Seminar Nasional Teknologi Komputer & Sains (SAINTEKS)
SAINTEKS 2020
ISBN: 978-602-52720-7-3
Februari 2020
Hal 171-175

# Generate Pseudo-Random Numbers Linear-Feedback Shift Register (LSFR) Pada Kunci Algoritma One Time Pad (OTP)

**Oris Krianto Sulaiman**

Fakultas Teknik, Program Studi Teknik Informatika, Universitas Islam Sumatera Utara, Medan, Indonesia
Email: oris.ks@ft.uisu.ac.id

**Abstrak**−One Time Pad (OTP) merupakan salah satu algoritma kriptografi simetris (konvensional) dimana kunci untuk enkripsi dan dekripsi pesan adalah sama. Algoritma ini di anggap paling handal untuk keamanan pesan, algoritma One Time Pad memiliki panjang kunci sama dengan panjang plainteks atau teks asli, hal ini berakibat sulitnya membuat kunci jika plainteks mempunyai banyak karakter. LSFR merupakan generate pseudo-random numbers yang dapat digunakan untuk membuat kunci otomatis dari algoritma tertentu sehingga kunci tidak di input satu persatu melainkan terbentuk oleh LSFR. LSFR memanfaatkan karakter dari plainteks OTP untuk dijadikan kunci dengan menggunakan rumus dari LSFR sehingga panjang kunci yang sama dengan panjang karakter dari plainteks yang akan di enkripsi oleh OTP dapat dibuat secara otomatis.

**Kata Kunci:** One Time Pad (OTP), LSFR, Panjang Kunci.

## 1. PENDAHULUAN

Keamanan digital saat ini menjadi sangat penting untuk menjaga dari kejahatan siber, salah satu kemanan yang dibutuhkan adalah kerahasiaan pesan[1], pesan yang sangat rahasia tentunya tidak ingin sampai orang lain mengetahuinya, namun ketika proses komunikasi terjadi pesan yang dikirim akan melewati jalur publik *(internet)* salah satu contohnya adalah ketika melakukan komunikasi via email, pengirim yang ingin mengirimkan pesan melalui email akan mengirimkan ke tujuan dengan melewati jalur internet dimana jalur tersebut digunakan oleh semua orang, hal ini tentunya berakibat dari keamanan isi email tersebut karna bisa saja terjadi penyadapan oleh pihak ketiga atau sering juga disebut dengan istilah *man-in-the-middle attack* [2], [3].

Berbagai metode yang dapat digunakan untuk mengamankan kerahasiaan pesan salah satu nya adalah dengan menggunakan teknik kriptografi. Kriptografi merupakan teknik merubah pesan asli *(plaintext)* menjadi pesan acak *(ciphertext)*. Salah satu algoritma dari kriptografi yang dapat melakukan pengacakan pesan adalah *One Time Pad (OTP)* dikenal dengan algoritma yang cukup handal dan di klaim sempurna untuk keamanan pesan [4]. Algoritma OTP merupakan algoritma simetris dimana kunci yang digunakan untuk proses enkripsi dan dekripsi merupakan kunci yang sama[4], [5]. Panjang karakter kunci dari algoritma OTP harus sesuai dengan panjang karakter dari plainteks [5]. Jika karakter plainteks terdiri dari 3 karakter maka untuk kunci dari algoritma OTP harus terdiri dari 3 karakter, hal ini sangat menyulitkan ketika karakter plainteks terdiri dari banyak karakter sehingga untuk membuat kunci yang sama dibutuhkan waktu yang panjang[6]–[11]. Untuk itu dibutuhkan *generate pseudo-random numbers* untuk melakukan pengacakan kunci secara otomatis berdasarkan plainteks yang ada, untuk melakukan *generate pseudo-random numbers* dapat dilakukan dengan menggunakan *Linear-Feedback Shift Register* (LSFR).

