = 0.25 \times 0.004029 + 0.75/256 \approx 0.004036$, which matches with the experiment. This does not occur in RC4 as $\Pr(K[1] = K[0]) = 1/N$ is insignificant in that case due to independent values of $K[0]$ and $K[1]$.

3.1 Improvement in Broadcast Attack for WPA

We present a significantly improved broadcast attack against WPA over the existing works. In [6], only RC4 was studied and thus the idea of using the IV of WPA did not arise. In [1], broadcast attack on WPA has been mounted similar to that on TLS (which is almost equivalent to traditional RC4) and the correlations of the keystream bytes with the IV of WPA have not been explored at all. Exploiting the IV correlations significantly improves the recovery of the plaintext bytes $\{1, 3, 256, 257\}$ in broadcast attack on WPA.

Existing Attacks. In [6], the first byte was obtained using the conditional probability $P(Z_1 = 0|Z_2 = 0)$; thus the order of samples required would be $\Omega(N^2)$. The biases in bytes $\{3, 256, 257\}$ are of the order of $1/N^2$ over the random association probability of $1/N$; thus the order of samples required would be $\Omega(N^3)$ if one carries out the broadcast attack on RC4 as in [6]. The broadcast attack on WPA presented in [1] also considers biases to absolute values and those biases are again of the order of $1/N^2$ over the random association probability $1/N$. In this case as well, $\Omega(N^3)$ samples will be required to mount the attack. For actual broadcast attack, the constant involved in the order notation is quite high (around 2^6) in order to attain a success probability close to 1.

Our Attack. Contrary to the existing approaches, our correlations between the keystream bytes and $K[0], K[1], K[2]$ are of the order of $1/N$ over the random association probability $1/N$, and thus we require only $\Omega(N)$ samples to mount the broadcast attack in theory. It has been pointed out in [1] that the WPA

IV structure actually allows more efficient recovery of plaintext bytes in some positions than with uniform keys in RC4. However, they could only obtain practical results with 2^{24} samples or more to achieve a certain probability of success. In fact, the requirement was around 2^{30} samples for a success probability close to 1. We show that the WPA IV structure actually provides significantly better results (much lower number of samples) for certain plaintext bytes.

In the broadcast scenario, we obtain ciphertext bytes $C_r^{(u)}$ corresponding to various keystream bytes $Z_r^{(u)}$ and fixed message bytes M_r. For each user u, we substitute the $K_u[i]$ values in our list of $L_{r,q}$ to obtain Q_r many votes for $Z_r^{(u)}$. With absolute biases in RC4, the idea of [6] was to use the maximum (or a few top votes) for Z_i to obtain the target plaintext, but these votes could not be accumulated. In [1], the idea of multinomial distribution allowed the biases of Z_i to all possible absolute values to be utilized cumulatively. However, this idea does not work in our case as there is no immediate way to represent the linear combinations of the IV in the form of a probability distribution.

We do not use the votes for all Q_r relations of $Z_r^{(u)}$; we choose the votes corresponding to a few relations out of $L_{r,q}$, and merge those votes for all users. These votes in turn provide us with votes for the target plaintext byte M_r. For Z_1, we get the best result using two relations, while in other cases, we obtain the finest results using only the best biases in $L_{r,q}$. After merging the chosen votes, we consider the byte with the maximum votes as the probable plaintext byte $\hat{M}_r$. Table 3 presents our experimental results for broadcast attack. The success probability in each case is close to 1, and we have attained success in every practical experiment we performed with the claimed packet complexities.

We show the $(Z_2 = 0)$ case to illustrate that while the theoretical complexity of obtaining the byte in broadcast attack is only $\Omega(N)$, it requires 2^{14} samples to reach a success probability close to 1. Our results show significant improvements for recovering the four plaintext bytes $\{1, 3, 256, 257\}$, where the existing works require around 2^{30} samples to achieve the same success probability. It remains an open problem to utilize all biases in $L_{r,q}$ simultaneously in this attack.

Table 3. Experimental results for our plaintext recovery attack.

Byte	Event	Complexity
Z_1	$Z_1 = -K[0] - K[1]$, $Z_1 = K[0] + K[1] + K[2] + 3$	$5 \cdot 2^{13} \approx 2^{15.322}$
Z_2	$Z_2 = 0$	2^{14}
Z_3	$Z_3 = K[0] + K[1] + K[2] + 3$	2^{19}
Z_{256}	$Z_{256} = -K[0]$	2^{19}
Z_{257}	$Z_{257} = -K[0] - K[1]$	2^{21}

3.2 New Key Correlations in WPA

To strengthen the set of biases applicable towards a plaintext recovery attack against WPA, we investigated for correlations of the keystream bytes in WPA to the IV in the general linear form for the events $(Z_r = a{\cdot}K[0]+b{\cdot}K[1]+c{\cdot}K[2]+d)$. In particular, we tried with $a, b, c \in \{-1, 0, 1\}$ and $d \in \{-3, -2, -1, 0, 1, 2, 3\}$. As the first three bytes of the key, $K[0], K[1], K[2]$, are known parameters, these biases may be added to the set of known biases for Z_r, and this may potentially result in a stronger plaintext recovery attack on WPA.

