JURNAL FOURIER | Oktober 2026, Vol. 15, No. 2, 63-77 ISSN 2252-763X
DOI: 10.14421/fourier.2026.152.63-77 E-ISSN 2541-5239

Implementasi *Linear Congruential Generator* dan Algoritma Fisher Yates Shuffle dalam Playfair Cipher untuk Pengamanan Pesan Teks

Aurellisa Maira Sari[1], Rini Marwati[1], Ririn Sispiyati[1]

[1]*Program Studi Matematika, Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam, Universitas Pendidikan Indonesia*
Jalan. Dr. Setiabudi No.229, Bandung 40154, Indonesia

Korespondensi; Rini Marwati, Email: rinimarwati@upi.edu

Abstrak

Perkembangan teknologi informasi mendorong meningkatnya kebutuhan akan sistem pengamanan pesan teks dalam komunikasi digital. Meskipun Playfair Cipher sebagai salah satu metode kriptografi klasik berbasis *digraph* memiliki tingkat kompleksitas kunci yang tinggi, metode ini masih memiliki keterbatasan dalam variasi matriks kunci dan potensi analisis pola bigram apabila digunakan tanpa modifikasi tambahan. Oleh karena itu, penelitian ini mengusulkan integrasi *Linear Congruential Generator* (LCG) sebagai pembangkit bilangan acak semu dan algoritma Fisher Yates Shuffle sebagai mekanisme pengacakan alfabet secara merata dalam pembentukan matriks kunci Playfair Cipher. LCG digunakan untuk menghasilkan deret bilangan acak semu berdasarkan parameter tertentu dan nilai awal (*seed*), yang selanjutnya dimanfaatkan dalam proses Fisher Yates Shuffle untuk mengacak alfabet A–Z tanpa huruf J sebelum dikombinasikan dengan prefix key guna membentuk matriks Playfair berukuran 5×5. Program aplikasi dikonstruksi menggunakan bahasa pemrograman Python dengan antarmuka *Graphical User Interface* (GUI) untuk mendukung proses enkripsi dan dekripsi secara efektif. Hasil penelitian menunjukkan bahwa integrasi LCG dan Fisher Yates Shuffle menghasilkan variasi matriks yang lebih dinamis dibandingkan Playfair klasik dengan matriks statis. Selain itu, hasil enkripsi dan dekripsi yang diperoleh melalui program telah tervalidasi sesuai dengan perhitungan manual, sehingga sistem yang dikembangkan mampu meningkatkan kompleksitas kunci serta mengurangi pola penyandian dalam pengamanan pesan teks.

Kata Kunci: Fisher Yates Shuffle, Kriptografi Klasik, Linear Congruential Generator (LCG), Pengamanan Pesan Teks, Playfair Cipher.

Abstract

The rapid development of information technology has increased the need for secure text message systems in digital communication. Although the Playfair Cipher, as a classical digraph-based cryptographic method, offers high key complexity, it still has limitations in key matrix variation and vulnerability to bigram pattern analysis when used without modification. Therefore, this study proposes the integration of the Linear Congruential Generator (LCG) as a pseudo-random number generator and the Fisher Yates Shuffle algorithm as a uniform alphabet randomization mechanism in constructing the Playfair Cipher key matrix. The LCG is utilized to generate a sequence of pseudo-random numbers based on specific parameters and an initial seed value, which are then applied in the Fisher–Yates Shuffle process to randomize the alphabet A–Z without the letter J before being combined with a prefix key to form a 5×5 Playfair matrix. The application program is developed using Python with a Graphical User Interface (GUI) to facilitate encryption and decryption processes effectively. The results show that the integration of LCG and Fisher–Yates Shuffle produces a more dynamic key matrix variation compared to the classical static Playfair matrix. Furthermore, the encryption and decryption results generated by the program have been validated through manual calculations, indicating that the developed system enhances key complexity and reduces predictable encryption patterns in text message security.

Keywords: Classical Cryptography, Fisher Yates Shuffle, Linear Congruential Generator (LCG), Playfair Cipher, Text Message Security.

© 2026 JURNAL FOURIER

Pendahuluan

Perkembangan teknologi informasi yang pesat telah meningkatkan kebutuhan akan sistem pengamanan data, khususnya dalam komunikasi digital. Pesan teks yang dikirim melalui jaringan rentan terhadap penyadapan dan penyalahgunaan oleh pihak yang tidak berwenang, sehingga berpotensi menimbulkan kebocoran informasi [1], [2]. Oleh karena itu, diperlukan suatu mekanisme pengamanan yang mampu menjaga kerahasiaan data. Kriptografi merupakan salah satu metode yang digunakan untuk melindungi informasi dengan cara mengubah pesan menjadi bentuk yang tidak dapat dipahami tanpa kunci tertentu [3]. Untuk mencapai tujuan tersebut, dikembangkan metode penyandian berupa proses enkripsi dan dekripsi terhadap data yang ingin dilindungi [4].

Salah satu teknik kriptografi klasik untuk merahasiakan pesan adalah substitution cipher, yang berbasis pada konsep permutasi. Dengan jumlah kemungkinan kunci sebesar 26!, metode ini secara teoritis sulit diretas menggunakan pendekatan brute force. Namun, metode ini masih dapat dianalisis menggunakan distribusi frekuensi huruf [5]. Metode lain yang lebih kompleks adalah Playfair Cipher, yang merupakan teknik digraph substitution cipher yang mengenkripsi pasangan huruf berdasarkan posisi dalam matriks kunci. Dengan ruang kunci sebesar 25!, metode ini memiliki tingkat kompleksitas yang lebih tinggi dibandingkan cipher monoalfabetik, sehingga lebih sulit dikriptanalisis secara manual [6].

Meskipun demikian, Playfair Cipher masih memiliki keterbatasan, terutama pada variasi matriks kunci yang terbatas serta potensi analisis pola bigram apabila digunakan tanpa modifikasi tambahan [7]. Beberapa penelitian telah dilakukan untuk meningkatkan keamanan metode ini. Penelitian sebelumnya menunjukkan bahwa modifikasi Playfair Cipher dapat memengaruhi performa enkripsi dan dekripsi, khususnya dalam hal efisiensi waktu [8]. Selain itu, pengembangan lain dilakukan dengan memperluas ukuran matriks kunci dan menggabungkannya dengan algoritma pembangkit bilangan acak, sehingga mampu meningkatkan kompleksitas dan mengurangi pola yang mudah dianalisis [6].

Penggunaan pembangkit bilangan acak semu (Pseudo Random Number Generator/PRNG) seperti Linear Congruential Generator (LCG) telah banyak dimanfaatkan dalam berbagai aplikasi untuk menghasilkan nilai acak [9]. Selain itu, penelitian lain juga menunjukkan bahwa kombinasi teknik kriptografi dengan pembangkit bilangan acak semu mampu meningkatkan keamanan data, seperti pada pengamanan file audio menggunakan algoritma Blum Blum Shub [10]. Namun, kualitas bilangan acak semu dari LCG masih memiliki keterbatasan apabila tidak diolah lebih lanjut [11].

