PENGEMBANGAN ALGORITMA ENKRIPSI DEKRIPSI BERBASIS LFSR MENGGUNAKAN POLINOMIAL PRIMITIF

PUBLIKASI HASIL PENELITIAN SKRIPSI

Diajukan Untuk Memenuhi Sebagian Persyaratan
Memperoleh Gelar Sarjana Teknik

DISUSUN OLEH:
ANGGUN TRIYOGO
NIM. 0710630016-63

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS BRAWIJAYA
MALANG

2013

	KEMENTRIAN PENDIDIKAN DAN KEBUDAYAAN UNIVERSITAS BRAWIJAYA FAKULTAS TEKNIK JURUSAN TEKNIK ELEKTRO Jalan MT Haryono 167 Telp& Fax. 0341 554166 Malang 65145	**KODE** **PJ-01**

PENGESAHAN
PUBLIKASI HASIL PENELITIAN SKRIPSI
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS BRAWIJAYA

NAMA : ANGGUN TRIYOGO

NIM : 0710630016

PROGRAM STUDI : TEKNIK REKAYASA KOMPUTER

JUDUL SKRIPSI : PENGEMBANGAN ALGORITMA ENKRIPSI DEKRIPSI BERBASIS LFSR MENGGUNAKAN POLINOMIAL PRIMITIF

TELAH DI-REVIEW DAN DISETUJUI ISINYA OLEH:

Pembimbing 1	Pembimbing 2
Waru Djuriatno ST., MT	**Adharul Muttaqin, ST., MT., NIP.**
NIP.19690712199702	**19760121 200501 1 001**

PENGEMBANGAN ALGORITMA ENKRIPSI DEKRIPSI BERBASIS LFSR MENGGUNAKAN POLIMOMIAL PRIMITIF

Anggun Triyogo – NIM 0710630016
Program Studi Teknik Elektro
Fakultas Teknik
Universitas Brawijaya, Jl. Mayjend. Haryono no.167,Malang,65145,Indonesia
angguneletro@yahoo.com

***Abstrak— Stream chiper* adalah salah satu metode kriptografi modern yang populer karena di samping prosesnya yang memakan waktu lebih singkat, *stream chiper* juga menggunakan memori yang lebih sedikit.LFSR (*Linear Feedback Shift Register*) adalah salah satu jenis generator yang dapat menghasilkan bit semu acak. LFSR sering digunakan karena mampu menghasilkan bit semu acak dengan periode maksimal yang panjang dan mudah diaplikasikan dalam berbagai hal. Namun seiring dengan perkembangan zaman penggunaan sebuah LFSR sebagai generator bit semu acak rawan terhadap serangan kriptanalisis.Dalam makalah ini penulis akan membahas metode pengembangan n buah generator LFSR dengan teknik multipleksing dan melakukan studi analisis performanya dengan metode pengujian statistik.**

***Kata kunci*— pembangkit bilangan acak, kriptografi, PRNG, LFSR, pengujian statistik.**

I. PENDAHULUAN

Bilangan acak merupakan salah satu faktor yang sangat penting dalam kriptografi.Hal ini disebabkan karena bilangan acak menjadi dasar perhitungan dalam kriptografi, yang kemudian menentukan kekuatan dari kriptografi itu sendiri. Dalam kriptografi kunci simetri stream cipher,kekuatan algoritma stream cipher terletak pada keacakan rangkaian bit semu acak yang dihasilkan bukan tergantung pada kerahasiaan algoritmanya.

Tidak ada cara konvensional yang bisa benar-benar menghasilkan deret bilangan acak secara sempurna. Umumnya cara yang digunakan dalam membangkitkan bilangan acak adalah berdasarkan suatu algoritma atau fungsi tertentu yang deterministik, sehingga sebenarnya bilangan yang dibangkitkan tersebut bersifat pseudorandom, karena pembangkitan bilangannya dapat diulang kembali. Apabila bilangan acak yang menjadi dasar dalam kriptografi tersebut bersifat pseudorandom, akan memudahkan bagi kriptanalis untuk memecahkan enkripsi / dekripsi. Oleh karena itu, dibutuhkan suatu pembangkit bilangan yang benar-benar acak.

