A Proposal on a Multiple-Cycle Binary Sequence Generator with a ST-LFSR

이 훈 재*, 최 희 봉**
(Hoon-Jae Lee* and Hee-Bong Choi**)

요 약 출력 키수열의 사이클 수가 암호 분야에서 새로운 평가 요소로서 제안된 바 있으나 대부분의 이진 수열 발생기는 출력 사이클이 1개 뿐인 것으로 알려져 있다. 본 논문에서는 다수 사이클을 갖는 발생기의 기본 함수로서 switching-tap LFSR과 이를 응용한 Rueppel 개선형 다수열 발생기를 제안하였다. 그리고 개선된 발생기에 대하여 주기, 선형 복잡도 및 출력 사이클 수에 대하여 안전성을 분석하였다.

Abstract The number of keystream cycle sequences has been proposed as a characteristic of binary sequence generators for cryptographic applications, but in general most binary sequence generators have the only one-cycle sequence. In this paper, we propose a switching-tap LFSR as a basic function of a multiple-cycle binary sequence generator and the improved Rueppel's multiple-cycle generator. Finally we analyze its period, linear complexity, and the number of its keystream cycle sequences.

1. Introduction

The two basic methods for encrypting text into ciphertext are stream and block ciphers. Stream ciphers encrypt text bit-by-bit and are used in commercial applications such as RC4[8]. The advantage of a stream cipher is that it is faster and much more efficient than block ciphers. For example, RC4 is close to twice as fast as the nearest block cipher and can be written in 30 lines of code whereas the typical block cipher algorithm takes several hundred lines of code, making them ideal for Internet applications like SSL were speed and efficiency is most valuable[10].

The problem with stream ciphers is twofold[10]. First, to be implemented properly, each encryption key should be used only once. Using a key more frequently makes it much more vulnerable to attack. The creation of multiple keys can cause key management issues in applications where keys need to be stored for long periods of time. The second problem is that stream ciphers have not been widely adopted, so interoperability becomes an issue. If the person on the receiving end of the message does not have the ability to decrypt a stream cipher message, it is pointless to use.

On the other hand, Beker and Piper [1] proposed three basic requirements of stream cipher (cryptography) : a long period, randomness and a large linear complexity(LC). Subsequently, Siegenthaler [3] proposed to a requirement for a large correlation-immune property and Golic [4] suggested the number of output sequences (multiple-cycle sequence). Now we mean that the security requirement of the number of output sequence is a great solution of the multiple keys' problem, the first problem in reference [10].

Most binary sequence generators (BSG) in the literature have the only one-cycle sequence. In this case they always generate an output sequence having different starting points on the same cycle sequence. However, especially in cases of having multiple-cycle sequences, they generate different sequences from different cycles by changing initial values (keys). The

* 동서대학교 인터넷공학부(hjlee@dongseo.ac.kr),
** 국가보안기술연구소(hbchoi@etri.re.kr)

greater the number of cycle sequences, the stronger the resistance to attack on the BSG.

In this paper, we present that a single-cycle sequence generator has a cryptographic weakness to Dawson's attack [5] in a special case, but not in a multiple-cycle sequence. And we propose a switching-tap LFSR (ST-LFSR) which is selected from one of N $(2 \le N \le \lambda(n))$ feedback-taps in memory by the initial value (secret key). Then we apply it to the generalized binary sequence generator with a nonlinear combine function and improve upon Rueppel's generator as an example. Finally, we analyze the period, the linear complexity and the number of keystream cycle sequences of the improved Rueppel's multiple-cycle generator.

2. The number of keystream cycle sequences

Because n-stage LFSR (linear feedback shift register) generates linear recurring sequences, it is possible to predict feedback-connection-taps from a known 2 n-bit output sequence [1-2]. To strengthen nonlinearity in general we increase the linear complexity by a nonlinear combining function from N LFSRs.

Property 1. All of the n-stage LFSRs generate the maximum period, $P = (2^n - 1)$, only when non-null initial states [1-2]. And n-stage LFSR is obtained from the nth order of primitive polynomials.

Property 2. The number of the nth order of primitive polynomials is $\lambda(n) = \frac{\Phi(2^n - 1)}{n}$ [2]. Where $\Phi(\cdot)$ is the Euler's totient function.