*One Time Pad (OTP)* dapat dirumuskan:

$C_i = (P_i+K_i) \bmod 26$ (1)

$P_i = (C_i-K_r) \bmod 26$ (2)

Contoh dalam penerapan algoritma One Time Pad (OTP) adalah sebagai berikut:

*Plaintext:* ORIS

*Key:* SRIE

Masing-masing karakter akan di konversi menjadi angka mulai dari 0 sampai 25 contoh: A=0, B=1,……Z=25

**Tabel 1.** Proses enkripsi OTP

| *Plaintext* | *Numerical Plaintext* | *OTP* | *Numerical OTP* | *Numerical Ciphertext* $(P_i+K_i) \bmod 26$ | *Ciphertext* |
|---|---|---|---|---|---|
| O | 14 | S | 18 | 6 | G |
| R | 17 | R | 17 | 8 | I |
| I | 8 | I | 8 | 16 | Q |
| S | 18 | E | 4 | 22 | W |

Dari tabel 1 dapat terlihat hasil perhitungan *plaintext* menjadi *ciphertext* seperti berikut:14 + 18 mod 26 = 6 dikonversi menjadi huruf G

17 + 17 mod 26 = 8 dikonversi menjadi huruf I

8 + 8 mod 26 = 16 dikonversi menjadi huruf Q

18 + 4 mod 26 = 22 dikonversi menjadi huruf W

Sehingga *plaintext* = ORIS dengan kunci OTP=SRIE menghasilkan *ciphertext*=GIQW.

Dari sini dapat terlihat bahwa diharuskan panjang kunci OTP untuk melakukan enkripsi *plaintext* harus sama.

**Tabel 2.** Proses dekripsi OTP

ISBN: 978-602-52720-7-3
Februari 2020
Hal 171-175

| *Ciphertext* | *Numerical Ciphertext* | *OTP* | *Numerical OTP* | *Numerical Plaintext* $(P_i\text{-}K_i)$ mod 26 | *Plaintext* |
|---|---|---|---|---|---|
| G | 6 | S | 18 | 14 | O |
| I | 8 | R | 17 | 17 | R |
| Q | 16 | I | 8 | 8 | I |
| W | 22 | E | 4 | 18 | S |

Dari tabel 2 dapat terlihat hasil perhitungan *ciphertext* menjadi *plaintext* seperti berikut:
6 - 18 mod 26 = 14 dikonversi menjadi huruf O
8 - 17 mod 26 = 17 dikonversi menjadi huruf R
16 - 8 mod 26 = 8 dikonversi menjadi huruf I
22 - 4 mod 26 = 18 dikonversi menjadi huruf S
Sehingga *ciphertext*=GIQW dengan kunci OTP=SRIE menghasilkan *plaintext*=ORIS.

*Linear-Feedback Shift Register* (LSFR) merupakan *shift register* yang bit masukkanya merupakan fungsi linear dari *state* sebelumnya. Satu-satunya fungsi linear pada bit satuan adalah xor, oleh karena itu LSFR adalah *shift register* yang bit masukkannya dibangkitkan oleh *exlusive-or* (XOR) dari beberapa bit keseluruhan nilai *shift register* [8], [10], [11]. LSFR pada dasarnya terdiri dari 3 komponen yaitu [6]: rrutan input (inisialisasi vector), umpan balik (tap sequence) dan keluaran (output).
LSFR dapat dirumuskan:

$$e^{i+n+1} = a^1e^{i+1} + a^2e^{i+2} + \ldots + a^ne^{i+n} \bmod 26 \quad (3)$$

$$e^{1+3} = a^1e^{i+1} + a^2e^{i+2} \bmod 26 \quad (4)$$

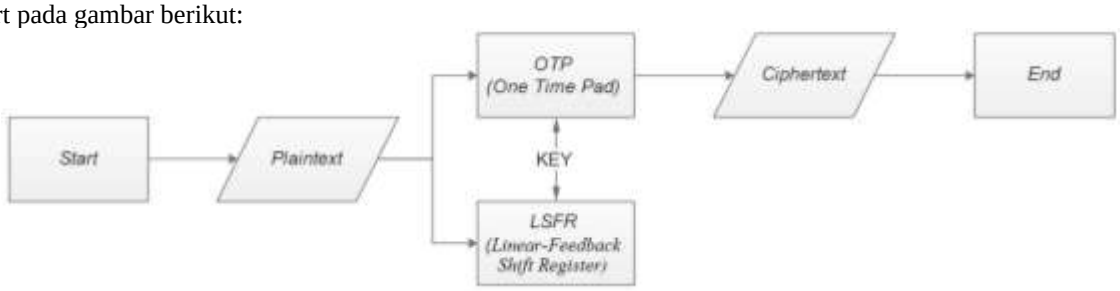
$$a^1 = 1 \qquad a^2 = 2 \quad (5)$$

$$e^{1+3} = e^{i+1} + 2e^{i+2} \bmod 26 \quad (6)$$

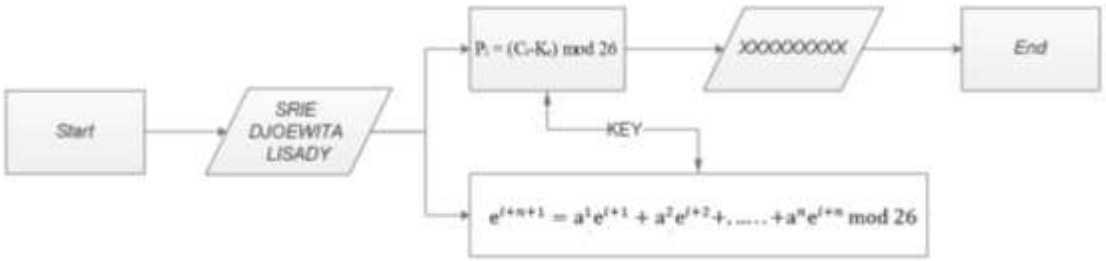
Contoh untuk *generate pseudo-random numbers* dengan menggunakan *Linear-Feedback Shift Register* adalah: menentukan 2 angka acak untuk $e^1$ dan $e^2$, contoh $e^1 = 5$ dan $e^2 = 10$ maka untuk melekukan pencarian $e^3 = e^1 + 2 * e^2$, dan seterusnya. LSFR pada text SRIE yang terdiri dari 4 karakter, maka akan terjadi sift:
$e^1 = 5 \; mod \; 26$
$e^2 = 10 \; mod \; 26$
$e^3 = 5 + (2*10) \bmod 26 = 25 \bmod 26$
$e^4 = 10 + (2*25) \bmod 26 = 8 \bmod 26$
Hasil yang didapat dari perhitungan LSFR adalah 5, 10, 25 dan 8.

## 2. METODOLOGI PENELITIAN

Penelitian ini menggunakan algoritma OTP untuk mengubah *plaintext* menjadi *ciphertext* sebagai keamanan pesan dan mempermudah dalam pembuatan kunci OTP dengan menggunakan fungsi dari *Linear-Feedback Shift Register* (LSFR) agar membentuk sebuah *generate pseudo-random numbers*. *Plaintext* yang digunakan adalah SRIE DJOEWITA LISADY, lalu kunci dari OTP akan bentuk melalui *plaintext* dengan *Linear-Feedback Shift Register* (LSFR). Hasil dari LSFR yang telah menjadi kunci OTP akan di operasikan dengan *plaintext* sehingga menghasilkan *ciphertext*. Alur ini dapat terlihat dari flowchart pada gambar berikut:




**Gambar 1.** Flowchart enkripsi *generate pseudo-random numbers* LSFR pada kunci OTP (1)

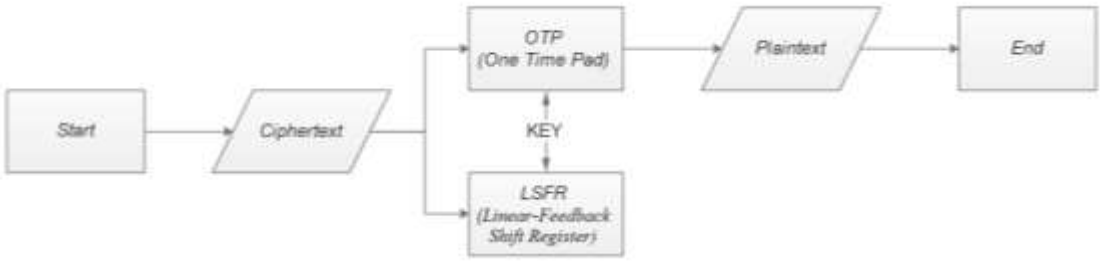



**Gambar 2.** Flowchart enkripsi *generate pseudo-random numbers* LSFR pada kunci OTP (2)

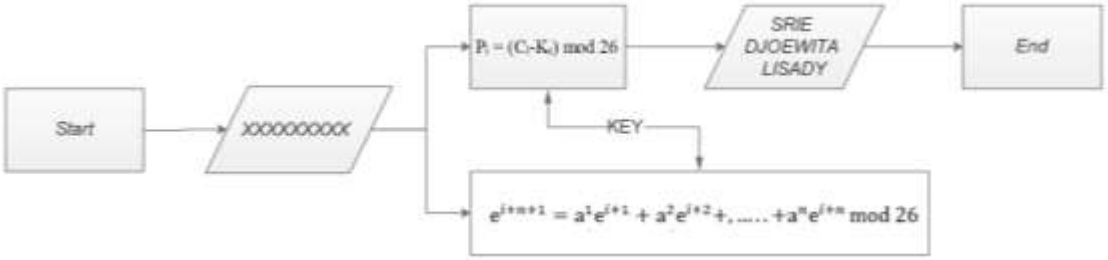
Untuk proses dekripsi merupakan proses *invers* dari *plaintext*. *Ciphertext* akan di dekripsi menggunakan kunci yang sama dari hasil sebelumnya yaitu hasil *generate pseudo-random numbers* melalui *Linear-Feedback Shift Register* (LSFR).

ISBN: 978-602-52720-7-3
Februari 2020
Hal 171-175

*ciphertext* akan di proses dengan menggunakan OTP dengan kunci yang diberikan dari LSFR sehingga menghasilkan *plaintext.*




**Gambar 3.** Flowchart dekripsi *generate pseudo-random numbers* LSFR pada kunci OTP (1)




**Gambar 4.** Flowchart dekripsi *generate pseudo-random numbers* LSFR pada kunci OTP (2)

## 3. ANALISA DAN PEMBAHASAN

*Plaintext* yang digunakan untuk pengoperasian OTP adalah "S R I E D J O E W I T A L I S A D Y" dengan kunci yang di *generate* melalui LSFR $e^1 = 5$ dan $e^2 = 10$. Adapun hasil proses enkripsi dan dekripsi dapat dilihat pada pembahasan berikut:

### 3.1 Proses Enkripsi

Untuk penentuan kunci dari *plaintext* = "S R I E D J O E W I T A L I S A D Y" dengan *18* karakter digunakan *Linear-Feedback Shift Register* (LSFR) melalui perhitungan berikut:
$e^1 = 5 \bmod 26$
$e^2 = 10 \bmod 26$
5 + (2*10) mod 26 = 25 mod 26
10 + (2*25) mod 26 = 8 mod 26
25 + (2*8) mod 26 = 15 mod 26
8 + (2*15) mod 26 = 12 mod 26
15 + (2*12) mod 26 = 13 mod 26
12 + (2*13) mod 26 = 12 mod 26
13 + (2*12) mod 26 = 11 mod 26
12 + (2*11) mod 26 = 8 mod 26
11 + (2*8) mod 26 = 1 mod 26
8 + (2*1) mod 26 = 10 mod 26
1 + (2*10) mod 26 = 21 mod 26
10 + (2*21) mod 26 = 0 mod 26
21 + (2*0) mod 26 = 21 mod 26
0 + (2*21) mod 26 = 16 mod 26
21 + (2*16) mod 26 = 2 mod 26
16 + (2*2) mod 26 = 20 mod 26
Hasil kunci dari LSFR: 5, 10, 25, 8, 15, 12, 13, 12, 11, 8, 1, 10, 21, 0, 21, 16, 2, 20
Setelah mendapatkan kunci melalui proses LSFR maka akan dilakukan proses OTP
S = 18 + 5 mod 26 = 23 mod 26
R = 17 + 10 mod 26 = 1 mod 26
I = 8 + 25 mod 26 = 7 mod 26
E = 4 + 8 mod 26 = 12 mod 26
D = 3 + 15 mod 26 = 18 mod 26
J = 9 + 12 mod 26 = 21 mod 26
O = 14 + 13 mod 26 = 1 mod 26
E = 4 + 12 mod 26 = 16 mod 26
W = 22 + 11 mod 26 = 7 mod 26
I = 8 + 8 mod 26 = 16 mod 26
T = 19 + 1 mod 26 = 20 mod 26

