Performance Evaluation of Stream Ciphers on Large Databases

Dr.M.Sikandar Hayat Khiyal **Aihab Khan** **Saria Safdar**

Department of Software Engineering Fatima Jinnah Women University Rawalpindi,Pakistan

Abstract
Alternating step generator and shrinking generator are most commonly used clock controlled based stream ciphers for the generation of key stream. In this study we attempt to provide solution to correlation attack by gradually increasing the lengths of initial input bits of linear feedback shift registers (LFSR's), which result in the increase of the key length. We implement both the algorithms and found that Shrinking Generator is secure at length of 64 and Alternating Step Generator is secure at the length of 128 against the correlation attack .It is also found that Shrinking Generator is more efficient and secure than Alternating Step Generator.

Keywords:
Correlation Attack, Linear Feedback Shift Registers (LFSR), Linear Complexity, Period, Pseudo Randomness, Stream Cipher.

1. Introduction

Stream cipher is a symmetric cipher in which plaintext bits are combined with a key stream, typically by an exclusive-or (XOR) operation. In stream cipher the plaintext digits are encrypted and the transformation of successive digits varies during the encryption. An alternative name of a stream cipher is *state cipher*, as the encryption of each digit is dependent on the current state [1,9,10]. Two clock controlled based stream ciphers are presented in this study and their common weaknesses are analyzed. The most important general attacks on Linear Feedback shift register (LFSR) based stream ciphers are correlation attacks. In correlation attack if a cryptanalyst can detect a correlation between the output sequences (known) and the output of one individual LFSR's sequence, this can be used in a divide and conquer attack on the individual LFSR. In this study we start with smaller inputs to LFSR's and then by gradually increasing the length of inputs; we found that key length also increases which gradually makes it difficult to find correlation between the known output and the output of individual LFSR.

2. Related Work

Gunther [2] described the Alternating Step generator (ASG) which is a suitable crypto-generator for stream ciphers applications because its output sequences have periods and large linear complexities. The key stream produced is the XOR of the output sequences of second and third register.

The shrinking generator of *Coppersmith et al [3]*, (SHKG) is a suitable crypto-generator for stream ciphers applications for two reasons; firstly it has a nice statistical output, and secondly, it has a large period and large linear complexity**.** Coppersmith's construction uses two sources of pseudorandom bits to create a third source of pseudorandom bits of potentially better quality than the original sources.

Tasheva et al [4], described a Pseudo Random Number Generator (PRNG), named N-adic Summation-Shrinking (NSumSG), which uses parallel working slave summation generators (registers) and one summation generator, controlling the nonlinearity in the generator. The *NSumG* architecture uses an increased number of slaved registers in comparison with Shrinking Generator. The control and slave registers in shrinking multiplexing generator are replaced with *N*-adic and 2-adic summation generators in the *NSumG* respectively.

3. Correlation Attack

The most important general attacks on LFSR-based stream ciphers are correlation attacks. Correlation attack was originally proposed by Siegenthaler [5].If an opponent can detect a correlation between the output sequence and the output sequence of one individual LFSR, this can be used in a "divide-and-conquer" attack to recover the initial inputs of the individual LFSR .

In the case of the Shrinking Generator the sequence **a** (output of register 1) can be recovered from the output sequence **z** if we can solve the corresponding decoding problem on the deletion channel. [6]

In the case of ASG we assume that the sequence a = a1, a2…is the input to the insertion channel and the sequence z = z1, z2… is the output, the requirements for the insertion channel is fulfilled and the parameter q defines the probability for the insertion channel which is q = 1/2.[6]

Manuscript received September 5, 2008.
Manuscript revised September 20, 2008.

3.1 MAP Decoding Algorithm:

MAP decoding algorithm describes the process of correlation attack, as well as the way how it proceeds on alternating step generator and shrinking generator. By definition a "MAP decoding algorithm needs an input sequence a that for given z maximizes P(a transmitted |z received), whereas a ML decoding algorithm needs a sequence a maximizing P(z received |a transmitted)".[6]
The decoding algorithm used in correlation attacks can be divided into two families. The first one consists of decoding procedures which make use of the inherent structure of the code, especially when it corresponds to a LFSR. The second family contains general algorithms that can be applied to any linear code. [7]

