I. J. Computer Network and Information Security, **2015, 7, 37-45**
Published Online June 2015 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2015.07.05

A Survey on RC4 Stream Cipher

Poonam Jindal
National Institute of Technology, Kurukshetra, 136119, Haryana
Email: poonamjindal81@nitkkr.ac.in

Brahmjit Singh
National Institute of Technology, Kurukshetra, 136119, Haryana
Email: brahmjit@nitkkr.ac.in

Abstract—RC4 is one of the most widely used stream cipher due to its simplicity, speed and efficiency. In this paper we have presented a chronological survey of RC4 stream cipher demonstrating its weaknesses followed by the various RC4 enhancements from the literature. From the recently observed cryptanalytic attempts on RC4 it is established that innovative research efforts are required to develop secure RC4 algorithm, which can remove the weaknesses of RC4, such as biased bytes, key collisions, and key recovery attacks specifically on WEP and WPA. These flaws in RC4 are offering open challenge for developers. Hence our chronological survey corroborates the fact that even though researchers are working on RC4 stream cipher since last two decades, it still offers a plethora of research issues related to statistical weaknesses in either state or keystream.

Index Terms—Security attacks, Symmetric key encryption, Stream cipher, RC4, Weaknesses of RC4

I. Introduction

The concept of security is generally interpreted as the idea of confidentiality of data being transmitted, particularly the digital information transmitted over the wireless network. Most commonly security is provided using cryptographic primitives. As shown in Fig. 1 the cryptographic primitives are classified into three main categories; not using key, symmetric key and asymmetric key [1]. Although Fig. 1 is not presenting an exhaustive list of these primitives but is highlighting the important and relevant areas. In this paper we have focused on symmetric key ciphers which are also known as secret key or single key ciphers. Secret key ciphers are further classified as block ciphers and stream ciphers. In block ciphers, a block of bits/bytes is processed at a time. DES, IDEA, RC5, AES, BLOWFISH, TWOFISH are the different available block ciphers. Whereas in stream ciphers one bit or a byte of data is processed at a time. Stream ciphers are further classified as synchronous and self-synchronous stream ciphers. Synchronous stream ciphers (SSC) are prominently discussed in literature. However, generally due to the design problems, self-synchronizing stream cipher (SSSC) are not much explored in literature and are less used in practice [2]. Different synchronous stream ciphers available in the literature are RC4, E0 (a stream cipher used in Bluetooth), A5/1 and A5/2 (stream ciphers used in GSM), SNOW 3G, ZUC (4G stream ciphers), Rabbit, FISH, and HC-256 etc. [3-8].

In this paper we have considered stream ciphers. A keystream is produced in stream ciphers which is a pseudorandom sequence of bits. A plaintext (a sequence of bits/bytes) is converted into ciphertext (again a sequence of bits/bytes of same length as that of plaintext) by hiding the plaintext with a generated keystream, using a simple XOR operation. The strength of stream ciphers is a random keystream which ensures the computational security of the cipher. In cryptographic primitives non-random events which can be computationally recognized either in the internal states and in the output keystream are generally not desirable. Thus the cryptanalysis of stream ciphers is imperatively focused on the identification of non-random events and hence extensive analysis of stream ciphers is done till date to identify the occurrence of non-random events. Table 1 and Figs. (2, 3) demonstrates the overview of various cryptanalytic attack models, modes of attacks and goals of intruder in stream ciphers respectively. The general classification of the cryptanalytic attacks on stream ciphers with the assumption that what is known to the intruder is shown in Table. 1. These cryptanalytic attacks are also known as attack models. Further on the basis of these attack models and the knowledge of intruder (what is known to intruder), Fig. 2 presents the different modes in which the intruder can attack the cipher. Intruder mount these models and modes of attack on stream ciphers with the goals as shown in Fig. 3 [1-2].

A chronological comprehensive survey of the most prevalent and commercially used RC4 stream cipher along with the countermeasures is presented in this paper. We have focused on RC4 because it outperforms amongst all the modern stream ciphers. Though the algorithm is publicly revealed in 1994 through internet but due to its design simplicity everyone gets attracted towards it and has been adopted worldwide [9]. The cipher is widely adopted in various software and web applications. It is used in different network protocols such as WEP (Wireless equivalent privacy), WPA (Wi-Fi protected access), and SSL (Secure socket layer). Also it is extensively used in Microsoft windows, Apple OCE (Apple Open Collaboration Environment), secure SQL (a server for database management and data warehousing

solution) etc. Throughout the paper we have tried to explore the various weaknesses of the cipher till date. It is found that regardless of many efforts made by researchers in improving the flaws of RC4 cipher, still there are number of biases exist in the keystream, key recovery can be made from state and certain sets of keys do exist that can generate similar states. It corroborate the fact that even after the decades of research the RC4 stream cipher continues to offer research problems of interest to researchers.

Rest of the paper is organized as follows. Section 2 gives the brief description of RC4 encryption algorithm. Different weaknesses and their related cryptanalysis is presented in section 3. Section 4 describes existing proposals for the enhancement of the cipher are given in section 5. Conclusion and future scope is drawn in section 6.

II. RC4 Description

RC4 follows the design strategy used in stream ciphers. To extract the pseudorandom data bytes from a pseudorandom permutation is the basic design principle of RC4 stream cipher. RC4 has two working modules: first there is a KSA with key K as input (with typical size of 40-256 bits), and second is PRGA which generates a pseudo-random output sequence. The pseudo code for RC4. Fig 4 presents the complete working of RC4 encryption algorithm. KSA generates the 256 byte initial state vector S, by scrambling input state vector with a random key *K*. The S contains a permutation of 8 bit words i.e. 256 bytes. The secret key *k* is generally of length between 8 to 2048 bits and the expanded key K (K of length N=256 bytes) is produced by performing simple repetitions. The expanded key is generated in the manner such that if secret key *k* is of length *l* bytes, the expanded key will be K[i] = *k* [*i* mod *l*] for $0 \leq i \leq N-1$. Further *S* pairs are swapped and an initial state S_{N-1} is achieved at the end which is the input to the second module PRGA. It generates the keystream of words and is further XORed with the plaintext to produce a ciphertext. To figure axis labels, use words rather than symbols. Do not label axes only with units. Do not label axes with a ratio of quantities and units. Figure labels should be legible, about 9-point type. It is to be noted that each time a new keystream byte *O* is required, RC4 runs the loop of PRGA and each time with the generation of new keystream the internal state *S* is updated.