ISBN: 978-602-52720-7-3
Februari 2020
Hal 171-175

A = 0 + 10 mod 26 = 10 mod 26
L = 11 + 21 mod 26 = 6 mod 26
I = 8 + 0 mod 26 = 8 mod 26
S = 18 + 21 mod 26 = 13 mod 26
A = 0 + 16 mod 26 = 16 mod 26
D = 3 + 2 mod 26 = 5 mod 26
Y = 24 + 20 mod 26 = 18 mod 26

**Tabel 3.** Proses enkripsi OTP dengan LSFR

| *Plaintext* | *Numerical Plaintext* | *LSFR* $e^{i+n+1} = a^1e^{i+1} + a^2e^{i+2}+,\ldots.+a^ne^{i+n} \bmod 26$ | *Numerical Ciphertext* $(P_i+K_i) \bmod 26$ | *Ciphertext* |
|---|---|---|---|---|
| S | 18 | 5 | 23 | X |
| R | 17 | 10 | 1 | B |
| I | 8 | 25 | 7 | H |
| E | 4 | 8 | 12 | M |
| D | 3 | 15 | 18 | S |
| J | 9 | 12 | 21 | V |
| O | 14 | 13 | 1 | B |
| E | 4 | 12 | 16 | Q |
| W | 22 | 11 | 7 | H |
| I | 8 | 8 | 16 | Q |
| T | 19 | 1 | 20 | U |
| A | 0 | 10 | 10 | K |
| L | 11 | 21 | 6 | G |
| I | 8 | 0 | 8 | I |
| S | 18 | 21 | 13 | N |
| A | 0 | 16 | 16 | Q |
| D | 3 | 2 | 5 | F |
| Y | 24 | 20 | 18 | S |

Dari hasil perhitungan OTP dengan LSFR maka didapat:
*Plaintext:* SRIEDJOEWITALISADY
*Ciphertext:* XBHMSVBQHQUKGINQFS

### 3.2 Proses Dekripsi

Untuk proses dekripsi dari *ciphertext* X B H M S V B Q H Q U K G I N Q F S menggunakan kunci dari LSFR: 5, 10, 25, 8, 15, 12, 13, 12, 11, 8, 1, 10, 21, 0, 21, 16, 2, 20 dapat dilakukan proses perhitungan dekripsi OTP menggunakan kunci dari LSFR:
X = 23 - 5 mod 26 = 18 mod 26
B = 1 - 10 mod 26 = 17 mod 26
H = 7 - 25 mod 26 = 8 mod 26
M = 12 - 8 mod 26 = 4 mod 26
S = 18 - 15 mod 26 = 3 mod 26
V = 21 - 12 mod 26 = 9 mod 26
B = 1 - 13 mod 26 = 14 mod 26
Q = 16 - 12 mod 26 = 4 mod 26
H = 7 - 11 mod 26 = 22 mod 26
Q = 16 - 8 mod 26 = 8 mod 26
U = 20 - 1 mod 26 = 19 mod 26
K = 10 - 10 mod 26 = 0 mod 26
G = 6 - 21 mod 26 = 11 mod 26
I = 8 - 0 mod 26 = 8 mod 26
N = 13 - 21 mod 26 = 18 mod 26
Q = 16 - 16 mod 26 = 0 mod 26
F = 5 - 2 mod 26 = 3 mod 26
S = 18 - 20 mod 26 = 24 mod 26

**Tabel 4.** Proses dekripsi OTP dengan LSFR

| *Ciphertext* | *Numerical Ciphertext* | *LSFR* $e^{i+n+1} = a^1e^{i+1} + a^2e^{i+2}+,\ldots.+a^ne^{i+n} \bmod 26$ | *Numerical Plaintext* $(P_i-K_i) \bmod 26$ | *Plaintext* |
|---|---|---|---|---|
| X | 23 | 5 | 18 | S |
| B | 1 | 10 | 17 | R |