3.1.1 Working of MAP Decoding Algorithm:

If we assume that $a_1 \ldots a_{LA}$ is the given initial state of LFSR A at time zero .Each initial state gives rise to a corresponding infinite sequence $a = a_1, a_2 \ldots$
Let *A* denotes the set of possible sequences. Output sequence z is also an infinite sequence $z = z_1, z_2 \ldots$ obtained by transmitting some sequence a over the deletion channel, i.e if the sequence $a = a_1, a_2 \ldots$ gives the output $z = z_1, z_2 \ldots$
Let $A = A_1, A_2 \ldots$ and $Z = Z_1; Z_2 \ldots$ be the corresponding random variables. Continuing, we consider input sequences of fixed length t. Let a^t denote the sequence $a^t = a_1, a_2 \ldots a_t$ and let $A^t = A_1, A_2 \ldots A_t$ be the corresponding random variable. For a fixed length t the MAP decoding procedure calculates. [6]

$$P(At = a^t | Z = z);$$

A_t are the random variables input by the hacker to get the sequence. The length of the output sequence after t input symbols can be any value in [0, t].
We can then write the above equation as

$$P(At = at \mid Z = z) = \sum_{i=0}^{t} P(At = at, \Phi t = i \mid Z = z)$$

where *i* is the number of iterations used by the opponent.
By iteratively increasing i we get

$$P(At = at, \Phi t = i \mid Z = z) = P(At - 1 = at - 1, \Phi t - 1 = i \mid Z = z)$$
$$P(At = at, \Phi t = i \mid \Phi t - 1 = i, z = z) + P(At - 1 = at - 1, \Phi t - 1 = i - 1, Z = z)$$

We further observe that

$$P(At = at, \Phi t = i \mid \Phi t - 1 = i, Z = z) = 1/4$$

Since deletion occurs with probability 1/2 and then At = at also with probability
1/2. Furthermore

$$P(At = at, \Phi t = i \mid \Phi t - 1 = i - 1, Z = z)$$

This equation is equal to
= {1/2 if at=zi,
0 otherwise}

Because in this case is no deletion, which occur with probability 1/2. Then At = zi and thus At = at has probability 1 if at = zi and 0 otherwise. By moving step by step a strong correlation can easily be found. [6]

4. Clock Controlled Generators:

In a clock-controlled generator, main idea is to introduce nonlinearity into LFSR based key stream generators by having the output of one LFSR control the clocking of a second LFSR. As second LFSR is clocked in an irregular manner, it may be expected that attacks based on the regular motion of LFSRs can be foiled. Two clock controlled generators are; the alternating step generator and the shrinking generator [8].

4.1 Alternating Step Generator:

Alternating Step Generator (ASG) is considered as cryptographic pseudorandom number generator intended to be used in a stream cipher. The design was published in 1987 by C. G. Günther [2]. Another name of Alternating Step Generator is stop-and-go generator.

4.1.1Model:

Fig 1 represents the model of Alternating Step Generator.

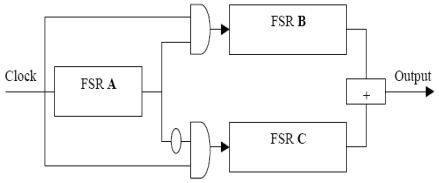


Fig. 1 Alternating Step Generator [8]

4.1.2Algorithm:

Following steps are repeated until a key stream of desired length is produced.
1. Register R1 is clocked.
2. If the output of R1 is 1 then:
R2 is clocked; R3 is not clocked but its previous output bit is repeated. (For the first clock cycle, the "previous output bit" of R3 is taken to be 0.)
3. If the output of R1 is 0 then: R3 is clocked; R2 is not clocked but its previous output bit is repeated.

(For the first clock cycle, the "previous output bit" of R2 is taken to be 0.)
4. The output bits of R2 and R3 are XORed; the resulting bit is part of the key stream. [8]

4.1.3 Input to Algorithms:

Both the algorithms are applied on Telecommunication Company Limited's exchange dataset.

Name
Abazai
ABBA KHEL
ABDUL KHEL DIK
ABDUL KHEL LKI
ACHOO KHEL
Ade Zai
AGARAI
AGHZER KHEL
AGRA
AHMAD ABAD
AHMAD KHEL GHARBI
AHMAD KHEL SHARKI
Aka Khel
Akbar Pura
AKHAGRAM
Akora Khattak
Ali Masjid
ALIZAI
ALLO QASIMI

Fig. 2 Input to Algorithms

In Fig 2 name column is encrypted using Alternating Step Generator and Shrinking Generator algorithms.

4.1.4 Analysis of Alternating Step Generator (ASG) through Different Case Studies:

Case 1:

We started with initial input bits to Linear Feedback Shift Registers with lengths 3 4 5.

Table 1 ASG with Lengths 3 4 5

LFSR	Length	Sequence (Period 2^L-^{1})
1	3	7
2	4	15
3	5	31

KEY: 1011101
In case 1 key produce is 7 bits as LFSR1 is the control register according to its length key bits are produced. As the length of the key is very small there are more chances of correlation attack. After generating 4095 sequence bits, patterns start repeating. So for these lengths ASG is vulnerable to correlation attack.