Berdasarkan penelitian-penelitian sebelumnya, pengembangan Playfair Cipher umumnya dilakukan melalui modifikasi matriks kunci atau penggabungan dengan pembangkit bilangan acak semu. Namun, integrasi LCG dan algoritma Fisher–Yates Shuffle secara khusus dalam proses pembentukan matriks kunci Playfair Cipher untuk pengamanan pesan teks masih belum banyak dikaji [12][13].

Oleh karena itu, penelitian ini mengusulkan pendekatan dengan mengombinasikan LCG sebagai sumber bilangan acak semu dan algoritma Fisher–Yates Shuffle sebagai mekanisme pengacakan alfabet secara merata dalam pembentukan matriks kunci Playfair Cipher. Pendekatan ini bertujuan untuk menghasilkan variasi matriks yang lebih dinamis, meningkatkan kompleksitas kunci, serta mengurangi pola penyandian yang mudah diprediksi. Hasil penelitian menunjukkan bahwa metode yang diusulkan mampu menghasilkan sistem pengamanan pesan teks yang lebih efektif dan tervalidasi melalui perhitungan manual serta implementasi program.

Landasan Teori

Penelitian ini menggunakan beberapa konsep dasar matematika sebagai landasan dalam implementasi algoritma yang digunakan, yaitu aritmatika modular, permutasi dan kombinasi, serta matriks. Aritmatika modular merupakan operasi yang menghasilkan sisa pembagian suatu bilangan terhadap

bilangan lain dan menjadi dasar dalam proses pembangkitan bilangan acak semu pada algoritma *Linear Congruential Generator* (LCG), dengan operasi modulo didefinisikan sebagai

$$a \bmod m = r, 0 \leq r < m! \quad (1)$$

serta memenuhi sifat penjumlahan, pengurangan, dan perkalian dalam sistem modulo [14], [15]. Selanjutnya, konsep permutasi digunakan untuk menyusun objek dengan memperhatikan urutan, sedangkan kombinasi digunakan untuk memilih objek tanpa memperhatikan urutan, yang masing-masing dinyatakan dengan rumus sebagai berikut [16]

$$P(n) = n! \quad (2)$$

$$P(n,r) = \frac{n!}{(n-r)!} \quad (3)$$

$$C(n,r) = \frac{n!}{r!(n-r)!} \quad (4)$$

di mana:
n: jumlah total objek
r: jumlah objek yang dipilih

Dalam penelitian ini, permutasi diterapkan pada proses pengacakan alfabet menggunakan algoritma Fisher–Yates Shuffle, sedangkan kombinasi digunakan secara implisit dalam pembentukan pasangan huruf (*digraph*) pada proses enkripsi dan dekripsi. Selain itu, matriks digunakan sebagai dasar pembentukan tabel kunci pada algoritma Playfair Cipher, di mana susunan elemen dalam matriks menentukan proses substitusi pasangan huruf. Bentuk umum matriks berukuran $m \times n$

Linear Congruential Generator (LCG)

LCG merupakan metode pembangkit bilangan acak semu yang memerlukan beberapa parameter utama, yaitu nilai pengali a, nilai penambah b, modulus m, dan nilai awal (*seed*/X_0). Berdasarkan parameter tersebut, LCG didefinisikan sebagai berikut [16]:

$$X_{k+1} = (aX_k + b) \bmod m \quad (5)$$

di mana
X_{k+1} : bilangan acak ke-$k+1$
X_k : bilangan acak ke-k
m : modulus
a : faktor pengali
b : penambah

Dalam penelitian ini, LCG digunakan untuk menghasilkan bilangan acak semu yang akan digunakan sebagai dasar pengacakan pada algoritma Fisher–Yates Shuffle.

Fisher Yates Shuffle

Algoritma Fisher Yates Shuffle digunakan untuk melakukan pengacakan elemen secara merata terhadap alfabet yang akan digunakan dalam pembentukan matriks Playfair Cipher. Proses pengacakan pada Fisher–Yates Shuffle dilakukan secara bertahap dengan menukar posisi elemen berdasarkan indeks acak yang dihasilkan. Pada setiap iterasi, sebuah indeks acak dipilih dari rentang indeks yang tersedia, kemudian elemen pada indeks tersebut ditukar dengan elemen pada posisi terakhir dari rentang yang belum diacak. Proses ini diulang hingga seluruh elemen dalam daftar mengalami pertukaran dan tidak ada elemen yang tersisa untuk diproses.

Dalam penelitian ini, nilai indeks acak yang digunakan pada setiap proses pertukaran diperoleh dari deret bilangan acak semu yang dihasilkan oleh LCG. Dengan demikian, proses pengacakan Fisher Yates Shuffle bersifat deterministik selama parameter LCG dan nilai awal (*seed*) yang digunakan sama. Hal ini memungkinkan proses pembentukan matriks Playfair pada tahap enkripsi dan dekripsi menghasilkan susunan alfabet yang identik. Indeks acak ditentukan dengan persamaan berikut [17]:

$$j = X_k \, mod \, (k + 1) \tag{6}$$

di mana
X_k : bilangan acak dari LCG
k : indeks elemen

Playfair Cipher

Playfair Cipher merupakan teknik kriptografi klasik berbasis *digraph substitution*, yaitu mengenkripsi pasangan huruf berdasarkan posisi dalam matriks kunci berukuran 5×5 dalam [1]. Sebelum proses enkripsi dilakukan, plainteks melalui tahap normalisasi, seperti menghapus karakter non-alfabet, mengubah huruf menjadi kapital, mengganti huruf J dengan I, serta memastikan jumlah karakter genap dengan penambahan huruf X jika diperlukan. Proses enkripsi dilakukan berdasarkan beberapa aturan[18], yaitu:

1. Baris yang sama
   Jika kedua huruf berada pada baris yang sama, maka masing-masing huruf diganti dengan huruf di sebelah kanannya.
2. Kolom yang sama
   Jika kedua huruf berada pada kolom yang sama, maka masing-masing huruf diganti dengan di bawahnya.
3. Baris dan kolom berbeda
   Jika kedua huruf berada pada baris dan kolom yang berbeda, maka masing-masing huruf diganti dengan huruf pada baris yang sama dan kolom pasangan hurufnya (membentuk persegi panjang).

Proses dekripsi merupakan kebalikan dari proses enkripsi, dengan aturan sebagai berikut:

1. Baris yang sama
   Jika kedua huruf berada pada baris yang sama, maka masing-masing huruf diganti dengan huruf di sebelah kirinya.
2. Kolom yang sama
   Jika kedua huruf berada pada kolom yang sama, maka masing-masing huruf diganti dengan di atasnya.
3. Baris dan kolom berbeda
   Jika kedua huruf berada pada baris dan kolom yang berbeda, maka masing-masing huruf diganti dengan huruf pada baris yang sama dan kolom pasangan hurufnya (membentuk persegi panjang).