LFSR (*Linear Feedback Shift Register*) adalah salah satu jenis generator yang dapat menghasilkan bit semu acak. LFSR sering digunakan karena mampu menghasilkan bit semu acak dengan periode maksimal yang panjang dan mudah diaplikasikan dalam berbagai hal. Namun seiring dengan perkembangan zaman penggunaan sebuah LFSR sebagai generator bit semu acak rawan terhadap serangan kriptanalisis.

Pada perkembangan selanjutnya dilakukan penggabungan beberapa buah LFSR dalam proses enkripsi dekripsi untuk meningkatkan kekuatan algoritma enkripsi dekripsi tersebut. Penggunaan beberapa buah LFSR dalam proses enkripsi dekripsi ini sangat menarik dan perlu dianalisa lebih lanjut.

II. PEMBANGKIT BILANGAN ACAK SEMU

Pembangkit Bilangan Acak Semu, atau PRNG (*Psedo Random Number Generator)*,adalah generator penghasil bilangan, dimana bilangan yang dibangkitkannya tidak benar-benar acak, hanya kelihatannya saja acak, Hal ini dikarenakan untuk menghasilkan bilangan acak merupakan hal yang sulit. Bilangan yang dihasilkan oleh Pembangkit Bilangan Acak Semu selalu memilki perulangan pola tertentu pada periodenya.Semua deretan bilangan acak yang dibangkitkan dari rumus matematika, serumit apapun, dianggap sebagai deret acak semu, karena dapat diulang pembangkitannya.Dalam dunia kriptografi terdapat beberapa algoritma Pembangkit Bilangan Acak Semu yang dapatdigunakan untuk membangkitkan deretan bilangan acak semu. Berikut adalah beberapa algoritmayang sering digunakan:

1. *Linear Feedback Shift Register* (LFSR).

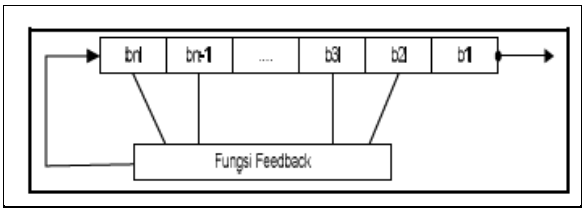
2. *Non Linear Feedback Shift Register* (NLFSR).
3. *Indirection, Shift, Accumulate, Add and Count* (ISAAC).
4. *Lagged Fibonacci Generator* (LFG).
5. *Mersenne Twister*.
6. *Fortuna*.
7. *Blum-Blum Shub*.

III. LINEAR FEEDBACK SHIFT REGISTER(LFSR)

Linear Feedback Shift Register (LFSR) adalah suatu mekanisme untuk menghasilkan sekuens bit biner. *Register* memiliki sebarisan sel yang ditentukan oleh vektor inisialisasi yang biasanya menjadi kunci rahasia. Tingkah laku *register* diatur oleh sebuah *counter* (*clock*). Pada setiap saat isi sel dari register digeser (*shift*) ke kanan sejauh satu posisi, dan hasil operasi XOR terhadap subset dari isi sel ditempatkan pada sel paling kiri.
Register geser umpan balik ini (Feedback Shift Register) mempunyai 2bagian yang sangat penting :

1. Register Geser
Yaitu barisan bit bit (bn, bn-1, bn-2, ... b4,b3,b2,b1) yang panjangnya n (disebut juga register geser n- bit)

2. Fungsi Umpan Balik
Yaitu suatu fungsi yang menerima masukan dari register geser danmengembalikan nilai fungsi ke register geser.