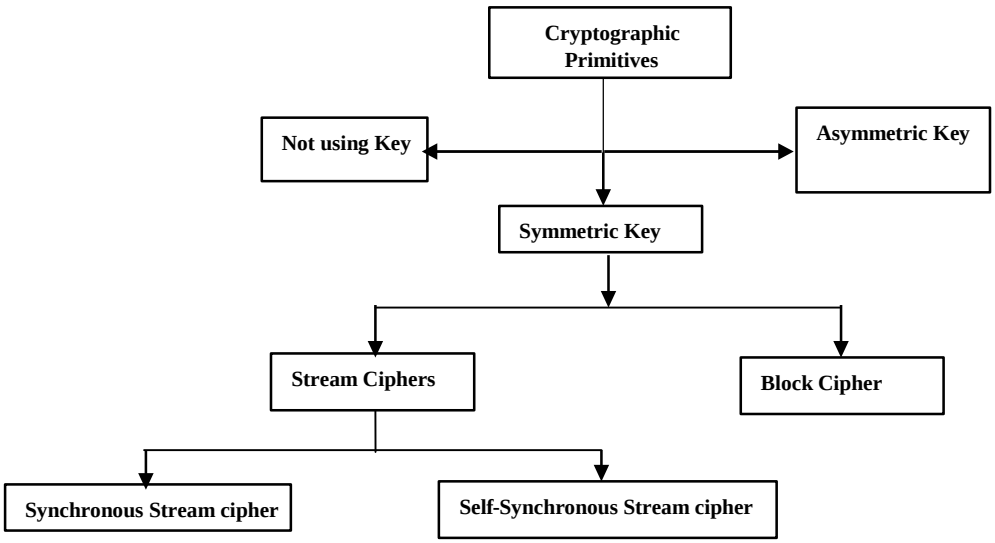

Fig. 1. Cryptographic primitives

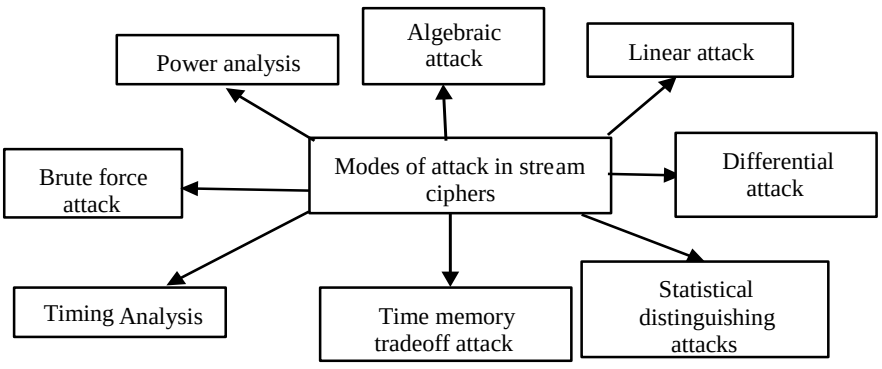

Fig. 2. Modes of attack in stream ciphers

Table 1. Classification of the Cryptanalytic attacks on Stream ciphers (RC4)

Type of Cryptanalytic Attacks	Information known to cryptanalytic
Ciphertext only	Intruder has partial knowledge of some ciphertext (CT) messages but does not know anything about plaintext message (PT)
Known plaintext	Intruder has some knowledge of the PT-CT pairs
Chosen plaintext	Intruder knows the encryption algorithm that produces CT for the PT messages chosen by intruder using a secret key
Known initialization vector (IV)	Intruder either has some knowledge of IV or choose some IV and obtains the corresponding output keystream with the secret key. This is also known as resynchronization attack and follows known plaintext attack for obtaining keystream and CT.
Chosen ciphertext	Intruder knows the encryption algorithm that produces PT for the CT messages chosen by intruder using a secret key