In line of discussion in Sect. 2.1, one may easily note that the distribution of $K[0] \pm K[1] \pm 1$ is not uniform at all. We specifically identify three cases, as presented in Fig. 7, after an experimentation with 2^{35} samples, where we identify many biases of the order of μ/N^2 over random association ($\mu > 0.3$). Towards sharpening the broadcast attack against WPA, these biases need to be explored in more details and it would be an interesting open question how to use these biases in conjunction with the absolute biases as explained in [1].

3.3 Absolute Bias in Z_{259}

In [1,6], several new biases were identified in the first 257 bytes of RC4, and exploited in broadcast attack. In [6,23], the long term biases of RC4 were exploited to mount broadcast attack on later bytes. However, it may be interesting to find absolute biases little farther than byte 257, if they are better than using the long term biases, or if they could be used in conjunction with the long term biases. In this regard, we present a new bias at round $N+3 = 259$, described in Theorem 4. To the best of our knowledge, this is the farthest absolute bias in the initial keystream bytes of RC4 that is of the order of $O(1/N^2)$ over $1/N$.

Theorem 4. *The probability that the* $(N+3)$*-th keystream byte of RC4 is* 3 *is approximately* $\Pr(Z_{N+3} = 3) = 1/N + 0.18/N^2$.

Proof. The main path leading to the target event is as follows.

- Start with $S_0[1] = 3$ and $S_0[2] = 0$ to obtain $S_3[2] = 3$ and $S_3[3] = 0$ after the third round, with probability 1.
- Suppose that none of $j_4, \ldots, j_{N+1}$ touches the locations $\{2, 3\}$ and $j_{N+2} \neq 3$. This happens with probability $\left(1 - \frac{2}{N}\right)^{N-2}\left(1 - \frac{1}{N}\right)$, and eventually leads to $Z_{N+3} = 3$ with probability 1.

Considering the above events to be independent, the probability that the main path holds is given by $\alpha = \Pr\left(S_0[1] = 3 \wedge S_0[2] = 0\right)\left(1 - \frac{2}{N}\right)^{N-2}\left(1 - \frac{1}{N}\right)$. If it does not occur, we assume that $Z_{N+3} = 3$ holds due to random association, with probability $1/N$. Using [10, Theorem 6.2.1] for $\Pr\left(S_0[1] = 3 \wedge S_0[2] = 0\right)$, we compute $\Pr(Z_{N+3} = 3) \approx \alpha + (1 - \alpha) \cdot (1/N) \approx 1/N + 0.18/N^2$. □

Experiments with 2^{33} random keys show that $\Pr(Z_{N+3} = 3) = 0.003909$, both in the case of RC4 and WPA; thus conforming to the theoretical value.