It is required to synchronize two keystream output sequences both in the sender and in the receiver for a secure communication; we call this "keystream synchronization". Keystream synchronization initiates all of the shift registers in the generator by the secret key and coincides with each two starting points of a keystream cycle both in the sender and in the receiver.

On the other hand, Dawson [5] proposed that re-using the same keystream may be subject to attack. He suggested that the attack was fully automated and based on knowledge of the plaintext statistics only.

Let the past plaintext be $P' = p_0', p_1', p_2', \ldots$, the past keystream $K' = k_0', k_1', k_2', \ldots$, the past ciphertext $C' = c_0', c_1', c_2', \ldots$, the present plaintext $P = p_0, p_1, p_2, \ldots$, the present keystream $K = k_0, k_1, k_2, \ldots$ and the present ciphertext $C = c_0, c_1, c_2, \ldots$.

From the assumption that the same key was used in generating keystream ($K = K'$) both the sender and the receiver,

$$C' = p_0' \oplus k_0', p_1' \oplus k_1', p_2' \oplus k_2', p_3' \oplus k_3', p_4' \oplus k_4', p_5' \oplus k_5', \ldots \quad (1)$$

$$C = p_0 \oplus k_0, p_1 \oplus k_1, p_2 \oplus k_2, p_3 \oplus k_3, p_4 \oplus k_4, p_5 \oplus k_5, \ldots \quad (2)$$

$$C \oplus C' = p_0' \oplus p_0, p_1' \oplus p_1, p_2' \oplus p_2, p_3' \oplus p_3, p_4' \oplus p_4, p_5' \oplus p_5, \ldots \quad (3)$$

Because XORing the present ciphertext with the past ciphertext results in XORing of two plaintexts, it is breakable from the redundancy of the plaintext by Dawson's attack. Therefore in each communication it is required that a different keystream cycle be used and that synchronization on the secret session key be estabilished.

Definition 3. *The number of keystream cycle sequences* is defined to the total number of different- and same-sized keystream cycle sequences, which is changeable by the initial values on a keystream generator.

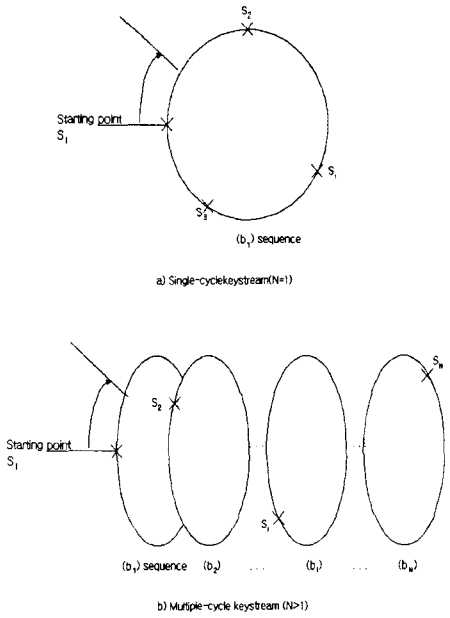


Fig. 1. Keystream cycle structures.

Fig. 1 illustrates the relationship between the number of keystream cycles and the crypto-degree in a stream cipher. In general, in single-cycle sequence cases (Fig. 1 a)), keystreams are generated from only one cycle, which has different starting points, S_1 and S_2, from different session keys. In this case their folding up on the same cycle makes them vulnerable to Dawson's attack. On the other hand, in multiple-cycle sequence cases (Fig. 1 b)), the probability of the cycle sequence folding up is nearly zero, because most of the cycle sequences are generated from different cycles on different session keys. The sequences are more difficult to predict in multiple-cycle than those in single-cycle, and so they are similar to the one-time pad in the best cases (a different key leads to a different cycle). Let the two different cycles $(b_i), (b_j)\,(i \neq j)$ of period P be as follows:

$$(b_i) = b_{i0}, b_{i1}, b_{i2}, b_{i3}, \ldots, b_{ik}, \ldots, b_{i,P-1}, \quad b_{i0}, b_{i1}, \ldots, i = 1, 2, \ldots, N \tag{4}$$

$$(b_j) = b_{j0}, b_{j1}, b_{j2}, b_{j3}, \ldots, b_{jk}, \ldots, b_{j,P-1}, \quad b_{j0}, b_{j1}, \ldots, j = 1, 2, \ldots, N \tag{5}$$

Definition 4. For integer k ($0 \leq k \leq P$) and j, if two sequences (b_i) and $\overleftarrow{(b_j)}$, k times cyclic rotations of the sequence (b_i), are different from each other and coincide only when $j = i$ over N, then *the number of keystream cycle sequences* is N:

$$\overleftarrow{(b_j)} = Rot((b_j), k) \begin{cases} \neq (b_i) & \text{for } j \neq i \\ = (b_i) & \text{for } j = i \\ & \text{over } N \end{cases} \tag{6}$$

where $Rot((x), k)$ is k times cyclic rotations of the sequence (x).