Case 2:

Table 2 ASG with Lengths 11 13 13

LFSR	Length	Sequence(Period 2^L-^{1})
1	11	2047
2	13	8191
3	13	8191

KEY: 0101011001010001000…
In case 2 key produce is 2047 bits, as the length of the key is not very large there is probability of correlation attack. After generating $1.374389534*10^{11}$ key bits, patterns start repeating

Case 3:

Table 3 ASG with Lengths 128 128 128

LFSR	Length	Sequence(Period 2^L-^{1})
1	128	$3.402823667*10^{38}$
2	128	$3.402823667*10^{38}$
3	128	$3.402823667*10^{38}$

In case 3 generated key is very large so possibility of correlation attack is reduced.

4.2 Shrinking Generator:

Shrinking generator is considered as pseudorandom number generator which is intended to be used in a stream cipher as a sequence generator. It was published in 1993 by Don Coppersmith, Hugo Krawczyk and Yishay Mansour [3].Two linear feedback shift registers (LFSR) are used to generate the sequence. LFSR A, generates output bits, while the other, LFSR S, controls their output.

4.2.1 Model:

Fig 3 represents the model of Shrinking Generator.

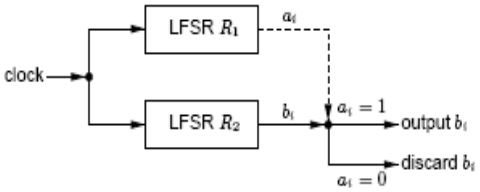


Fig. 3 Shrinking Generator [8]

4.2.2 Algorithm:

The following steps are repeated until a key stream of desired length is produced.
1. Registers R1 and R2 are clocked.

2. If the output of R1 is 1, the output bit of R2 forms part of the key stream.
3. If the output of R1 is 0, the output bit of R2 is discarded.[8]

4.2.3 Analysis of Shrinking Generator (SG) through Different Case Studies:

Case 1:

We started with initial input bits to Linear Feedback Shift Register with lengths 3 5.

Table 4 SG with Lengths 3 5

LFSR	Length	Sequence
1	3	7
2	5	31

KEY: 1000011
In case 1 key produced is 7 bits as LFSR1 is the control register according to the length of its initial input bits key is produced. As the length of the key is very small there are more chances of correlation attack.

Case 2:

Table 5 SG with Lengths 11 13

LFSR	Length	Sequence
1	11	2047
2	13	8191

KEY: 1011100001…
In case 2 key produced is 2047, as the length of the key is very small there are more chances of correlation attack. After generating 16777215 key bits patterns starts repeating.

Case 3:

Table 6 SG with Lengths 64 64

LFSR	Length	Sequence
1	64	$1.844674406*10^{19}$
2	64	$1.844674406*10^{19}$

In case 3 when initial inputs to both LFSR's are 64, generated key is in millions, so the possibility of Correlation attack is reduced.

4.3 Comparison of Alternating Step Generator and Shrinking Generator:

Table 7 comparison of ASG and SG

Properties	Alternating Step Generator	Shrinking Generator
Linear Feedback Shift Registers	3	2
Secure Length against correlation attack	128	64
Construction(Logic Gates)	And ,Not	No Gates
Structure	Complex	Simple
Period	$2^{L1}.(2^{L2}-1).(2^{L3}-1)$	$(2^{L2}1).2^{L1-1}$
Security	2^{L}	2^{2L}

After comparing both the algorithms with respect to the properties in table 7 it is found that Shrinking Generator seems to be a better choice to generate a key stream because of its simple structure and efficient nature.

4.4 Performance of Algorithms: Execution Time for Generation of Sequence:

Table 8 Alternating Step Generator with Time

Lengths of LFSR'S	Alternating Step Generator (Time in μsec)
LFSR1= 3 LFSR 2= 4 LFSR 3= 5	0.015625
LFSR1= 7 LFSR 2= 9 LFSR 3= 9	0.125
LFSR1= 11 LFSR 2= 13 LFSR 3= 13	1.96875

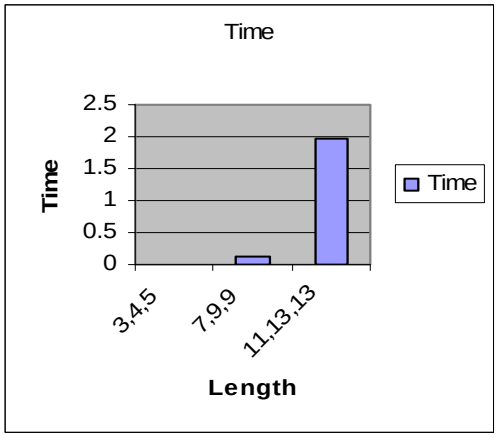
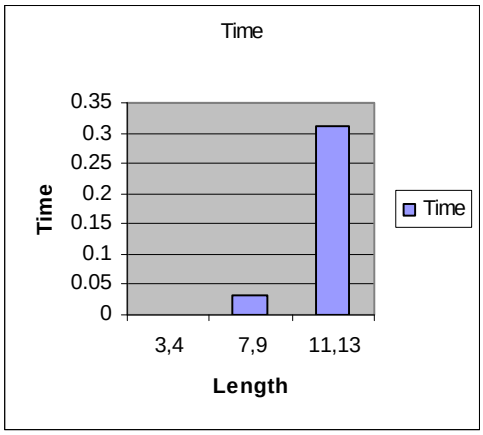