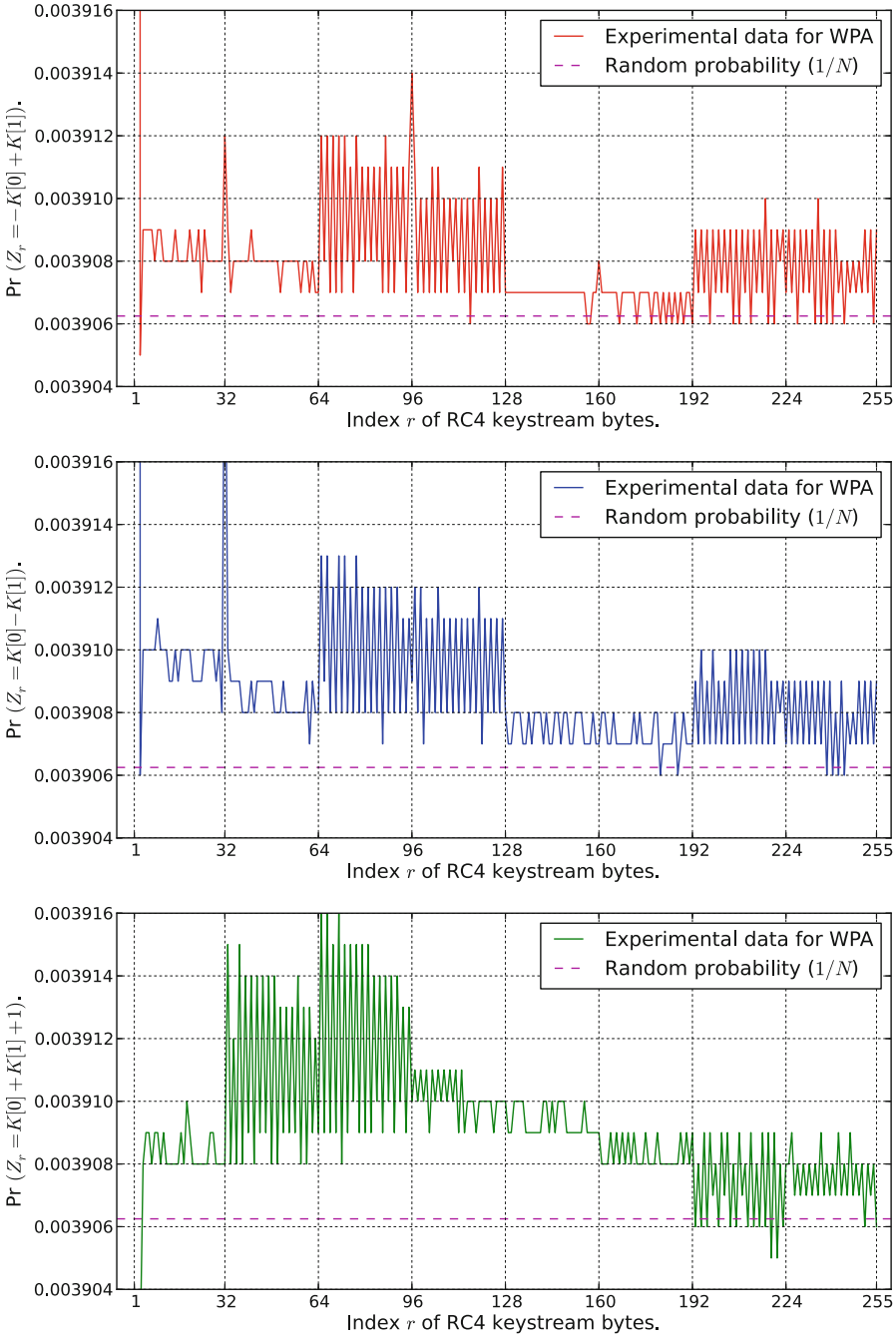

Fig. 7. Linear correlations between the IV and the initial keystream bytes of WPA.

4 Conclusion

In this paper, we present various non-randomness results on RC4 when used in the WPA protocol. We analyze several biases of RC4 and also note how they evolve in WPA as the initial three key bytes are derived from the IV. We prove the interesting sawtooth distribution of the first byte and the similar nature for the biases in $(Z_r = 0)$, as pointed out in [1]. We also improve the theoretical estimate for the $(Z_r = r)$ bias of RC4 to obtain better results than [6].

In another direction, we revisit the correlation of certain keystream bytes to the first three IV bytes in WPA and we notice that they provide much higher biases than what had been presented in [1]. This improves the broadcast attack on WPA significantly towards obtaining certain plaintext bytes. Our combinatorial results complement the existing literature in understanding the reason of some interesting empirical biases in WPA, as well as in adding some new observations and biases in the scenario of broadcast attack against WPA.

Acknowledgments. We are thankful to the anonymous reviewers of FSE 2014 for their detailed review reports containing invaluable feedback, which helped in substantially improving the technical and editorial quality of our paper.

References

1. Alfardan, N., Bernstein, D. J., Paterson, K. G., Poettering, B., Schuldt, J.: On the security of RC4 in TLS. In: USENIX Security Symposium Presented at FSE 2013 as an Invited Talk [2] by Daniel J. Bernstein (2013). http://www.isg.rhul.ac.uk/tls/
2. Bernstein, D. J.: Failures of secret-key cryptography. Invited talk at FSE 2013; session chaired by Bart Preneel (2013)
3. Chaabouni, R.: Break WEP faster with statistical analysis. IACR Cryptology ePrint Arch. **2013**, 425 (2013). http://eprint.iacr.org/2013/425
4. Fluhrer, S.R., Mantin, I., Shamir, A.: Weaknesses in the key scheduling algorithm of RC4. In: Vaudenay, S., Youssef, A.M. (eds.) SAC 2001. LNCS, vol. 2259, p. 1. Springer, Heidelberg (2001)
5. IEEE Computer Society. 802.11iTM - IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks -Specific requirements, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Amendment 6: Medium Access Control (MAC) Security Enhancements, July 2004
6. Isobe, T., Ohigashi, T., Watanabe, Y., Morii, M.: Full plaintext recovery attack on broadcast RC4. In: Moriai, S. (ed.) FSE 2013. LNCS, vol. 8424, pp. 179–202. Springer, Heidelberg (2014)
7. Klein, A.: Attacks on the RC4 stream cipher. Des. Codes Crypt. **48**(3), 269–286 (2008). Published online in 2006
8. Maitra, S., Paul, G.: New form of permutation bias and secret key leakage in keystream bytes of RC4. In: Nyberg, K. (ed.) FSE 2008. LNCS, vol. 5086, pp. 253–269. Springer, Heidelberg (2008)
9. Maitra, S., Paul, G., Sen Gupta, S.: Attack on broadcast RC4 revisited. In: Joux, A. (ed.) FSE 2011. LNCS, vol. 6733, pp. 199–217. Springer, Heidelberg (2011)