In this paper we propose to enlarge the number of keystream cycle sequences by changing the linear input part in a generalized sequence generator with the nonlinear combine function.

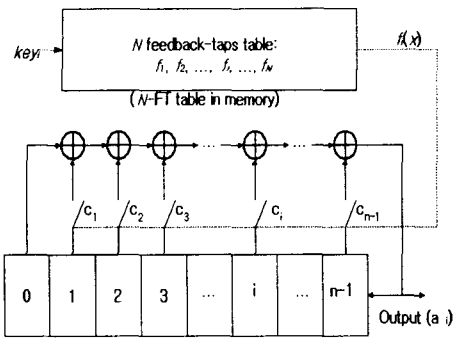
3. Switching-tap multiple-cycle binary sequence generator

1) Switching-tap LFSR

Theorem 5. A n-stage LFSR has a unique cycle.

Proof. A n-stage LFSR has the maximum period, $2^n - 1$, and it excepts the only null state over all 2^n states. Therefore, it has a unique cycle, and altering the initial value to the LFSR feedback connection generates the same cycle sequence, too, but at a different starting point. □

A LFSR with a fixed feedback-tap has a unique cycle, and the total number of LFSR's feedback connections with length n are $\lambda(n) = \frac{\Phi(2^n - 1)}{n}$, which is the maximum possible number of configurations in feedback-tap. In this way the switching-tap LFSR (ST-LFSR, Fig. 2) is a specially configured LFSR which switches the feedback-tap function f_i of the FT (feedback-tap) table in memory, selected by session key.



$$f_i(x) = x^n + c_{n-1}x^{n-1} + c_{n-2}x^{n-2} + \ldots + c_1 x + 1, \quad i = 1, 2, \ldots, N$$

Fig. 2. A n-stage ST-LFSR.

Property 6. Let $N(2 \le N \le \lambda(n))$ be the number of feedback-tap functions in a n-stage ST-LFSR, stored in memory. Then a n-stage ST-LFSR has the number of keystream cycle sequences with N.

Reference [6] mentioned an easy method for finding n-stage feedback connections, primitive polynomials. From a properly selected N, smaller than or equal to $\lambda(n)$, we can generate the sequences with N multiple-cycles.

2) Switching-tap multiple-cycle binary sequence generator

The generalized binary sequence generator with a nonlinear combine function has a lot of LFSRs. We apply ST-LFSR to the generalized binary sequence generator by changing the LFSR feedback connections to improve the number of cycle sequences with N. We can choose a proper N in a LFSR or a set of M LFSRs : t $(t=0,1,2,\ldots)$ instant output sequence of (b_i), $i=1,2,..N$, is $b_i(t) = f(a_1(t), a_2(t), \ldots, a_{Mi}(t))$ where $a_j(t)$ is a t-instant output sequence of $LFSR_j$ $(j=1,2,\ldots,M)$ and $a_{Mi}(t)$ is the t-instant output sequence of $LFSR_M$ which is a ST-LFSR and at least $a_{Mi}(t)$ has N multiple-cycles, so (b_i) has N multiple-cycles too. For example, we can improve Rueppel's generator [7] by changing the first m-stage LFSR to a m-stage ST-LFSR as in Fig. 3.

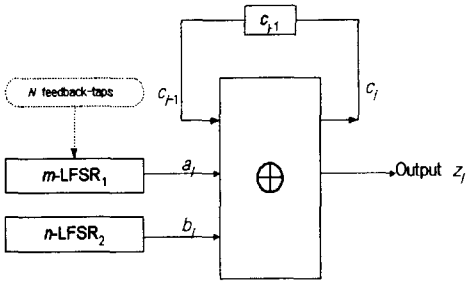


Fig. 3. An improved Rueppel's multiple-cycle generator.

Theorem 7. The improved Rueppel's generator has the following cryptographic properties:

1) Period, $P = (2^m - 1)(2^n - 1)$.