Table 9 Shrinking Generator with Time

Lengthsof LFSR'S	Shrinking Generator (Time in μsec)
LFSR1= 3 LFSR 2= 4	0
LFSR1= 7 LFSR 2= 9	0.03125
LFSR1= 11 LFSR 2= 13	0.3125



After executing both the algorithms at different lengths it is found that shrinking generator takes less time in generating sequence than Alternating Step Generator.

Conclusion:

Alternating Step Generator and Shrinking Generator are most commonly used stream ciphers for the generation of key stream. General attacks on these two stream ciphers are correlation attacks. Both the algorithms are implemented by gradually increasing the lengths of initial input bits to LFSR's. The increase in the initial input bits of LFSR's results in increase of the key length. In case of Shrinking Generator if L1, L2=l i.e GCD(L1,L2)=1 then shrinking generator has a security level approximately equal to 2^{2L} [8]. Secure lengths of Shrinking Generator and Alternating Step Generator against the correlation attack are 64 and 128. When initial inputs to LFSR's are 64 and 128, generated key length is very large, so the possibility of correlation attack is reduced. After comparing both the algorithms it is found that Shrinking Generator is a better choice to generate a key stream because of its simple structure and efficient nature.

Keystream sequence generators that produce sequences with large periods, high linear complexities and good statistical properties are very useful as building blocks for stream cipher applications. The use of clock-controlled generators in keystream generators appears to be a good way of achieving sequences with these properties.[11,12]

References:

[1] William Stallings "Cryptography and Network security" fourth edition

[2] C. G. Gunther, "Alternating step generators controlled by de Bruijn sequences", In proceedings of Eurocrypt 87, lecture notes in computer science,Berlin: Spinglerverlag vol .309, 1988 ,pp 5-14.

[3] Don Coppersmith, Hugo krawczyk, Yishay Mansour "Shrinking Generator" IBM T.J. Watson Research Center Yorktown Heights NY 10598. 1988

[4] Zhaneta Tasheva, Borislav Bedzhev, Borislav Stoyanov, "*N*-adic Summation-Shrinking Generator" Basic properties and empirical evidences

[5] T.Siegenthaler, "Decrypting a class of stream ciphers using ciphertext only," IEEE *Trans. Computers*, vol. C-34, no. 1, pp. 81-84, 1985

[6] Thomas Johansson" Reduced Complexity Correlation Attackson Two Clock-controlled Generators "Dept. of Information TechnologyLund University, P.O. Box 118, 221 00 Lund, Sweden

[7]Anne Canteaut "Fast correlation attacks against stream ciphers and related open problems"INRIA- project codes B.P. 105

[8] A. Menezes, P. van Oorschot and S. Vanstone "*Handbook of Applied Cryptography"* CRC Press, p. 780, 1997,

[9]G:\search\Stream cipher - Wikipedia, the free encyclopedia.htm

[10]G:\search\Shrinking generator - Wikipedia, the free encyclopedia.htm

[11]Ali Adel Kanso "Clock-Controlled Alternating Step Generator" King Fahd University of Petroleum and Minerals. 2002

[12] D. Gollmann and W. Chambers, "Clock-Controlled Shift Register: A Review", IEEE J.Sel. Ar. Comm. vol. 7, NO.4, May 1989, pp. 525-533.

Dr.M.Sikandar H.Khiyal born at Khushab, Pakistan. He is Chairman Dept. Computer Sciences and Software Engineering in Fatima Jinnah Women University Pakistan. He Served in Pakistan Atomic Energy Commission for 24 years and involved in different research and development program of the PAEC. He developed software of underground flow and advanced fluid dynamic techniques. He was also involved at teaching in Computer Training Centre, PAEC and International Islamic University. His area of interest are Numerical Analysis of Algorithm, Theory of Automata and Theory of Computation. He has more than forty five research publications published in National and International Journals and Conference proceedings. He has supervised more than sixty research projects at graduate and postgraduate level.

Mr. Aihab Khan works in Dept. of Computer Sciences Fatima Jinnah Women University Pakistan. His research interests are in the field of Data Mining, Data Warehousing as well as Information security.

Saria Safdar is a graduate from Dept. of Computer Science, Fatima Jinnah Women University Pakistan.