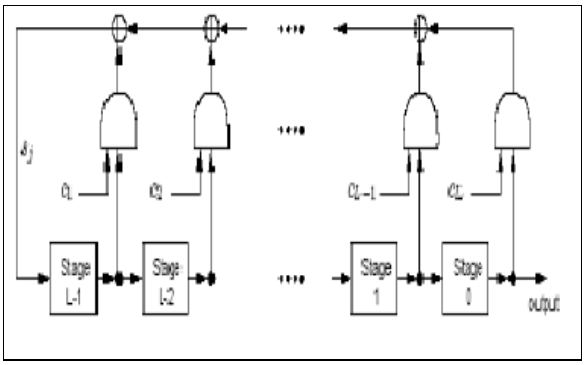


Gambar 1Linear Feedback Shift Register

Tiap kali sebuah bit dibutuhkan, semua bit didalam register digeser 1 bit ke kanan. Bit paling kiri (bn) dihitung sebagai fungsi bit bit lain di dalam register tersebut. Keluaran dari register geser adalah 1 bit (yaitu bit b1 yang tergeser). Periode register geser adalah 1bit (yaitu bit b1 yang tergeser). Periode register geser adalah panjang keluaran sebelum ia berulang kembali.LFSR dengan panjang L terdiri dari L tahap yang diberi nomor 0, 1, 2, 3, ... , L-1.Masing-masing tahap menyimpan satu bit yang diperoleh dari hasil masukan satu bit tahap sebelumnya, dan selanjutnya akan memberikan satu bit sebelum diganti miliknya ke tahap berikutnya; dan menggunakan kendali jam sebagai pengatur pergerakan data. Selama satu satuan waktu LFSR akan melakukan operasi berikut:

1. Isi dari tahap 0 adalah keluaran yang akan digunakan untuk membentuk rangkaian bit kunci.
2. Isi pada tahap i akan dipindahkan ke tahap i-1 untuk setiap i, $1 \leq i \leq L-1$.
3. Isi dari tahap L – 1 adalah bit umpan balik Sj yang diperoleh dari hasil penambahan isi tahap sebelumnya (0, 1, 2, ... , L - 1) dan hasilnya dimod 2.

Berikut menggambarkan skema LFSR yang memiliki panjang L.



Gambar 2LFSR dengan panjang L

LFSR di atas dapat dilambangkan dengan (L,C(D)), dengan $C(D) = {}_{c0}D^0 + {}_{c1}D^1 + {}_{c2}D^2 + \ldots + {}_{cL}D^L$,elemen dari $Z_2[D]$ adalah polinomial penghubung.

$$C(D) = {}_{c0}D^0 + {}_{c1}D^1 + {}_{c2}D^2 + \ldots + {}_{cL}D^L, \text{elemen dari} Z_2[D]$$

Gambar 3Fungsi Feedback LFSR

Jika status inisial LFSR pada Gambar 2.11 di atas adalah $[s_0, s_1 \ldots, s_{L-1}]$, maka alirankeluaran s = s0, s1, s2, ... ditentukan oleh rumus :

$$s_{k+L} = c_0 s_k + c_1 s_{k+1} + c_2 s_{k+2} + c_3 s_{k+3} + \ldots + c_{(L-1)} s_{k+(L-1)}$$

Gambar 4Output bit LFSR

Jika C(D) merupakan polinomial primitif, maka untuk masing-masing $2^L - 1$ non nol status inisial sebuah *non singular* LFSR (L,C(D)) akan menghasilkan aliran output dengan periode maksimum yang mungkin sebesar $2^L - 1$ (Menezes *et*

al. 1996). Polinomial primitif merupakan polynomial berderajat n yang tidak dapat direduksi.

III. POLINOMIAL PRIMITIF

Penggunaan jenis polinomial dalam LFSR akan mempengaruhi hasil bit semu acak yang dihasilkan oleh LFSR. Polinomial berfungsi sebagai fungsi *feedback* dari LFSR.Polinomial primitif adalah polinomial berderajat n yang tidak dapat direduksi.Terdapat dua jenis polinomial primitif yaitu yaitu *sparse* dan *dense*.Polinomial primitif disebut *sparse* jika hanya memiliki sedikit koefisien tidak nol, sedangkan polinomial primitif disebut *dense* jika memiliki banyak koefisien tidak nol. Untuk aplikasi kriptografi, sebaiknya digunakan polinomial primitif yang bersifat *dense*.Berikut adalah contoh polinomial primitif dari derajat tinggi hingga derajat rendah:

$x^{31} + x^{30} + x^{29} + x^{28} + x^{27} + x^{26} + x^{25} + x^{24} + x^{23} + x^{22} + x^{21} + x^{20} + x^{19} + x^{18} + x^{17} + x^{16} + x^{15} + x^{14} + x^{13} + x^{12} + x^{11} + x^{10} + x^{9} + x^{8} + x^{7} + x^{6} + x^{5} + x^{4} + x^{2} + x^{1} + 1$, adalah contoh polinomial primitif *dense* derajat tinggi, dan $x^{31} + 1$, adalah contoh polinomial *sparse* derajat tinggi.