Bahan dan Metode

Penelitian ini diawali dengan identifikasi permasalahan terkait keamanan pertukaran informasi, khususnya dalam bentuk pesan teks yang rentan terhadap akses oleh pihak yang tidak berwenang. Oleh karena itu, diperlukan suatu metode yang mampu meningkatkan kompleksitas kunci serta mengurangi pola penyandian yang mudah diprediksi.

Data yang digunakan dalam penelitian ini berupa data teks (plainteks) yang terdiri dari karakter alfabet A–Z dengan menghilangkan huruf J sesuai dengan aturan Playfair Cipher. Selain itu, digunakan parameter berupa *key*, nilai awal (*seed*), serta parameter pada algoritma LCG, yaitu konstanta pengali a, konstanta penambah b, dan modulus m. Seluruh proses pengolahan data dilakukan menggunakan bahasa

pemrograman Python dengan *bantuan Integrated Development Environment* (IDE) Visual Studio Code untuk mendukung implementasi sistem enkripsi dan dekripsi secara efektif.

Proses pengamanan pesan teks dalam penelitian ini dilakukan dengan mengintegrasikan algoritma LCG, Fisher Yates Shuffle, dan Playfair Cipher. LCG digunakan sebagai pembangkit bilangan acak semu yang menghasilkan deret bilangan berdasarkan parameter yang telah ditentukan. Bilangan acak semu tersebut kemudian digunakan dalam algoritma Fisher Yates Shuffle untuk melakukan pengacakan alfabet secara merata tanpa bias, sehingga diperoleh susunan alfabet yang acak dan sulit diprediksi.

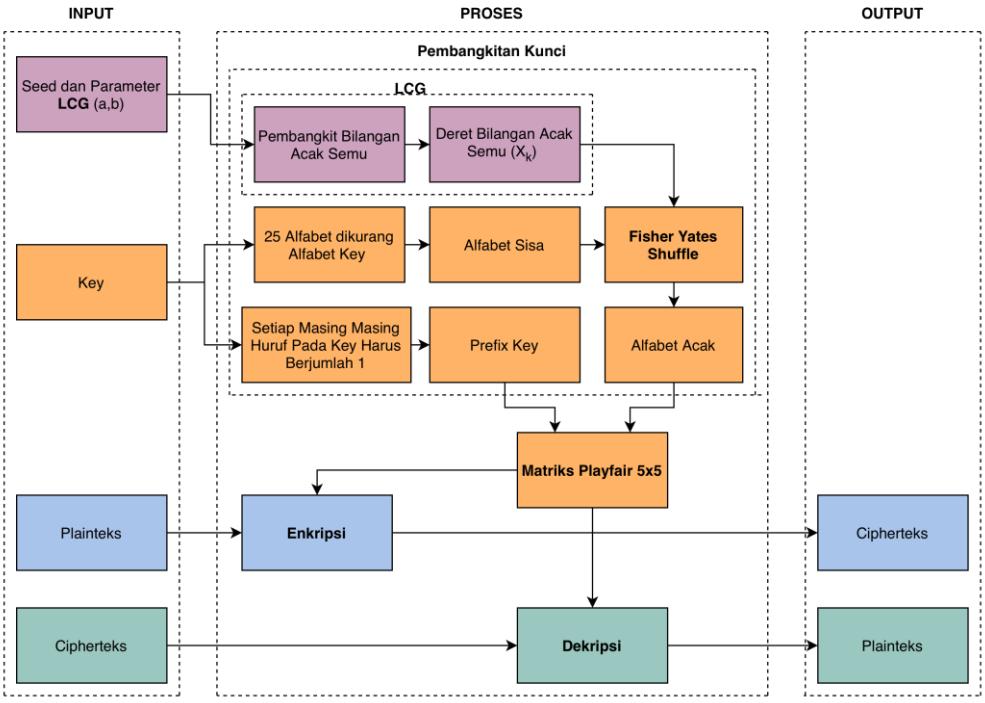
Selanjutnya, alfabet hasil pengacakan dikombinasikan dengan prefix key untuk membentuk matriks Playfair berukuran 5×5. Matriks ini digunakan sebagai dasar dalam proses enkripsi dan dekripsi pesan teks [18]. Sebelum proses enkripsi dilakukan, plainteks terlebih dahulu melalui tahap normalisasi, yaitu penghapusan karakter non-alfabet, pengubahan huruf menjadi kapital, penggantian huruf J menjadi I, serta pembentukan pasangan huruf (*digraph*). Jika terdapat pasangan huruf yang sama, maka disisipkan huruf X, dan jika jumlah karakter ganjil, ditambahkan huruf X di akhir teks.

Proses enkripsi dilakukan dengan menggunakan aturan Playfair Cipher, yaitu aturan baris, kolom, dan persegi panjang untuk menggantikan pasangan huruf menjadi cipherteks. Selanjutnya, proses dekripsi dilakukan dengan menggunakan matriks kunci yang sama, yang dibentuk kembali berdasarkan *key*, *seed*, dan parameter LCG yang identik, sehingga diperoleh kembali plainteks asli.

Implementasi sistem dilakukan dalam bentuk aplikasi berbasis *Graphical User Interface* (GUI) untuk mempermudah pengguna dalam melakukan proses enkripsi dan dekripsi. Hasil dari sistem yang dikembangkan kemudian divalidasi dengan membandingkan hasil enkripsi dan dekripsi program dengan perhitungan manual, sehingga dapat dipastikan bahwa sistem bekerja dengan benar dan sesuai dengan teori yang digunakan.

Hasil dan Pembahasan

Pengembangan sistem pengamanan pesan teks dalam penelitian ini dilakukan dengan mengintegrasikan algoritma LCG, Fisher Yates Shuffle, dan Playfair Cipher. Secara umum, alur proses sistem dimulai dari pembangkitan bilangan acak, pengacakan alfabet, pembentukan matriks kunci, hingga proses enkripsi dan dekripsi pesan teks.