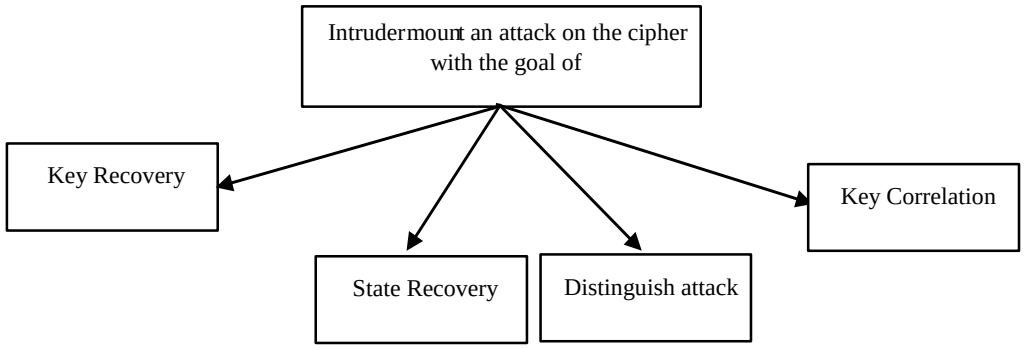

Fig. 3 Goal of Intruder

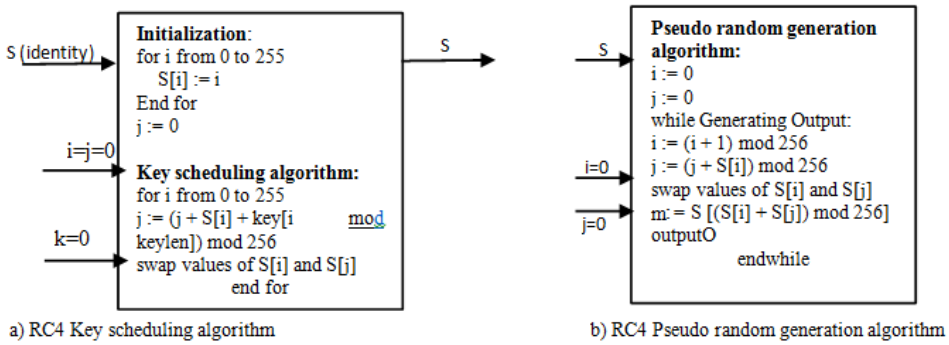

Fig. 4. RC4 Stream Cipher

III. Existing Weaknesses of RC4 and the Related Cryptanalysis

RC4 is known to be one of the simplest and widely adopted cipher. However the simplicity of RC4 makes it vulnerable to different security attacks. The cipher was designed in 1984 and was anonymously released on mails and news groups in 1994. Since then many cryptanalyst have exploited the weaknesses of the cipher for having access on either input state or key. From the basic structure of RC4 it is observed that PRGA generates a pseudorandom output sequence (bytes) from the permuted internal state which itself is a random sequence. The cryptanalyst is always in search of the statistical weaknesses of the output sequence. Statistical weaknesses are the biases in the random keystream that can be exploited with a very high probability of success, to differentiate the generated RC4 keystream from a truly random sequence of bytes. Hence the main goal of an intruder while attacking RC4 is to investigate the non-random behavior either in the internal state or in the output keystream. The brief summary of security attacks on RC4 since its first public appearance to date is shown in Table 2. Various weaknesses of RC4 algorithm which are the roots to several attacks are detailed as below:

A. Weak Keys

Weak keys are the small set of keys in RC4 which leaves some traces in the keystream generated after KSA or in the output bytes after PRGA. If such traces are followed by the intruder he/she can easily recover the key from the internal state or the output stream. In 1995, the first attack

first attack was made by Roos after discovering first set of weak keys in the RC4 algorithm. He found experimentally that there is significant probability for the first byte produced by RC4 to be O_1=K [2] + 3 for a given key length *K [0]….K[l]*, where *K [0]+K[1]=0*. He found that this event occurs with the probability ranges from 0.12 to 0.16 [10]. These results were later proved theoretically by authors in [11]. Roos also extended his research for first biased output byte to first two output bytes. Some more observations regarding the weak keys were given by Wagner in [12].

B. Key Collisions

In RC4 KSA, it may be possible to generate a similar state even if two different keys are used and hence a similar output keystream will be produced. Such a scenario is known as key collision or related key pairs. Construction of such key pairs is the goal of attacker. Such key pairs were constructed by Grosul and Wallach in [13]. They constructed the key pair in a manner such that the second key of the pair is made by simply making two complementary modifications in the first key with an objective of not disturbing the state update process. A similar cryptanalysis was made by authors in [14], where key collision was obtained by varying the key bytes in two places such that during KSA the effect of one modification is nullified by the other. In 2009, a more practical way of constructing colliding key pairs by making the modification only in one key byte instead of two of RC4 stream cipher is given in [15]. The author also reported a 20 byte colliding key pairs for which the generated sate after KSA changed only in two places. Similarly a 22 byte key collision was obtained in [16]. Recently in 2013 authors have proposed certain ways to construct colliding key pairs such that the states produced after KSA vary only in few bytes [17].

C. Key Recovery From State

RC4 PRGA is reversible in nature. From any given state of PRGA it is easy to reach the internal state and it is quite easy to recover the secret key from the internal state. If one could be able to efficiently reverse the KSA and obtain the secret key, it becomes possible to convert state recovery attack to key recovery attack. This weakness of RC4 was remained unexplored till the year 2007, when key recovery was done by solving modular equations for the first time by Paul and Maitra in [18]. As revealed by Roos in [10] that key bytes and the PRGA state bytes are correlated, the work presented in [18] was motivated from the same observation. The idea behind the work presented by Paul and Maitra is to select suitable equations, with known values from S, and solve those methodically for the key bytes. An improved key recovery approach with high probability of success as compared to [18] is achieved by Biham and Carmeli in [19], where authors have used differential equations instead of basic modular equations. The key recovery from state was further improved by authors in [20] by using equation solving approach. The technique discussed by authors in [20] is known to be the more efficient and faster approach form key recovery from state. Another key recovery approach was discussed in [21] which is again an improvement over the technique given by [19]. Authors have used the same differential equation as in [19], but key was recovered using bit by bit approach. A new bidirectional search algorithm for key recovery given in [22], is a faster and efficient as well.

D. Key Recovery From Keystream

Key can be easily recovered from output keystream and this weakness of RC4 was exploited when used in WEP (Wired equivalent privacy) and WPA (Wi-Fi protected access). RC4 based WEP was the first security protocol used for Wi-Fi security in IEEE 802.11 LANs and thus always remained a target for cryptanalysis. As reported in the literature the adversary attack the WEP protocol by recovering the secret key K from the known values of IV (initialization vector) and known values of the RC4 keystream bytes found from the plaintext and ciphertext pairs. The number of attacks on WEP reported in the literature are Fluhrer, Mantin and Shamir attack (FMS) [23], Korek practical attacks [24-25], Mantin attack on RC4 [26] and WEP, Klien attack [27], Tews, Weinmann and Pyshkin (TWP) attack [28], Vaudenay and Vuagnoux (VV) attacks [29], Tews and Beck (TB) attack [30], Shepehrdad, Vaudenay and Vuagnoux (SVV) attack [31-33], and Shepehrdad, Susil, Vaudenay and Vuagnoux (SSVV) attack [34], WEP was declared as an insecure protocol. Later it is replaced by WPA (Wi-Fi protected access) which also make use of RC4 as its core element. WPA defended against many attacks in WEP. WPA has again proved to be a weak protocol due to TB data injection attacks [30], and SVV attacks [33]. Further a new protocol WPA2 was proposed by the Wi-Fi alliance which uses AES block cipher as an encryption algorithm instead of RC4. Though WPA2 is a secure protocol, removing many weaknesses of WEP and WPA but it's hardware based applications are not cost effective as compare to WEP and WPA where RC4 was used as a basic module. Inspite of so many attacks and weaknesses in WEP, it is enormously opted in large number of applications due to its simplicity over WPA and WPA2. Different WEP attacks on the basis of packet complexity are summarized in Table 3.