2) Linear complexity, $LC \approx P$.

3) The number of keystream cycle sequences with N.

Proof. 1) and 2). After setting the initial values to two LFSR, LFSR1 and LFSR2, the improved generator operates the same as the origin. Therefore, the properties of the period and the linear complexity of the output sequence are the same as that of the origin[7-10].

3) Because the improved generator can be changeable of the feedback functions by N, It should have been the number of N-multiple cycle sequences by Definition 3. □

In terms of the cryptographic strength of the improved Rueppel's generator, it has the same period and the same linear complexity, but the number of cycle sequences is improved N times over the original Rueppel's generator because ST-LFSR is a kind of LFSR for a full period.

4. Conclusion

In this paper we adapted the number of cycle sequences to the element of cryptographic strength in stream cipher. In multiple-cycle sequence cases, the probability of the cycle sequence folding up was nearly zero, because the most of the sequences were generated from different cycles on different session keys. In these cases the sequences were more difficult to predict in multiple-cycle than those in single-cycle and so they were similar to the one-time pad in the best cases (a different key leads to a different cycle). We proposed a multiple-cycle binary sequence generator based on a switching-tap LFSR, which was selected from one of N $(2 \leq N \leq \lambda(n))$ feedback-taps in the memory table by an initial value (key). We then applied it to Rueppel's generator as an example, which had the same period and the same linear complexity, but the number of keystream cycle sequences was improved N times to the original generator.

References

[1] Henry J. Beker and Fred C. Piper, *Cipher systems: The Protection of Communica-tions*, Northwood Books, London, 1982.

[2] Henk C.A. van Tilborg, *Fundamentals of Cryptology*, KLUWER ACADEMIC PUBLISHERS, Boston, etc., 2000.

[3] T. Siegenthaler, "Correlation-Immunity of Nonlonear Combining Functions for Crypto-graphic Applications," *IEEE Trans. on Infor. Theo.*, Vol. IT-30, No.5, pp.776-780, 1984.

[4] J. Dj. Golic, "The Number of Output Sequences of a Binary Sequence Generator," *Advances in Cryptology -EUROCRYPT'91, Lecture Notes in Computer Science*, Vol.547, pp. 160-167, 1991.

[5] E. Dawson, L. Nielsen, "Automated Cryptanalysis of XOR Plaintext Strings," *Cryptologia*, Vol. XX, No. 2, pp.165-181, 1996.

[6] B. Park, H. Choi, T. Chang and K. Kang, "Period of Sequences of Primitive Polynomials," *Electronics Letters*, Vol. 29, No. 4, pp.390-391, 1993.

[7] R. A. Rueppel, "Correlation Immunity and the Summation Generator," *Advances in Cryptology-Proceedings of CRYPTO'85*, LNCS 218, Springer-Verlag, Berlin, pp. 260-272, 1985.

[8] A. Menezes, P. Oorschotand, S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.

[9] Hoonjae Lee, Sangjae Moon, "Parallel Stream Cipher for Secure High-Speed Communications," *Signal Processing*, Vol. 82, No.2, pp.259-265, Feb. 2002.

[10] Wedbush Morgan Securities - Industrial Report, "Access Management/Internet Security Industry," on http://www.vikasqupta.com, Feb. 28, 2002.

이 훈 재(Hoon-Jae Lee)

1985년 2월 : 경북대학교 전자공학과 졸업(학사)
1987년 2월 : 경북대학교 전자공학과 졸업(석사)
1998년 2월 : 경북대학교 전자공학과 졸업(박사)
1987년 2월~1998년 1월 : 국방과학연구소 선임연구원
1998년 2월~2002년 2월 : 경운대학교 컴퓨터전자정보공학부 조교수
2002년 3월~현재 : 동서대학교 인터넷공학부 정보네트워크공학전공 조교수
<주관심 분야> 정보보호, 네트워크보안, 정보통신

최 희 봉(Hee-Bong Choi)

1984년 2월 : 부산대학교 전기공학과 졸업(학사)
1987년 2월 : 부산대학교 전기공학과 졸업(석사)
2002년 8월 : 성균관대학교 전전컴공학부 졸업(박사)
1987년 2월~2000년 1월 : 국방과학연구소 선임연구원
2000년 1월~현재 : 한국전자통신연구원 부설 국가보안기술연구소 선임연구원
<주관심 분야> 정보보호, 네트워크보안, 보안시스템 설계