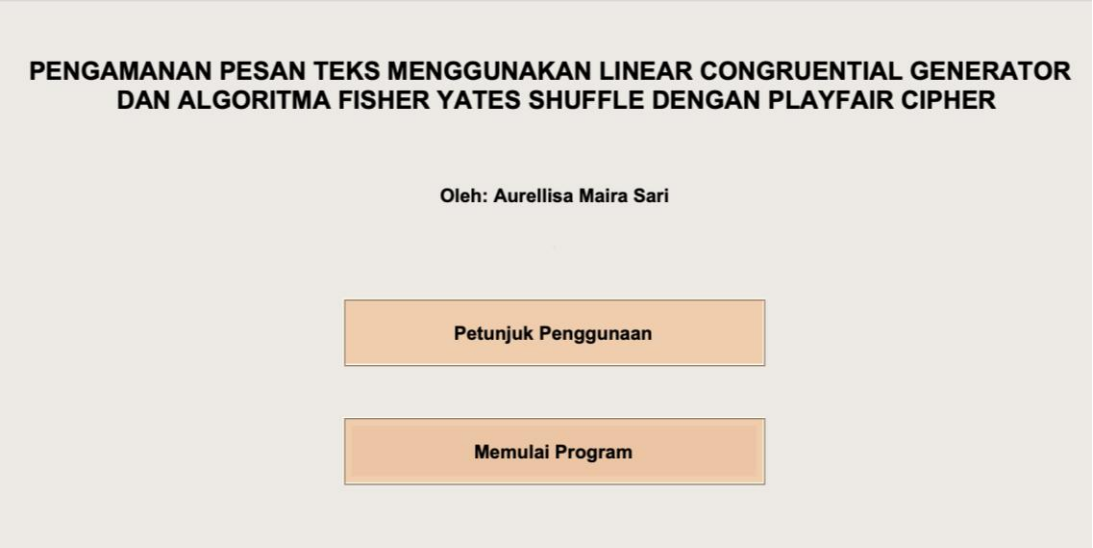


Gambar 1. Skema Aplikasi Pengamanan Pesan Teks Menggunakan LCG-Fisher Yates Shuffle dengan Playfair Cipher.

Proses pada Gambar 1 terdiri dari beberapa tahapan, mulai dari pembangkitan kunci hingga proses dekripsi kembali data terenkripsi. Tahapan-tahapan tersebut dijelaskan sebagai berikut:

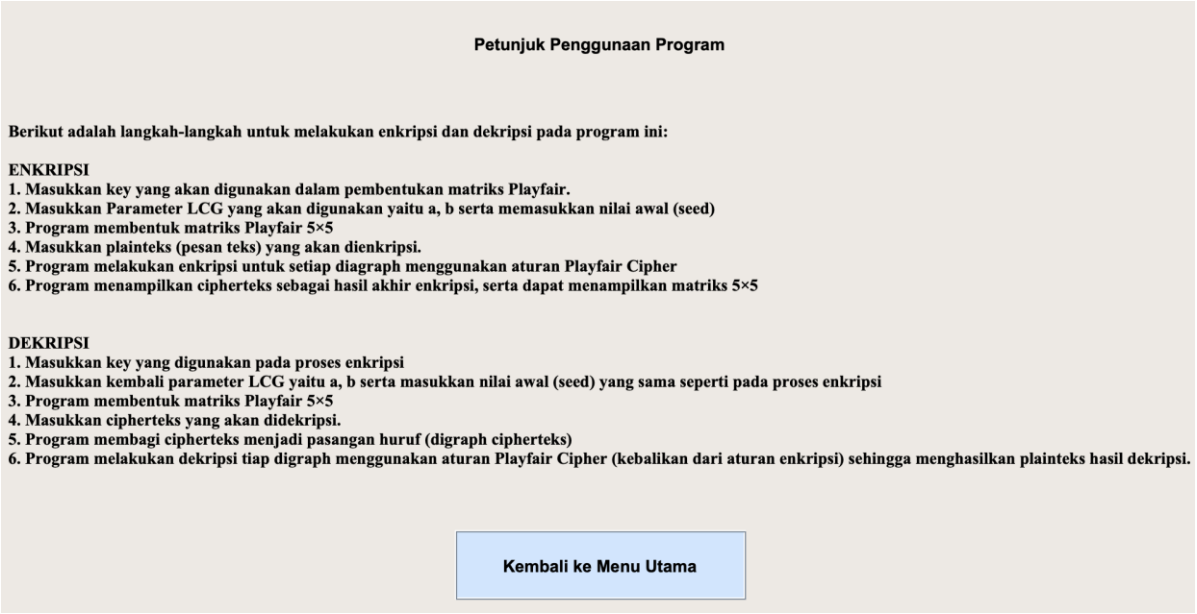
1. Pengguna memasukkan kunci, nilai seed, serta parameter LCG yang meliputi konstanta pengali dan konstanta penambah. Parameter ini digunakan untuk dasar pembangkitan bilangan acak semu.
2. Berdasarkan *seed* dan parameter LCG, proses ini akan menghasilkan deret bilangan acak semu atau X_k.
3. Alfabet yang telah digunakan untuk kunci awal playfair dihilangkan dari himpunan alfabet, sehingga diperoleh alfabet sisa. Alfabet sisa ini akan digunakan sebagai elemen yang akan diacak untuk melengkapi matriks Playfair Cipher.
4. Deret bilangan acak hasil LCG digunakan sebagai indeks dalam algoritma Fisher Yates Shuffle untuk mengacak urutan alfabet sisa secara merata dan tidak bias.
5. Kunci Playfair dan alfabet sisa yang telah diacak kemudian disusun ke dalam matriks Playfair berukuran $5x5$. Matriks ini berfungsi sebagai kunci utama dalam proses enkripsi dan dekripsi.
6. Pengguna memasukkan plainteks yang akan digunakan. Plainteks kemudian diproses menggunakan aturan Playfair Cipher.
7. Plainteks dienkripsi menggunakan matriks Playfair $5x5$ yang telah dibentuk. Proses enkripsi dilakukan berdasarkan posisi pasangan huruf dalam matriks, sehingga dihasilkan cipherteks.
8. Cipherteks akan kembali diproses menggunakan matriks Playfair $5x5$ yang sama dengan proses enkripsi, namun dengan aturan dekripsi.
9. Proses dekripsi menghasilkan kembali plainteks asli.

Program yang telah dikonstruksi memiliki dua pilihan menu utama, yaitu “Petunjuk Penggunaan” dan “Memulai Program”. Tampilan awal dari program ditunjukkan pada Gambar 2. masing-masing *button* akan memunculkan *window* baru jika *button* ditekan.