E. State Recovery

The state-space size in RC4 is $N! \times N^2$, where N! = 0 is the space of N bytes in the internal state S and N^2 comes from the all possible combinations of indices *i and j*. Hence in RC4, for N=256 the total state-space available is, $256! \times 265^2 \approx 2^{1700}$. Inspite of such a big state-space, the state recovery is possible in the cipher. The first state recovery attack on RC4 was proposed Knudsen, Meier and Preneel [35] in 1998 where the attack complexity was found to be 2^{779} for N=256. In the same year 1998, another state recovery attack was analyzed using some cycle –structures of RC4 [36] and observed that for N=32, state recovery can be done in 2^{42} steps. A probabilistic approach was used for RC4 state recovery in [37], having attack complexity similar to the one obtained in [35]. A

slightly different attempt to recover state of RC4 on the basis of the partial knowledge of state is proposed in [38]. Initially they presented that the attack complexity given in [35] would be 2^{220} with the knowledge of 112 known states and further it was proposed that similar search complexity can be obtained with the knowledge of only 73 state values. Again the attack complexity of [35] was further reduced to 2^{731} in [39]. In [40] a new state recovery approach based on the use of generative patterns, revealing the value of j in consecutive rounds, was discussed. The authors have claimed the search complexity to be reduced from 2^{731} to 2^{241}. The improved attack based on guess and determine policy proposed in [41] claimed the data complexity to be reduced further to 2^{211}.

F. Biased Bytes

In stream ciphers the event or bytes are said to be biased if an event occurs with different probability as that from the uniformly random sequence of bits/bytes. To study the non-random behavior of bytes is the goal of attacker. Several biases or correlation related to secret key, state variables, and short term and long term biases related to keystream bytes are in RC4 KSA and PRGA are available in literature.

- The first bias related to secret key was experimentally observed in [10], named as Roos key correlation, where the correlation between of the secret key bytes and the initial state bytes. Several key length dependent biases with their partial and conclusive proofs are presented in [42, 43] and [44] respectively. The first statement on the any bias in the first byte based on key length of RC4 is given by authors in [45].
- Some biases do exist that relate bytes in the output keystream to the internal state and make the state recovery attack possible. The relation between output keystream and initial state was first recognized in [46] but without proving it and later the bias is proved in [47, 48]. The results presented in [48, 49] proved that the internal state S_0 after KSA for the very first time is highly non-uniform. Mantin's correlation in S_0 distribution is known to be the most beneficial internal bias of RC4 to date and give rise to more non-uniformities in the output keystream. Later in 2010 [32] to find all the linear correlations in a single round of the algorithm the authors performed an exhaustive search of all relations between states and indices (i_r, j_r, $S_r[i_r]$, $S_r[j_r]$, Z_r, where r represents the round) on the space. Several new biases in RC4 relating the internal state S to the output keystream were obtained. The work presented in [32] was further extended in [50] with the goal to mount an attack on WEP and WPA by using these obtained biases.
- Another area of analysis is the identification of short term and long term biases in the keystream bytes. The keystream biases that do not linger on to the future rounds are named as short term biases. The first and the second keystream byte bias was analyzed in [43, 51] and [47, 52] respectively. Authors in [47] claimed that except for Z_2, there exist no significant initial keystream byte bias towards zero in RC4. But in [43, 53] it was proved that all the initial bytes ranging from Z_3 to Z_{255} are significantly biased towards zero. Several recent short term biases are discussed and proved in [44, 51, 53, 54]. Due to the existence of number of short term biases it was suggested by many researchers to discard the initial N to 6N bytes of the output keystream. The bytes generated thereafter are to be used for encryption. The remained keystream biases even after the removal of initial bytes are termed as long term biases. Different studies related to the long term biases in the keystream are discussed in [51, 55, 56].

The available literature reveals that although there had been many successful security breaches in the RC4, but the striking combination of robustness and design elegance of RC4 has made it most preferred cipher for last two decades.

Different researchers have proposed variety of its implementations to make the cipher more secure (discussed in section 4), But the available literature demonstrate the insecurity of RC4 till date. The most recent weaknesses observed in the year 2013 and 2014 on RC4 and its applications in WEP, WPA and TLS reveals the fact that the RC4 is not secure till date and is still an attraction for community.

IV. Enhancements In RC4 Stream Cipher

Due to the RC4 weaknesses and related cryptanalytic attempts as discussed in section 3, many variants of RC4 have been proposed by researchers. We have reported several enhancements of RC4 algorithm. A modified 32-bit RC4, named as *RC4 (n, m)* keystream generator, with good randomness and uniform distribution was proposed in [58]. The authors have claimed the resistance of cipher against all the attacks that were successful on conventional RC4. In [59] authors have developed two attacks against *RC4 (n, m)* on the basis of non-randomness of internal states. In [60] authors have studied theoretically the RC4 KSA. It is found that the expected number of times each value of the state permutation is moved by the indices *i, j* is not uniform and proposed a modified RC4 KSA+ and PRGA+ with three layers of scrambling. Analysis of RC4+ illustrates that although the modified algorithm destroys the correlation between the state and the key but the running time of KSA+ is approximately 2.94 times than that of original RC4 KSA and the running time of one round of our PRGA+ is 1.70 times than that of original RC4 PRGA. Recently in 2013 [61] authors have successfully mounted a distinguishing attack on RC4+. To increase the security of RC4, a new PRGA, based on conventional RC4 is proposed in [62, 63]. It is revealed that the

proposed proposed RC4 has two internal states and has removed some of the byte biases which are the foundation of many security attacks on RC4 and is also faster than the existing conventional RC4.