ISBN: 978-602-52720-7-3
Februari 2020
Hal 171-175

| | | | | |
|---|---|---|---|---|
| H | 7 | 25 | 8 | I |
| M | 12 | 8 | 4 | E |
| S | 18 | 15 | 3 | D |
| V | 21 | 12 | 9 | J |
| B | 1 | 13 | 14 | O |
| Q | 16 | 12 | 4 | E |
| H | 7 | 11 | 22 | W |
| Q | 16 | 8 | 8 | I |
| U | 20 | 1 | 19 | T |
| K | 10 | 10 | 0 | A |
| G | 6 | 21 | 11 | L |
| I | 8 | 0 | 8 | I |
| N | 13 | 21 | 18 | S |
| Q | 16 | 16 | 0 | A |
| F | 5 | 2 | 3 | D |
| S | 18 | 20 | 24 | Y |

Dari hasil perhitungan OTP dengan LSFR maka didapat:
*Ciphertext:* XBHMSVBQHQUKGINQFS
*Plaintext:* SRIEDJOEWITALISADY

## 4. KESIMPULAN

Algoritma keamanan pesan *One Time Pad* (OTP) memang memiliki kemampuan yang handal, tetapi juga memiliki kelemahan yang sangat membebani pengguna. Hal ini disebabkan karena jumlah karakter pada *plaintext* harus sama dengan jumlah karakter yang ada pada kunci atau *key*. Penggunaan *Generate Pseudo-Random Numbers Linear-Feedback Shift Register* (LSFR) dalam menemukan kunci melalui *plaintext* dapat mengatasi permasalahan ini sehingga kunci akan otomatis terbentuk dari banyaknya *plaintext* yang ada.

## REFERENCES

[1] R. E. Blahut, *Cryptography and Secure Communication*, vol. 9781107014. Cambridge: Cambridge University Press, 2014.
[2] G. Hao and G. Tao, "Principle of and Protection of Man-in-the-middle Attack Based on ARP Spoofing," vol. 5, no. 3, pp. 131–134, 2009.
[3] K. M. Jain, M. V Jain, and J. L. Borade, "A Survey on Man in the Middle Attack," *IJSTE-International J. Sci. Technol. Eng.* |, vol. 2, no. 09, pp. 277–280, 2016.
[4] F. Diani and Y. Widhiyasana, "Enkripsi SMS dengan Menggunakan One Time Pad ( OTP ) dan Kompresi Lempel-Ziv-Welch ( LZW )," vol. 7, no. 3, pp. 3–8, 2018.
[5] Zaeniah and B. E. Purnama, "An Analysis of Encryption and Decryption Application by using One Time Pad Algorithm," *Int. J. Adv. Comput. Sci. Appl.*, vol. 6, no. 9, pp. 292–297, 2015.
[6] D. Sharma, A. Khalid, and S. Parashar, "Cryptographically Secure Linear feedback shift Register," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 3, no. 10, pp. 3504–3507, 2014.
[7] M. Kalyvas, K. Yiannopoulos, T. Houbavlis, and H. Avramopoulos, "Design algorithm of all-optical linear feedback shift registers," *AEU - Int. J. Electron. Commun.*, vol. 57, no. 5, pp. 328–332, 2003.
[8] T. B. Brock, "Linear Feedback Shift Registers and Cyclic Codes in SAGE 1 Introduction," vol. 7, no. 2, pp. 1–26, 2006.
[9] B. Ndaw, D. Sow, and M. Sanghare, "Construction of Maximum Period Linear Feedback Shift Registers (LFSR) (Primitive Polynomials and Linear Recurring Relations)," *Br. J. Math. Comput. Sci.*, vol. 11, no. 4, pp. 1–24, 2015.
[10] O. K. Sulaiman, K. Nasution, and S. Y. Prayogi, "Enkripsi Surat Elektronik Menggunakan Metode XXTEA," *Comput. Eng. Sci. Syst. J.*, vol. 4, no. 1, p. 99, Jan. 2019.
[11] W. H. Hutagalung, "Implementasi Vigenere Chiper Dengan Random Key Metode Linear Feedback Shift Register ( LFSR ) Pada Teks," *Maj. Ilm. INTI*, vol. 12, pp. 58–61, 2013.