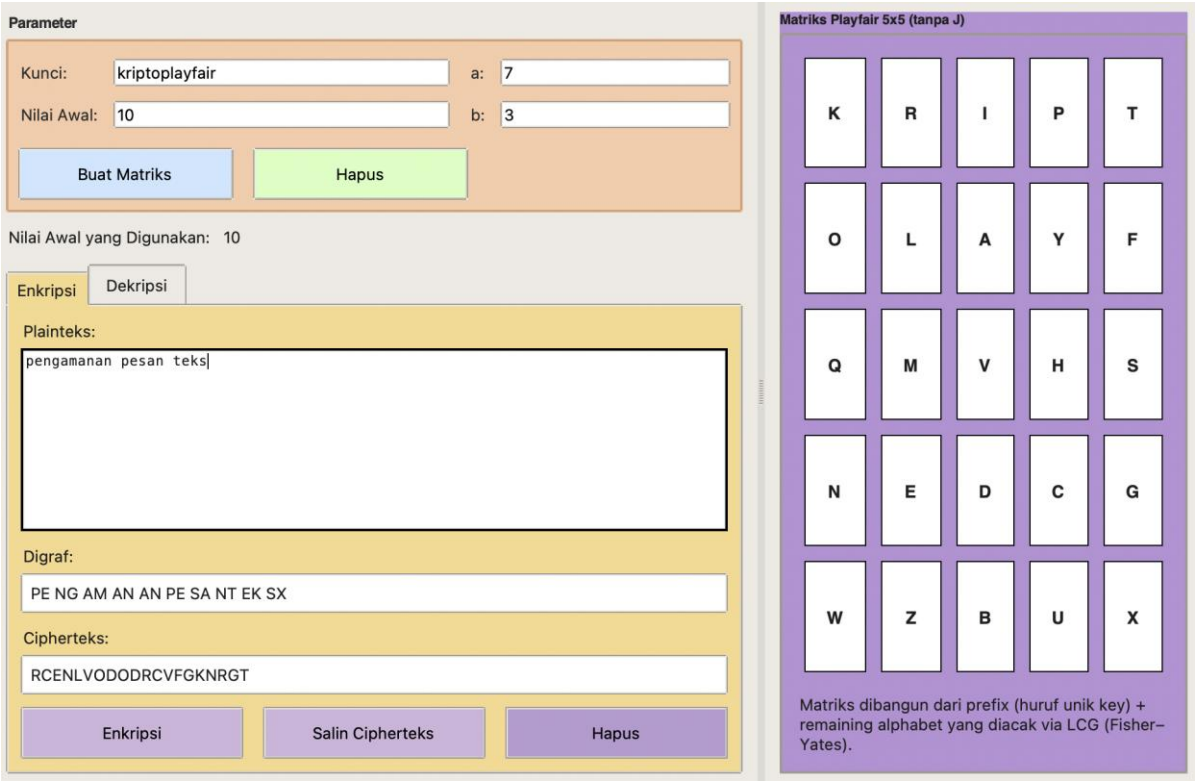


Gambar 2. *Window* Utama

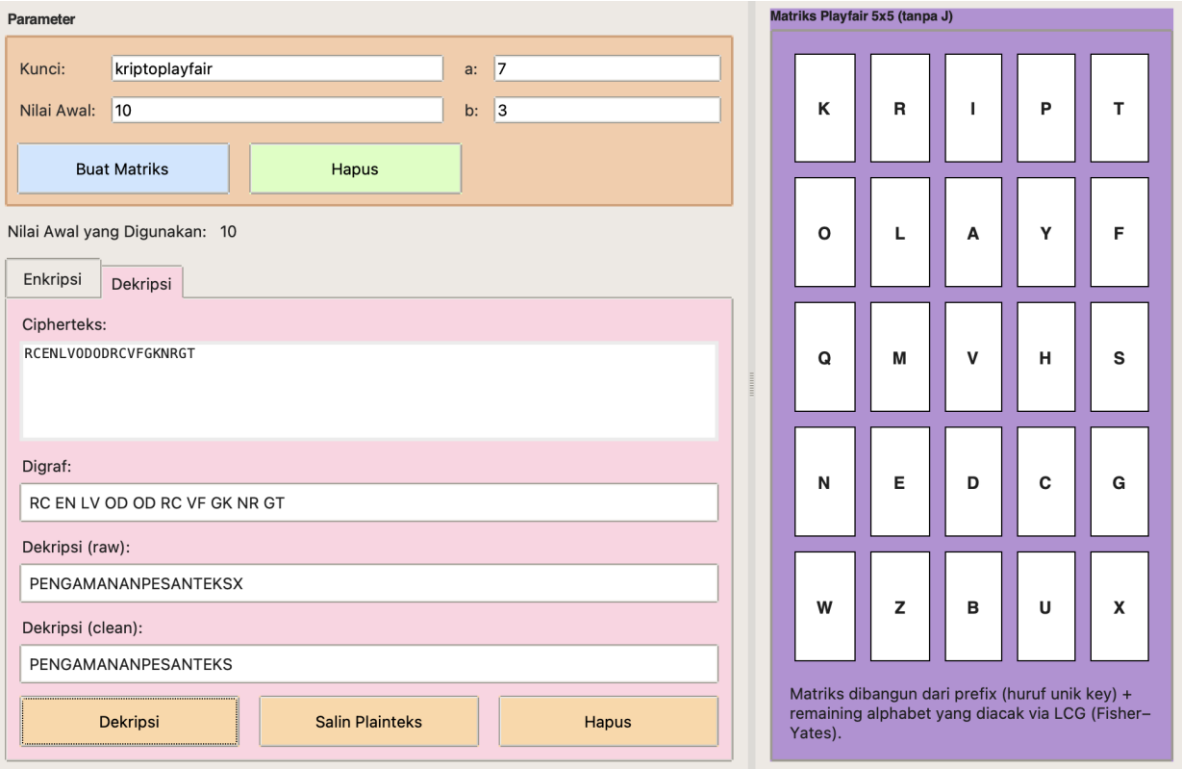
Window petunjuk pengunaan pada Gambar 3 berisi tentang panduan untuk melakukan enkripsi dan dekripsi untuk *window* selanjutnya yaitu *window* program.



Gambar 3. *Window* Petunjuk Penggunaan



Gambar 4. *Window* Memulai Program Enkripsi



Gambar 5. *Window* Memulai Program Dekripsi

Pada Gambar 4 dan Gambar 5 terdapat *window* utama enkripsi dan dekripsi yang digunakan untuk melakukan proses pengamanan dan pemulihan pesan menggunakan Playfair Cipher dengan matriks kunci 5×5 yang dibangkitkan menggunakan LCG dan algoritma Fisher Yates Shuffle. *Window* ini terdiri dari beberapa bagian, yaitu parameter pembangkitan kunci, area enkripsi, area dekripsi, serta tampilan matriks Playfair.

Validasi Program

Setelah program selesai dikonstruksi, validasi terhadap program dilakukan untuk mengetahui apakah penggabungan kriptografi ini berhasil dilakukan.

Validasi Enkripsi dan Dekripsi Berdasarkan Variasi Plainteks dan Kunci

Pada subbab ini dilakukan validasi proses enkripsi dan dekripsi secara manual dengan bantuan Microsoft Excel untuk memastikan bahwa algoritma yang diimplementasikan pada sistem bekerja dengan benar. Proses validasi dilakukan dengan menghitung setiap tahapan algoritma secara bertahap, mulai dari pembangkitan bilangan acak menggunakan LCG, proses pengacakan alfabet menggunakan Fisher Yates Shuffle, hingga proses enkripsi menggunakan Playfair Cipher.