Table 2. Cryptanalysis on RC4 stream cipher

Year	Weak keys and key recovery from state	Key recovery from key stream	State recovery attack	Biases and Distinguishers
1995	-Roos weak keys[10] -Wagner weak keys [12]	-	-	-Roos biases [10]
1996	-	-	-	-Glimpse bias [46]
1997	-	-	-	-Golic long term bias [53]
1998	-	-	- KMP branch and bound approach [35]	
2000	-Related key-pairs[13]	-	-Iterative probabilistic cryptanalysis [36-37]	-Digraph biases [55]
2001	-	FMS WEP attack.[23]	-	Broadcast attack [47]
2002	-	-	-	-
2003	-	-	State part known [38]	
2004	-	Korek WEP attack [24-25]	-	
2005	-	Mantin WEP attack [26]	-	
2006	-	Klein WEP attack [27]	-	-
2007	- short related keys attack [14]	-TWP WEP attack [28] -VV WEP attack [29]	Hill climb search attack [39]	
2008	-Difference equations[19] -bit by bit approach attack [21]	-	-generative pattern[40] -iterative probabilistic attack [41]	Maitra and Paul conditional Bias [56]
2009	-key collision attacks[15] -bidirectional search attacks	-TB WEP and WPA attacks [30]	-	-
2010		SVV WEP attack [31]	-	SVV biases in key and state variables [32]
2011	-New key collisions [16]	SVV WEP and WPA attack [32]	-	-keylength biases [42]
2012		SVV WEP and WPA attack [33]	-	
2013	-Near colliding keys [17]	SSVV passive attack on WEP [34]	-	-TLS and WPA attack [45,52] -Full Broadcast attack [44] - TLS related bias [45]
2014	-	-	-	-biased bytes [43]

Table 3. Summary of WEP attacks

Year	WEP Attack	Type	Packets required to recover WEP secret key
2001	FMS attack [23] FMS attack [57]	Passive attack and Theoretically estimated Passive attack and Proved practically	4 Lacs 5.5 Lacs
2004	Korek attack [24]	Practical attack with Aircrack-ng	1 lac
2006	Klien attack [27] Klien attack [32,33]	Passive attack and Theoretically estimated Passive attack and Proved practically	25000 60000
2007	TWP attack [28]	Passive attack and implemented Practically attack	10000
2007	VV attack [29]	Passive attack and implemented Practically attack	32700
2009	TB attack [30] TB attack [32,33]	Interactive with Aircrack-ng Non-interactive (proved) with Aircrack-ng	24200 30000
2010	SVV attack [31]	Passive attack and Theoretically estimated	9800
2011	SVV attack [32]	Passive attack and Theoretically estimated	4000
2013	SSVV attack [33]	Passive attack and Proved practically	27500
2013	SSVV attack [34]	Non-interactive (proved) with Aircrack-ng	22500
2013	SSVV attack [34]	Interactive with Aircrack-ng	19800

In [64] authors have proposed a new variant of RC4 called Quad-RC4 without changing the basic structure of conventional RC4. The proposed RC4 structure promises the reasonable security and a high throughput. In term of speed the proposed cipher performs much better in comparison with HC-128, the fastest software stream cipher amongst the e-STREAM finalists. A new variant of RC4 known as FJ-RC4 is proposed by authors in [65]. In FJ-RC4 is designed in a manner such that in KSA input key is divided into three parts and the structure of PRGA is same as with conventional RC4. A new keystream after KSA is generated in three rounds whereas PRGA performs only single round. Another variant of RC4 known as effective RC4 cipher is proposed in [66] where the security analysis is performed by using Shannon's Secrecy theory and numerical values are obtained to analyze the secrecy. It is proposed that the improved RC4 cipher can be used in software applications where there is requirement of both the throughput and secrecy. Further a new PRGA RC4B is

proposed in [67], which provides better immunity against the known attacks. The new variant of RC4 is proposed in [68] which provides high security along with long period of KSA keystream, large complexity and having good statistical properties.

From the available literature it is found that many recent RC4 variants have been proposed by researchers. Some are targeted towards achieving better security by removing the non-uniformity of bytes or by removing the correlation between key and the state bytes and some towards better performance in terms of time or throughput. Some of the proposals have entirely changed the basic structure of RC4 which is generally not desirable because the robust design of RC4 is the basic strength of the cipher. However, inspite of so many proposals on RC4, many open issues exists on RC4 till date and are mentioned below:

- RC4 keystream key collisions
- Key recovery attacks on WPA
- Keylength dependent anomalies in RC4.
- State recovery attacks on RC4
- Searches of more biases,

Therefore there is a strong need of the enhanced RC4 algorithm. It is recommended that while retaining the basic structure of RC4, one can design a new enhanced RC4 stream cipher exhibiting a sufficient resistance against the existing weaknesses of the cipher.

V. Conclusion

In this paper, a chronological survey of the cryptanalysis on RC4 was presented beginning with its first public appearance to date. We have identified and presented the various weaknesses of RC4 cipher followed by the measure taken by various researchers to improve the security of the cipher. It is found that though extensively deployed in the field, there are still certain flaws in its security. Although many improved variants of RC4 which removes the existing weaknesses and enhance the security of the cipher may be found in the literature, but the question about which is the best solution still remain unanswered, since each of them focus on specific attack or weakness. Further in spite of all the developments reported in the literature, there are still many open research challenges and issues related to searches of more biases, key collisions in keystream, and key recovery attack on WPA. Therefore it is concluded that there is ample scope to further investigate the issues in RC4 particularly the non-random behavior of bytes in the state permutation, and to develop a new, more efficient and effective RC4 encryption algorithm.

References

[1] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. Hand- book of Applied Cryptography. CRC Press, August 2011 edition, 1996. Fifth Printing.

[2] Douglas R. Stinson. *Cryptography: Theory and Practice*. CRC Press, third November 2005) edition, 1995.

[3] Alex Biryukov, Adi Shamir, and David Wagner. Real time cryptanalysis of A5/1 on a PC. In Bruce Schneier, editor, *FSE*, volume 1978 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2000.