10. Mantin, I.: Analysis of the stream cipher RC4. Master's thesis, The Weizmann Institute of Science, Israel (2001). http://www.wisdom.weizmann.ac.il/itsik/RC4/RC4.html
11. Mantin, I., Shamir, A.: A Practical attack on broadcast RC4. In: Matsui, M. (ed.) FSE 2001. LNCS, vol. 2355, pp. 152–164. Springer, Heidelberg (2002)
12. Paterson, K.G., Schuldt, J.C.N., Poettering, B.: Plaintext recovery attacks against WPA/TKIP. In: Cid, C., Rechberger, C. (eds.) FSE 2014, LNCS 8540, pp. 325–349 (2015)
13. Paul, G., Maitra, S.: Permutation after RC4 key scheduling reveals the secret key. In: Adams, C., Miri, A., Wiener, M. (eds.) SAC 2007. LNCS, vol. 4876, pp. 360–377. Springer, Heidelberg (2007)
14. Paul, G., Rathi, S., Maitra, S.: On non-negligible bias of the first output byte of RC4 towards the first three bytes of the secret key. Des. Codes Crypt. **49**(1–3), 123–134 (2008). Initial version in Proceedings of WCC 2007
15. Roos, A.: A class of weak keys in the RC4 stream cipher. Two posts in sci.crypt, 43u1eh$1j3@hermes.is.co.za and 44ebge$llf@hermes.is.co.za (1995). http://www.impic.org/papers/WeakKeys-report.pdf
16. Sarkar, S.: Proving empirical key-correlations in RC4. Inf. Proc. Lett. **114**(5), 234–238 (2014)
17. Sen Gupta, S., Maitra, S., Paul, G., Sarkar, S.: (Non-)random sequences from (non-)random permutations - analysis of RC4 stream cipher. J. Crypt. **27**, 67–108 (2014). doi:10.1007/s00145-012-9138-1. Published online in December 2012
18. Sepehrdad, P.: Statistical and algebraic cryptanalysis of lightweight and ultra-lightweight symmetric primitives. Ph.D. thesis No. 5415, École Polytechnique Fédérale de Lausanne (EPFL) (2012). http://lasecwww.epfl.ch/sepehrdad/Pouyan_Sepehrdad_PhD_Thesis.pdf
19. Sepehrdad, P., Vaudenay, S., Vuagnoux, M.: Discovery and exploitation of new biases in RC4. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) SAC 2010. LNCS, vol. 6544, pp. 74–91. Springer, Heidelberg (2011)
20. Sepehrdad, P., Vaudenay, S., Vuagnoux, M.: Statistical attack on RC4. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 343–363. Springer, Heidelberg (2011)
21. Tews, E.: Attacks on the WEP protocol. IACR Cryptology ePrint Arch. **2007**, 471 (2007). http://eprint.iacr.org/2007/471
22. Tews, E., Weinmann, R.-P., Pyshkin, A.: Breaking 104 bit WEP in less than 60 seconds. In: Kim, S., Yung, M., Lee, H.-W. (eds.) WISA 2007. LNCS, vol. 4867, pp. 188–202. Springer, Heidelberg (2008)
23. Ohigashi, T., Isobe, T., Watanabe, Y., Morii, M.: How to recover any byte of plaintext on RC4. In: Lange, T., Lauter, K., Lisoněk, P. (eds.) SAC 2013. LNCS, vol. 8282, pp. 155–173. Springer, Heidelberg (2014)
24. Vaudenay, S., Vuagnoux, M.: Passive–only key recovery attacks on RC4. In: Adams, C., Miri, A., Wiener, M. (eds.) SAC 2007. LNCS, vol. 4876, pp. 344–359. Springer, Heidelberg (2007)