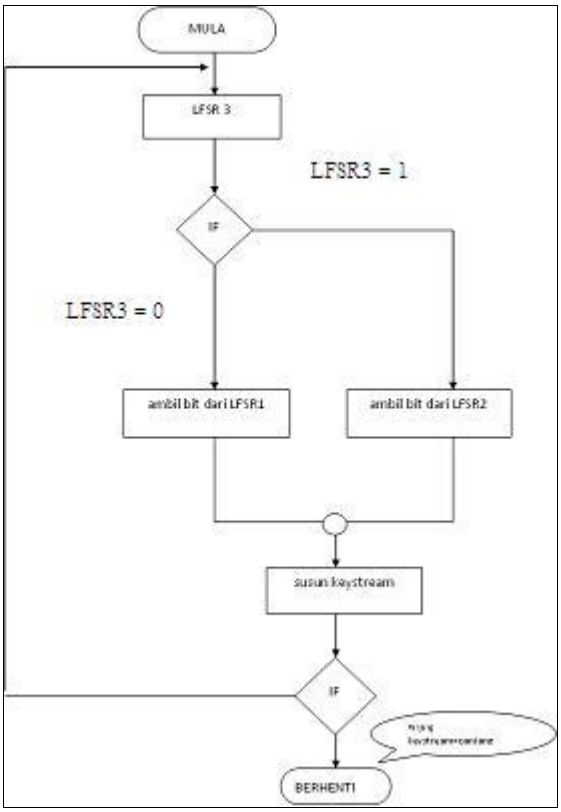
$x^{15} + x^{14} + x^{13} + x^{12} + x^{11} + x^{10} + x^{9} + x^{8} + x^{7} + x^{6} + x^{5} + x^{4} + x^{3} + x^{2} + 1$, adalah contoh polinomial primitif *dense* derajat sedang, dan $x^{15} + x^{1} + 1$,adalah polinomial primitif *sparse* derajat sedang.

$x^{2} + x^{1} + 1$, adalah polinomial primitif derajat rendah, $x^{15} + 1$ dan $x^{2} + 1$, adalah contoh polinomial non primitif.

IV. MULTI REGISTER

Multi Register adalah gabungan dari beberapa register dengan menggunakan operasi Boolean.Kombinasi biasa dilakukan dengan operasi XOR, fungsi perkalian, JK-Flip flop, multiplexing, atau MM movement. Kombinasi multi LFSR bertujuan untuk meningkatkan sifat non linearitas sebuah rangkaian bit semu acak.

Multipleksing register adalah salah satu cara melakukan proses multi register. Untuk melakukan multipleksing LFSR diperlukan minimal dua buah LFSR. Berikut adalah contoh perancangan multipleksing LFSR menggunakan tiga buah generator LFSR:



Gambar 5Multipleksing dengan 3 LFSR

Apabila bit dari LFSR nomor 3 bernilai 0, maka bit penyusun bit keluaran hasil multipleksing diambil dari bit LFSR nomor 1, dan apabila bit LFSR nomor 3 bernilai 1, maka bit keluaran hasil multipleksing diambil dari LFSR nomor 2. Proses tersebut akan berulang sampai jumlah bit keluaran hasil multipleksing bias dipakai untuk mengenkripsi *plaintext.*

IV. EKSPERIMEN PENGUJIAN

A. Metode Pengujian

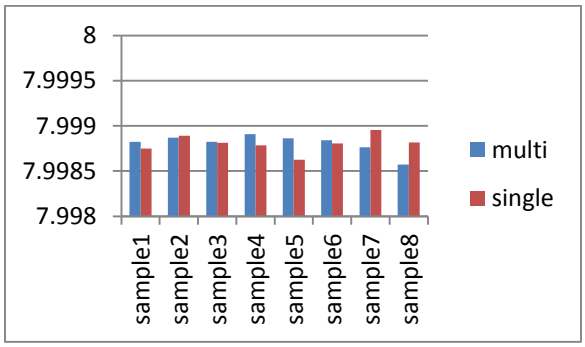
Pengujian dilakukan dengan melakukan uji statistik serangkaian bit semu acak yang dihasilkan generator multipleksing LFSR. Selanjutnya untuk melakukan tes kelayakan dilakukan tes uji statistik beberapa ciphertext yang dihasilkan oleh generator multipleksing LFSR.Program yang digunakan adalah program ENT yang dibuat oleh John Walker. Progam ini dapat mengukur nilai-nilai ukuran keacakan sekuens byte sebuah file *binary* berdasarkan teknik pengujian keacakan standar.