[4] Bluetooth T M. Bluetooth specification, v4.0, June 2010. E0 encryption algorithm described in volume 2, pages 1072–1081. Available online at http://www.bluetooth.org.

[5] Marc Briceno, Ian Goldberg, and David Wagner. A pedagogical implementation of the GSM A5/1 and A5/2 "voice privacy" encryption algorithms. Available online at http://www.scard.org/gsm/a51.html, 1998.

[6] 3rd Generation Partnership Project. Specification of the 3GPP confidentiality and integrity algorithms UEA2 & UIA2. ETSI/SAGE Specification Document 2: SNOW 3G Specification, v1.1, September 6, 2006.

[7] ECRYPT Stream Cipher Project eSTREAM. The current eSTREAM portfolio. Available online at http://www.ecrypt.eu.org/stream/ index.html.

[8] ECRYPT Stream Cipher Project eSTREAM. Software performance results from the eSTREAM project. Available online at http://www. ecrypt.eu.org/stream/perf/#results.

[9] Ronald L. Rivest. RSA security response to weaknesses in key scheduling algorithm of RC4. Technical note, RSA Data Security, Inc., 2001.

[10] Andrew Roos. A class of weak keys in the RC4 stream cipher. Two posts in sci.crypt, message-id 43u1eh$1j3@hermes.is.co.za 1995.

[11] Goutam Paul, Siddheshwar Rathi, and Subhamoy Maitra. On non-negligible bias of the first output byte of RC4 towards the first three bytes of the secret key. *Des. Codes Cryptography*, 49(1-3):123–134, 2008. Initial version in proceedings of WCC 2007.

[12] David A. Wagner. My RC4 weak keys. Post in sci.crypt, messageid 447o1l$cbj@cnn.Princeton.EDU, 1995. Available online at http://www.cs.berkeley.edu/~daw/my-posts/my-rc4-weak-keys.

[13] Alexander L. Grosul and Dan S. Wallach. A related-key cryptanalysis of RC4. Technical Report TR-00-358, Department of Computer Science, Rice University, 2000.

[14] Eli Biham and Orr Dunkelman. Differential cryptanalysis in stream ciphers. *IACR Cryptology ePrint Archive*, 2007:218, 2007.

[15] Mitsuru Matsui. Key collisions of the RC4 stream cipher. In Orr Dunkelman, editor, *FSE*, volume 5665 of *Lecture Notes in Computer Science*, pages 38–50. Springer, 2009.

[16] Jiageng Chen and Atsuko Miyaji. How to find short RC4 colliding key pairs. In Xuejia Lai, Jianying Zhou, and Hui Li, editors, *ISC*, volume 7001 of *Lecture Notes in Computer Science*, pages 32–46. Springer, 2011.

[17] Subhamoy Maitra, Goutam Paul, Santanu Sarkar, Michael Lehmann, and Willi Meier. New results on generalization of Roostype biases and related keystreams of RC4. In Amr Youssef, Abderrahmane Nitaj, and Aboul Ella Hassanien, editors, *AFRICACRYPT*, volume 7918 of *Lecture Notes in Computer Science*, pages 222–239. Springer, 2013.

[18] Goutam Paul and Subhamoy Maitra. Permutation after RC4 key scheduling reveals the secret key. In Carlisle M. Adams, Ali Miri, and BIBLIOGRAPHY Michael J. Wiener, editors, *Selected Areas in Cryptography*, volume 4876 of *Lecture Notes in Computer Science*, pages 360–377. Springer, 2007.

[19] Eli Biham and Yaniv Carmeli. Efficient reconstruction of RC4 keys from internal states. In Kaisa Nyberg, editor, *FSE*, volume 5086 of *Lecture Notes in Computer Science*, pages 270–288. Springer, 2008.

[20] Mete Akgün, Pinar Kavak, and Hüseyin Demirci. New results on the key scheduling algorithm of RC4. In Dipanwita Roy Chowdhury, Vincent Rijmen, and Abhijit Das, editors, *INDOCRYPT*, volume 5365 of *Lecture Notes in Computer Science*, pages 40–52. Springer, 2008.

[21] Shahram Khazaei and Willi Meier. On reconstruction of RC4 keys from internal states. In Jacques Calmet, Willi Geiselmann, and Jörn Müller-Quade, editors, *MMICS*, volume 5393 of *Lecture Notes in Computer Science*, pages 179–189. Springer, 2008.

[22] Riddhipratim Basu, Subhamoy Maitra, Goutam Paul, and Tanmoy Talukdar. On some sequences of the secret pseudo-random index *j* in RC4 key scheduling. In Maria Bras-Amorós and Tom Høholdt, editors, *AAECC*, volume 5527 of *Lecture Notes in Computer Science*, pages 137–148. Springer, 2009.

[23] Scott R. Fluhrer, Itsik Mantin, and Adi Shamir. Weaknesses in the key scheduling algorithm of RC4. In Serge Vaudenay and Amr M. Youssef, editors, *Selected Areas in Cryptography*, volume 2259 of *Lecture Notes in Computer Science*, pages 1–24. Springer, 2001.

[24] Korek. Need security pointers. Published online at http://www.netstumbler.org/showthread.php?postid=89036#pos%t89036, 2004.

[25] Korek. Next generation of WEP attacks? Published online at http://www.netstumbler.org/showpost.php?p=93942&postcount=%35, 2004.

[26] Itsik Mantin. A practical attack on the fixed RC4 in the WEP mode. In Bimal K. Roy, editor, *ASIACRYPT*, volume 3788 of *Lecture Notes in Computer Science*, pages 395–411. Springer, 2005.

[27] Andreas Klein. Attacks on the RC4 stream cipher. *Des. Codes Cryptography*, 48(3):269–286, 2008. Published online in 2006, and accepted in WCC 2007 workshop.

[28] Erik Tews, Ralf-Philipp Weinmann, and Andrei Pyshkin. Breaking 104 bit WEP in less than 60 seconds. In Sehun Kim, Moti Yung, and Hyung- Woo Lee, editors, *WISA*, volume 4867 of *Lecture Notes in Computer Science*, pages 188–202. Springer, 2007.