Dalam proses validasi ini digunakan satu contoh data uji yaitu plainteks “ALGORITMA PLAYFAIR DENGAN LCG” dengan kunci “RANDOMSHUFFLEKEY” yang memiliki panjang karakter key sebanyak 14. Parameter yang digunakan pada algoritma LCG didetapkan sebagai berikut:

Tabel 1 Parameter Linear Congruential Generator

Parameter	Nilai
a	7
b	3
X_0	10
m	2^{31}

Tabel 2 Parameter Pembentukan Matriks Playfair

Parameter	Nilai
Panjang *key*	14
Jumlah alfabet playfair	25
Sisa huruf (n)	11

Parameter tersebut digunakan untuk menghasilkan deret bilangan acak yang kemudian dimanfaatkan dalam proses pengacakan alfabet pada algoritma Fisher Yates Shuffle.

1) Pembangkitan Bilangan Acak Menggunakan LCG

Bilangan acak dibangkitkan menggunakan persamaan LCG sebagai berikut:

$$X_{k+1} = (aX_k + b)\ mod\ m \tag{10}$$

Berdasarkan parameter yang telah ditentukan, diperoleh deret bilangan acak seperti pada Tabel 3.

Tabel 3 Hasil Pembangkitan Bilangan Acak Menggunakan LCG

K	X_K
X_0	10
X_1	73
X_2	514
X_3	3601
X_4	25210
X_5	176473
X_6	1235314
X_7	8647201
X_8	60530410
X_9	423712873
X_{10}	818506466

Nilai bilangan acak tersebut selanjutnya digunakan untuk menentukan indeks pertukaran pada algoritma Fisher Yates Shuffle melalui operasi modulo terhadap panjang alfabet.

2) Pengacakan Alfabet dengan Fisher Yates Shuffle

Tahap berikutnya adalah melakukan pengacakan alfabet untuk menghasilkan susunan karakter yang lebih acak. Proses diawali dengan penggunaan 25 huruf alfabet (tanpa huruf J), kemudian dipisahkan antara huruf yang tidak terdapat pada kunci disebut sebagai alfabet sisa, sebagaimana ditunjukkan pada Gambar 6. Huruf yang terdapat pada kunci dan huruf yang belum digunakan.

0	1	2	3	4	5	6	7	8	9	10
b	c	g	i	p	q	t	v	w	x	z

Gambar 6. Alfabet Sisa

Alfabet sisa tersebut kemudian diacak menggunakan algoritma Fisher–Yates Shuffle. Indeks pertukaran J diperoleh dari bilangan acak LCG dengan persamaan:

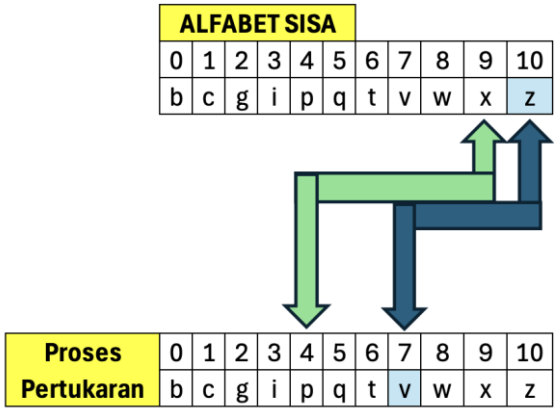
$$j = X_k\ mod\ (k+1) \tag{7}$$

Hasil proses iterasi pengacakan ditunjukkan pada Tabel 4.

Tabel 4 Hasil Iterasi Fisher Yates Shuffle

iterasi	K	X_K	J	Pertukaran
1	10	73	73 mod (11) = 7	A[10]↔A[7]
2	9	514	514 mod (10) = 4	A[9]↔A[4]
3	8	3601	3601 mod (9) = 1	A[8]↔A[1]
4	7	25210	25210 mod (8) = 2	A[7]↔A[2]
5	6	176473	176473 mod (7) = 3	A[6]↔A[3]
6	5	1235314	1235314 mod (6) = 4	A[5]↔A[4]
7	4	8647201	8647201 mod (5) = 1	A[4]↔A[1]
8	3	60530410	60530410 mod (4) = 2	A[3]↔A[2]
9	2	423712873	423712873 mod (3) = 1	A[2] ↔ A[1]
10	1	818506466	818506466 mod (2) = 0	A[1]↔A[0]

Proses ini menghasilkan perubahan posisi elemen secara bertahap hingga diperoleh urutan alfabet yang telah teracak secara merata, seperti ditunjukkan pada Gambar 7.



Gambar 7. Proses Pertukaran Elemen

Setelah seluruh proses pertukaran dilakukan sesuai jumlah iterasi Fisher Yates Shuffle, diperoleh urutan alfabet hasil pengacakan pada Gambar 8 sebagai berikut:

t	b	q	z	w	x	i	g	c	p	v

Gambar 8. Alfabet Hasil Pengacakan

Urutan alfabet hasil pengacakan ini kemudian digunakan untuk melengkapi karakter yang belum terdapat pada kunci dalam pembentukan matriks Playfair.

3) Pembentukan Matriks Playfair

Setelah proses pengacakan selesai, alfabet hasil pengacakan digunakan untuk melengkapi karakter pada kunci dalam pembentukan matriks Playfair berukuran 5×5. Matriks ini selanjutnya digunakan sebagai dasar dalam proses enkripsi dan dekripsi. Hasil pembentukan matriks ditunjukkan pada Gambar 9.

MATRIKS				
R	A	N	D	O
M	S	H	U	F
L	E	K	Y	T
B	Q	Z	W	X
I	G	C	P	V

Gambar 9. Hasil Pembentukan Matriks Playfair 5×5

4) Validasi Proses Enkripsi

Plainteks diubah menjadi pasangan huruf (*digraph*) dan dienkripsi menggunakan aturan Playfair Cipher, yaitu:

1. Jika kedua huruf berada pada baris yang sama, maka masing-masing huruf diganti dengan huruf di sebelah kanannya.
2. Jika kedua huruf berada pada kolom yang sama, maka masing-masing huruf diganti dengan huruf di bawahnya.
3. Jika kedua huruf berada pada baris dan kolom yang berbeda, maka masing-masing huruf diganti dengan huruf pada sudut persegi panjang yang berada pada baris yang sama.

Berdasarkan proses tersebut, diperoleh hasil enkripsi seperti ditunjukkan pada Tabel 5.