B. Hasil Pengujian dan Pembahasan

Hasil pengujian adalah sebagai berikut, dikelompokkan berdasarkan jenis tes yang dilaksanakan.

Entropy

Adalah nilai kepadatan informasi suatu file, yang diekspresikan sebagai jumlah bit per karakter. Range entropi adalah dari 0 hingga 8. Data yang acak akan memilki nilai entropi yang tinggi. Semakin acak suatu bilangan nilai entropinya akan semakin mendekati nilai 8. Hasil pengujian adalah sebagai berikut:

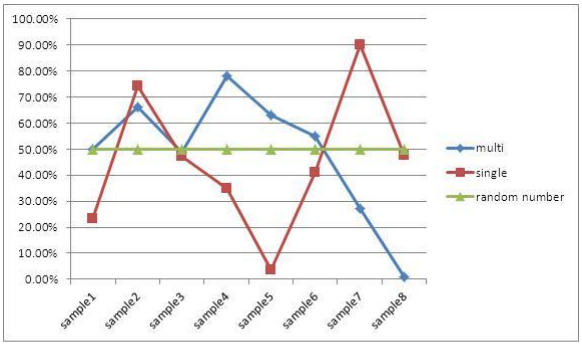


Gambar 6Hasil Entropi Cipherteks Multipleksing LFSR

Dari hasil tes terlihat bahwa semua sampel multipleksing LFSR menghasilkan sekuens byte dengan entropi yang baik.Hampir keseluruhan sampel memiliki nilai entropi yang mendekati nilai 8.

Chi-square Test

Pengujian Chi-square adalah tes yang paling umum digunakan untuk menguji keacakan data, dan sangat sensitif pada adanya error dalam RNG. Distribusi *Chi-square* dikalkulasikan pada sekuens byte dan akan menghasilkan sebuah nilai absolute dan persentase seberapa sering sekuens bilangan acak sejati akan melewati nilai tersebut. Persentase ini diinpretasikan apakah data patut dicurigai keacakannya atau tidak.Sekuens yang menghasilkan persentase lebih dari 99% atau kurang dari 1% adalah hampir pasti tidak acak. Persentase antara 99%-95% dan 1%-5% menunjukan sekuens patut dicurigai tidak acak, dan persentase antara 90%-95% serta 5%-10% mengindikasikan bahwa sekuens ini „hampir dicurigai". Hasil pengujian adalah sebagai berikut:

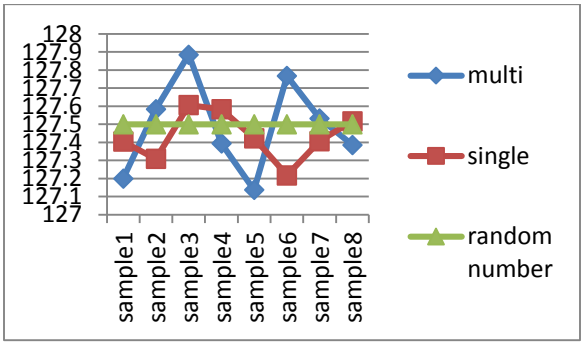


Gambar 7Hasil Chi SquareCipherteks Multipleksing LFSR

Tampak bahwa dari 8 kali sampel cipherteks,7 di antaranya lolos uji *chi square test*dengan predikat acak. Nilai uji terbaik terjadi di sampel no 3 dengan persentase nilai *chi square test* mendekati nilai 50%.

Arithmetic Mean

Nilai ini dihasilkan dengan menjumlahkan semua byte dalam data dan membaginya dengan panjang data. Jika data mendekati random, nilainya akan mendekati 127.5. Kualitas dapat dilihat dari seberapa besar deviasi nilai terhadap 127.5 Hasil pengujian adalah sebagai berikut:



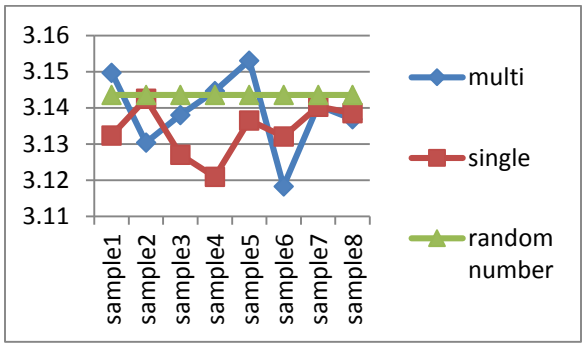
Gambar 8Hasil ArithmeticMean Cipherteks Multipleksing LFSR