[29] Serge Vaudenay and Martin Vuagnoux. Passive-only key recovery attacks on RC4. In Carlisle M. Adams, Ali Miri, and Michael J. Wiener, editors, *Selected Areas in Cryptography*, volume 4876 of *Lecture Notes in Computer Science*, pages 344– 359. Springer, 2007.

[30] Erik Tews and Martin Beck. Practical attacks against WEP and WPA. In David A. Basin, Srdjan Capkun, and Wenke Lee, editors, *WISEC*, pages 79–86. ACM, 2009.

[31] Pouyan Sepehrdad. *Statistical and Algebraic Cryptanalysis of Lightweight and Ultra-Lightweight Symmetric Primitives*. PhD thesis No. 5415, École Polytechnique Fédérale de Lausanne (EPFL), 2012. Available online at http://lasecwww.epfl.ch/~sepehrdad/Pouyan_Sepehrdad_PhD_Thesis.pdf.

[32] Pouyan Sepehrdad, Serge Vaudenay, and Martin Vuagnoux. Discovery and exploitation of new biases in RC4. In Alex Biryukov, Guang Gong, and Douglas R. Stinson, editors, *Selected Areas in Cryptography*, volume 6544 of *Lecture Notes in Computer Science*, pages 74–91. Springer, 2010.

[33] Pouyan Sepehrdad, Serge Vaudenay, and Martin Vuagnoux. Statistical attack on RC4 - distinguishing WPA. In Kenneth G. Paterson, editor, *EUROCRYPT*, volume 6632 of *Lecture Notes in Computer Science*, pages 343–363. Springer, 2011.

[34] Pouyan Sepehrdad, Petr Susil, Serge Vaudenay, and Martin Vuagnoux. Smashing WEP in a passive attack. In *Fast Software Encryption (FSE)*, 2013.

[35] Lars R. Knudsen, Willi Meier, Bart Preneel, Vincent Rijmen, and Sven Verdoolaege. Analysis methods for (alleged) RC4. In Kazuo Ohta and Dingyi Pei, editors, *ASIACRYPT*, volume 1514 of *Lecture Notes in Computer Science*, pages 327–341. Springer, 1998.

[36] Serge Mister and Stafford E. Tavares. Cryptanalysis of RC4-like ciphers. In Stafford E. Tavares and Henk Meijer, editors, *Selected Areas in Cryptography*, volume 1556 of *Lecture Notes in Computer Science*, pages 131–143. Springer, 1998.

[37] Jovan Dj. Golic. Iterative probabilistic cryptanalysis of RC4 keystream generator. In Ed Dawson, Andrew Clark, and Colin Boyd, editors, *ACISP*, volume 1841 of *Lecture Notes in Computer Science*, pages 220– 233. Springer, 2000.

[38] Yoshiaki Shiraishi, Toshihiro Ohigashi, and Masakatu Morii. An improved internal-state reconstruction method of a stream cipher RC4. In M.H. Hamza, editor, *Communication, Network, and Information Security, Track 440–088*, New York, USA, December 2003.

[39] Violeta Tomasevic, Slobodan Bojanic, and Octavio Nieto-Taladriz. Finding an internal state of RC4 stream cipher. *Inf. Sci.*, 177(7):1715–1727, 2007.

[40] Alexander Maximov and Dmitry Khovratovich. New state recovery attack on RC4. In DavidWagner, editor, *CRYPTO*, volume 5157 of *Lecture Notes in Computer Science*, pages 297–316. Springer, 2008.

[41] Jovan Dj. Golic and Guglielmo Morgari. Iterative probabilistic reconstruction of RC4 internal states. *IACR Cryptology ePrint Archive*, 2008:348, 2008.

[42] Sourav Sen Gupta, Subhamoy Maitra, Goutam Paul, and Santanu Sarkar. Proof of empirical RC4 biases and new key correlations. In Ali Miri and Serge Vaudenay, editors, *Selected Areas in Cryptography*, volume 7118 of *Lecture Notes in Computer Science*, pages 151–168. Springer, 2011.

[43] Sourav Sen Gupta, Subhamoy Maitra, Goutam Paul, and Santanu Sarkar. "(Non-) Random Sequences from (Non-) Random Permutations—Analysis of RC4 Stream Cipher." *Journal of Cryptology* 27, no. 1 (2014): 67-108.

[44] Takanori Isobe, Toshihiro Ohigashi, Yuhei Watanabe, and Masakatu Morii. "Full plaintext recovery attack on broadcast RC4." In *Proc. the 20th International Workshop on Fast Software Encryption (FSE 2013)*, 2013.

[45] Santanu Sarkar, Sourav Sen Gupta, Goutam Paul, and Subhamoy Maitra. Proving TLS-attack related open biases of RC4. *IACR Cryptology ePrint Archive*, 2013:502, 2013.

[46] Robert J. Jenkins Jr. ISAAC and RC4. Published on the Internet at http://burtleburtle.net/bob/rand/isaac.html, 1996.

[47] Itsik Mantin and Adi Shamir. A practical attack on broadcast RC4. In Mitsuru Matsui, editor, *FSE*, volume 2355 of *Lecture Notes in Computer Science*, pages 152–164. Springer, 2001.

[48] Itsik Mantin. Analysis of the stream cipher RC4. Master's thesis, The Weizmann Institute of Science, Israel, 2001. Available online at http: //www.wisdom.weizmann.ac.il/~itsik/RC4/RC4.html.

[49] Goutam Paul, Subhamoy Maitra, and Rohit Srivastava. On non-randomness of the permutation after RC4 key scheduling. In Serdar Boztas and Hsiao feng Lu, editors, *AAECC*, volume 4851 of *Lecture Notes in Computer Science*, pages 100–109. Springer, 2007.