Tabel 5 Hasil Perhitungan Manual Enkripsi

Pasangan Huruf Plainteks	**Enkripsi**
AL	RE
GO	VA
RI	MR
TM	LF
AP	DG
LA	ER
YF	TU
AI	RG
RD	AO
EN	KA
GA	AS
NL	RK
CG	PC

Dari hasil perhitungan, diperoleh cipherteks:

REVAMRLFDGERTURGAOKAASRKPC

5) Validasi Proses Dekripsi

Proses dekripsi dilakukan menggunakan matriks Playfair yang sama dengan proses enkripsi. Cipherteks dibagi menjadi pasangan huruf dan didekripsi berdasarkan aturan Playfair Cipher, yaitu:

1. Jika kedua huruf berada pada baris yang sama, maka masing-masing huruf diganti dengan huruf di sebelah kirinya.
2. Jika kedua huruf berada pada kolom yang sama, maka masing-masing huruf diganti dengan huruf di atasnya.
3. Jika kedua huruf berada pada kolom dan baris yang berbeda, maka masing-masing huruf diganti dengan huruf pada sudut persegi panjang yang berada pada baris yang sama.

Berdasarkan proses tersebut, diperoleh hasil enkripsi seperti ditunjukkan pada Tabel 6.

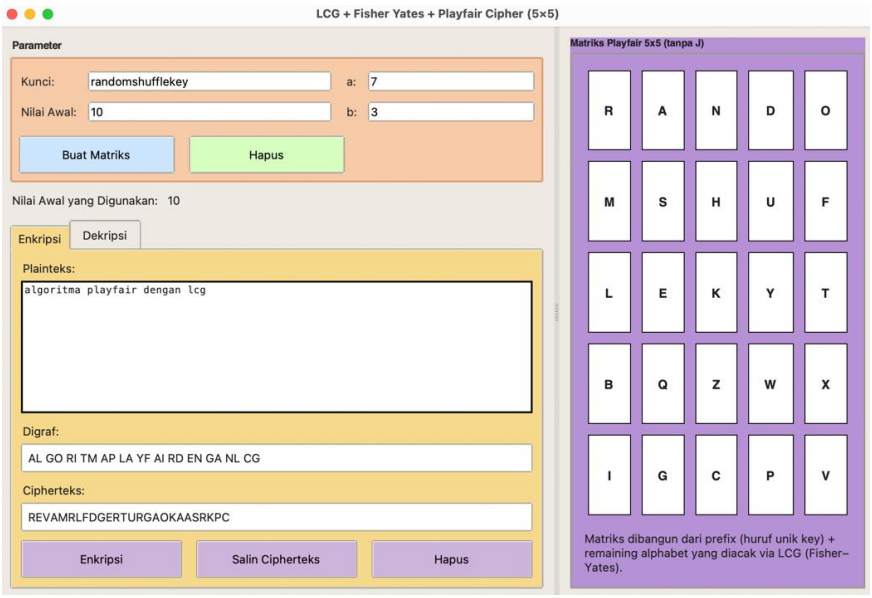
Tabel 6 Hasil Perhitungan Manual Dekripsi

Pasangan Huruf Cipherteks	Enkripsi
RE	AL
VA	GO
MR	RI
LF	TM
DG	AP
ER	LA
TU	YF
RG	AI
AO	RD
KA	EN
AS	GA
RK	NL
PC	CG

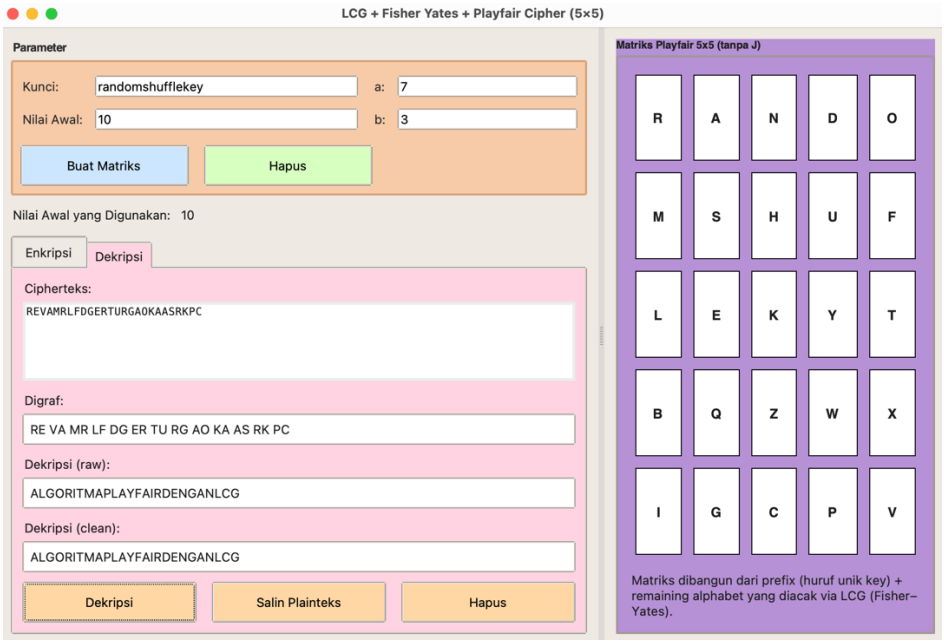
Dari hasil dekripsi, diperoleh plainteks:

ALGORITMAPLAYFAIRDENGANLCG

6) Visualisasi Hasil Program Enkripsi dan Dekripsi



Gambar 10. Hasil Perhitungan Program Enkripsi



Gambar 11. Hasil Perhitungan Program Dekripsi

Hasil validasi menunjukkan bahwa proses enkripsi dan dekripsi yang dilakukan oleh sistem sesuai dengan perhitungan manual. Selain itu, integrasi algoritma LCG dan Fisher Yates Shuffle dalam pembentukan matriks Playfair menghasilkan variasi susunan alfabet yang lebih acak.

Validasi Enkripsi dan Dekripsi Berdasarkan Pengaruh Parameter

Pada eksperimen ini dilakukan pengujian pengaruh parameter LCG terhadap proses pembentukan matriks Playfair. Plainteks dan *key* dijaga tetap, sedangkan parameter LCG berupa konstanta pengali a, konstanta penambah b, dan seed divariasikan. Variasi parameter yang digunakan pada pengujian ini terdapat pada Tabel 7.