Dari hasil tes terlihat bahwa semua sampel multipleksing LFSR menghasilkan sekuens byte dengan nilai *arithmetic mean* yang mendekati nilai 127.5, yaitu berada pada range 127.1 – 127.9

Monte Carlo Value for Pi

Setiap sekuens sepanjang 6 byte digunakan sebagian kordinat X dan Y 24 bit dalam sebuah bujur sangkar. Apabila jarak dari titik yang dihasilkan secara random lebih kecil dari setengah panjang sisi bujursangkar, sekuens ini dinyatakan "kena". Persentase titik yang "kena" dapat digunakan untuk menghitung nilai π. Untuk data yang besar, nilai yang

dihasilkan akan mendekati nilai π sebenarnya jika data tersebut bersifat acak. Suatu bilangan dikatakan akan mendekati acak apabila nilai *Monte Carlo*nya mendekati *Monte Carlo*500000 byte file yang diciptakan oleh peluruhan radioaktif yang nilainya adalah 3.143580574 (0.06 persen *error*).Berikut adalah hasil uji *Monte Carlo* dari multipleksing LFSR.

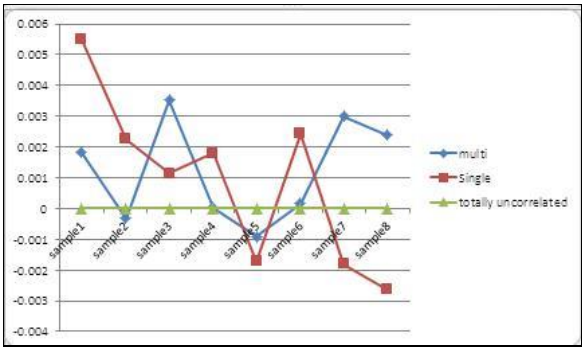


Gambar 9Hasil Monte Carlo CipherteksMultipleksing LFSR

Dari hasil tes uji *Monte Carlo* menunjukkan bahwa cipherteks hasil multipleksing LFSR memiliki nilai *Monte Carlo*yang mendekati 3.14 yaitu antara 3.118 – 3.153.

Serial Correlation Coefficient

Nilai ini mengukur seberapa besar pengaruh suatu byte pada byte setelahnya. Untuk data yang acak, nilai ini akan mendekati nol. Berikut adalah hasil pengujian, berupa nilai selisihnya dari nol:



Gambar 10Hasil Monte Carlo CipherteksMultipleksing LFSR

Tes uji *Serial Correlation* menunjukkan bahwa dari keseluruhan sampel tidak ada yang memiliki nilai *Correlation Coefficient*yang melebihi 0.1.Hal ini menunjukkan bahwa cipherteks multipleksing LFSR lolos uji *Serial Correlation Coefficient*.

V. KESIMPULAN

Cipherteks hasil enkripsi menggunakan generator multipleksing LFSR berhasil lolos uji statistik menggunakan program ENT dan memiliki *coefficient correlation* yang bagus.

Dari beberapa kali percobaan, didapatkan hasil bahwa *initial state* dari masing masing LFSR memberikan pengaruh yang besar terhadap nilai statistik bilangan semu acak yang akan dihasilkan.

Cipherteks hasil enkripsi menggunakan algoritma multipleksing LFSR masih belum bisa dikatakan sepenuhnya aman secara statistik karena plainteks juga mempengaruhi nilai statistik yang akan dihasilkan oleh cipherteks.Namun secara umum algoritma multipleksing LFSR masih layak dipakai dalam kebutuhan kriptografi dimana kualitas keacakan tidak menjadi masalah utama.

Penulis masih menganjurkan penggunaan multi LFSR dibandingkan single LFSR karena meskipun keduanya masih belum bisa dikatakan sepenuhnya aman secara statistik,namun kriptanalisis yang ditujukan untuk menyerang single LFSR akan lebih sulit diterapkan pada multi LFSR.

REFERENSI

Menezes,AlfredJ.*HandbookofAppliedCryptograph*. 1994. New York:CRC Press

http://www.theory.cs.uvic.ca/~cos/gen/poly.html

Schneier, Bruce (1996). *Applied Cryptography Second Edition: Protocols, Algorithms and Source Code in C*, John Wiley & sons, Inc.