[50] Santanu Sarkar. Further non-randomness in RC4, RC4A and VMPC. In *International Workshop on Coding and Cryptography (WCC)*, 2013.
[51] Subhamoy Maitra, Goutam Paul, and Sourav Sen Gupta. Attack on broadcast RC4 revisited. In Antoine Joux, editor, *FSE*, volume 6733 of *Lecture Notes in Computer Science*, pages 199–217. Springer, 2011.
[52] Nadhem AlFardan, Dan Bernstein, Kenneth G. Paterson, Bertram Poettering, and Jacob C.N. Schuldt. On the security of RC4 in TLS. In *USENIX Security Symposium*, 2013. Presented at FSE 2013 as an invited talk [14] by Dan Bernstein. Full version of the research paper and relevant results are available online at http://www.isg.rhul.ac.uk/tls/.
[53] Jovan Dj. Golic. Linear statistical weakness of alleged RC4 keystream generator. In Walter Fumy, editor, *EUROCRYPT*, volume 1233 of *Lecture Notes in Computer Science*, pages 226–238. Springer, 1997.
[54] Scott R. Fluhrer and David A. McGrew. Statistical analysis of the alleged RC4 keystream generator. In Bruce Schneier, editor, *FSE*, volume 1978 of *Lecture Notes in Computer Science*, pages 19–30. Springer, 2000.
[55] Itsik Mantin. Predicting and distinguishing attacks on RC4 keystream generator. In Ronald Cramer, editor, *EUROCRYPT*, volume 3494 of *Lecture Notes in Computer Science*, pages 491–506. Springer, 2005.
[56] Riddhipratim Basu, Shirshendu Ganguly, Subhamoy Maitra, and Goutam Paul. A complete characterization of the evolution of RC4 pseudo random generation algorithm. *J. Mathematical Cryptology*, 2(3):257–289, 2008.
[57] Adam Stubblefield, John Ioannidis, and Aviel D. Rubin. Using the Fluhrer, Mantin, and Shamir attack to break WEP. In *NDSS*. The Internet Society, 2002.
[58] Gong, Guang, Kishan Chand Gupta, Martin Hell, and Yassir Nawaz. "Towards a general RC4-like keystream generator." In *Information Security and Cryptology*, pp. 162-174. Springer Berlin Heidelberg, 2005.
[59] Orumiehchiha, Mohammad Ali, Josef Pieprzyk, Elham Shakour, and Ron Steinfeld. "Cryptanalysis of RC4 (n, m) Stream Cipher." In *Proceedings of the 6th International Conference on Security of Information and Networks*, pp. 165-172. ACM, 2013.
[60] Maitra, S., & Paul, G. Analysis of RC4 and proposal of additional layers for better security margin. In *Progress in CryptologyINDOCRYPT 2008* (pp. 27-39). Springer Berlin Heidelberg.
[61] Banik, Subhadeep, Santanu Sarkar, and Raghu Kacker. "Security Analysis of the RC4+ Stream Cipher." In *Progress in Cryptology–INDOCRYPT 2013*, pp. 297-307. Springer International Publishing, 2013.
[62] Xie, J., & Pan, X. An improved RC4 stream cipher. In *Computer Application and System Modeling (ICCASM), 2010 International Conference on* (Vol. 7, pp. V7-156). IEEE.
[63] Hammood, M. M., Yoshigoe, K., & Sagheer, A. M. (2013). RC4-2S: RC4 Stream Cipher with Two State Tables. In *Information Technology Convergence* (pp. 13-20). Springer Netherlands.
[64] Paul, G., Maitra, S., & Chattopadhyay, A. Quad-RC4: Merging Four RC4 States towards a 32-bit Stream Cipher. *IACR Cryptology ePrint Archive*, *2013*, 572.
[65] Kherad, F. J., Naji, H. R., Malakooti, M. V., & Haghighat, P. A new symmetric cryptography algorithm to secure e-commerce transactions. In *Financial Theory and Engineering (ICFTE), 2010 International Conference on* (pp. 234-237). IEEE.
[66] Weerasinghe, T. D. B. An Effective RC4 Stream Cipher. *IACR Cryptology ePrint Archive*, *2014*, 171
[67] Lv, J., Zhang, B., & Lin, D. Distinguishing Attacks on RC4 and A New Improvement of the Cipher. *IACR Cryptology ePrint Archive*, *2013*, 176.
[68] Khine, L. L. A New Variant of RC4 Stream Cipher. *World Academy of Science, Engineering and Technology*, *50*.

Authors' profiles

Poonam Jindal received B.E degree in Electronics and Communication Engineering from Punjab Technical University, Punjab in 2003, M.E degree in Electronics and Communication Engineering from Thapar University, Patiala in 2005 (India). She is working as Assistant Professor with Electronics and Communication Engineering Department, National Institute of Technology, Kurukshetra, India and currently pursuing her Doctoral Degree at National Institute of Technology, Kurukshetra, India**.** She has published 15 research papers in International/National conferences. Her research interests include security algorithms for wireless networks and mobile communication. She is a member of IEEE.

Prof. Brahmjit Singh as completed Bachelor of Engineering in Electronics Engineering from Malaviya National Institute of Technology, Jaipur, Master of Engineering with specialization in Microwave and Radar from Indian Institute of Technology, Roorkee and Ph.D. degree from GGS Indraprastha University, Delhi. He is with the Department of Electronics and Communication Engineering, National Institute of Technology, Kurukshetra working as Professor having 24 years of teaching and research experience. He has held several administrative and academic positions in NIT Kurukshetra. These include Chairman ECE Department, Chairman Computer Engineering Department, Professor in-Charge Centre of Computing and Networking, and Member Planning and Development Board. He has published 80 research papers in International / National Journals and conferences, organized several conferences and short term courses. His current research interests include Wireless Sensor Networks, Cognitive Radio, and Security Algorithms for Wireless Networks and Mobility Management in wireless networks and planning & designing of Mobile Cellular Networks. He has been awarded The Best Research Paper Award on behalf of 'The Institution of Engineers (India)'. He is the member of IEEE, Life member of IETE, and Life Member of ISTE.

How to cite this paper: Poonam Jindal, Brahmjit Singh,"A Survey on RC4 Stream Cipher", IJCNIS, vol.7, no.7, pp.37-45, 2015.DOI: 10.5815/ijcnis.2015.07.05