Tabel 7 Variasi Parameter LCG

No	a	*b*	*Seed*
1.	50	4	20
2.	30	11	25

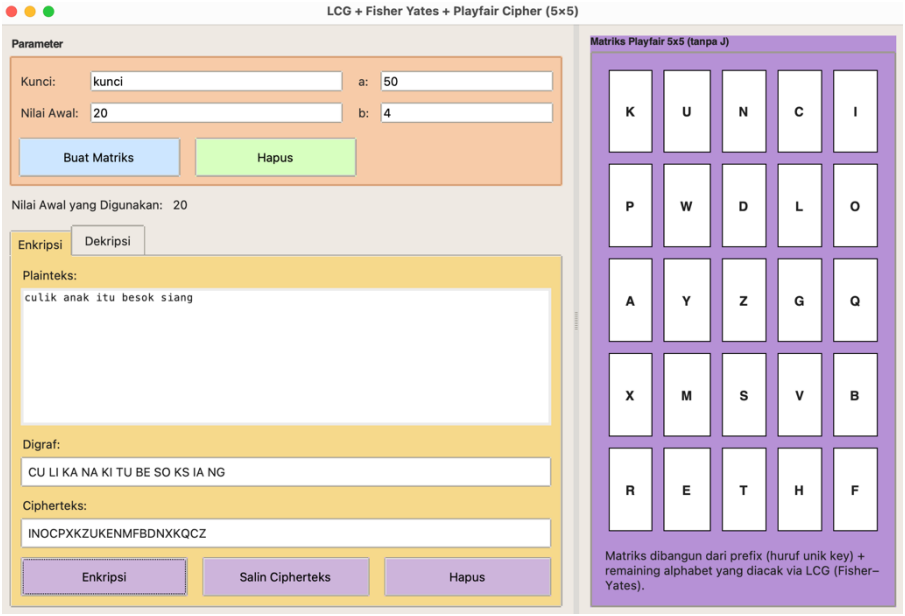
Pengujian dilakukan menggunakan plainteks “CULIK ANAK ITU BESOK SIANG”. Berdasarkan variasi parameter yang digunakan, diperoleh hasil enkripsi dan dekripsi seperti ditunjukkan pada Tabel 8.

Tabel 8 Hasil Enkripsi dan Dekripsi Berdasarkan Variasi Parameter LCG

No	Matriks Berubah	Cipherteks	Plainteks
1.	Ya	INOCPXKZUKENMFBDNXKQCZ	CULIKANAKITUBESOKSIANG
2.	Ya	INBFROKPUKZCOBYMUYKLGV	CULIKANAKITUBESOKSIANG

Berdasarkan hasil perhitungan manual yang ditunjukkan pada Tabel 8, perubahan parameter LCG menghasilkan susunan matriks Playfair yang berbeda pada setiap percobaan. Perbedaan susunan matriks tersebut menyebabkan cipherteks yang dihasilkan juga berbeda, meskipun plainteks dan key yang

digunakan tetap sama. Visualisasi hasil program enkripsi dan dekripsi untuk variasi parameter yang digunakan terdapat pada Gambar 11 dan Gambar 12.



Gambar 12. Hasil Perhitungan Program Enkripsi Berdasarkan Variasi Parameter ke-1

Berdasarkan hasil perhitungan manual dan hasil perhitungan menggunakan program, seluruh cipherteks berhasil didekripsi kembali menjadi plainteks semula. Hal ini menunjukkan bahwa sistem bersifat konsisten dan *reversible*, meskipun parameter LCG yang digunakan divariasikan.

Kesimpulan

Penelitian ini berhasil mengembangkan metode pengamanan pesan teks dengan mengintegrasikan *Linear Congruential Generator* (LCG) dan algoritma Fisher Yates Shuffle dalam pembentukan matriks Playfair Cipher. Pendekatan ini mampu menghasilkan variasi matriks kunci yang dinamis melalui proses pengacakan berbasis bilangan acak semu, sehingga meningkatkan kompleksitas sistem dibandingkan Playfair Cipher klasik yang menggunakan matriks statis.

Hasil pengujian menunjukkan bahwa perubahan parameter LCG secara langsung memengaruhi susunan matriks yang terbentuk, yang berdampak pada perbedaan cipherteks yang dihasilkan meskipun plainteks dan *key* tetap. Hal ini menunjukkan bahwa metode yang diusulkan memiliki kemampuan dalam menghasilkan variasi kunci yang tinggi, sehingga lebih sulit diprediksi dan berpotensi meningkatkan ketahanan terhadap analisis kriptanalisis berbasis pola.

Selain itu, implementasi sistem dalam bentuk aplikasi berbasis *Graphical User Interface* (GUI) menunjukkan bahwa metode yang diusulkan dapat diterapkan secara praktis. Proses enkripsi dan dekripsi yang dihasilkan terbukti konsisten dengan perhitungan manual, sehingga menjamin keakuratan dan keandalan sistem dalam mengamankan serta mengembalikan pesan teks ke bentuk semula.

Dengan demikian, integrasi LCG dan Fisher Yates Shuffle pada Playfair Cipher tidak hanya meningkatkan tingkat keacakan kunci, tetapi juga memberikan pendekatan yang efektif dan aplikatif dalam pengembangan sistem kriptografi klasik yang lebih adaptif terhadap kebutuhan keamanan informasi.

Referensi

[1] A.A. Siagian and Z. Indra, "Analisis Teknik Playfair dan Shift Cipher sebagai Metode Kriptografi Klasik untuk Keamanan Data", *Jurnal Komputer dan Teknologi (JUKOMTEK)*, vol. 4, no. 1, pp. 13-14, 2025

[2] Chandra, "Keamanan Data dengan Metode Kriptografi Kunci Publik", *Jurnal TIMES*, vol. 5, no. 2, pp. 11-15, 2016.

[3] A. Mittal and F. Sidney, "Secure Data Communication Using Padding Key Encryption Cryptography Algorithm," IEEE International Conference on Integrated Circuits and Communication Systems (ICICACS), Raichur, India, pp. 1-5, 2023

[4] K. Sasikumar and S. Nagarajan, "Comprehensive Review and Analysis of Cryptography Techniques in Cloud Computing", *IEEE Access*, vol. 12, pp. 52325-52351, 2024.

[5] R. Marwati & K. Yulianti, "Cryptanalysis on classical cipher based on Indonesian language*", Journal of Physics: Conference Series*, vol. 1013, 012147, 2018.

[6] D. Kurniawan and B. Priyatna, "Pengamanan Data Berbasis Mobile Android dengan Penggabungan Linear Feedback Shift Register (LFSR) dan Modifikasi Matriks Kunci Algoritma Kriptografi Playfair Cipher*", Jurnal Telematika MKOM*, vol. 10, no. 1, 2018.

[7] T.M.P. Sancaka dan V. Lusiana, "Penerapan Metode Playfair Cipher dalam Aplikasi Enkripsi–Dekripsi File Teks*", Jurnal Ilmiah Elektronika dan Komputer,* vol. 15, no. 2, pp.260–*270, 2022.*

[8] D. Susanti, 'Analisis Modifikasi Metode Playfair Cipher Dalam Pengamana Data Teks*", Indonesian Journal of Data and Science*, vol. 1, no.1, pp. 11-8, 2020.

[9] G. R. Fahrezi, D. V. S. Y. Sakti, and H. Muhardi, "Implementasi *Pseudo Random Number Generator* (PRNG) dengan Algoritma *Linear Congruential Generator* (LCG) pada Permainan Mengetik Bertema Candi Prambanan," *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, vol. 9, no. 2, pp. 289-298, 2023.

[10] F. Naufal, M., R. Marwati, & R. Sispiyati, "Kriptografi audio menggunakan transposisi dan *Affine Cipher* yang dikembangkan dengan algoritma *Blum Blum Shub*", Jurnal EurekaMatika, vol. 9, no. 1, pp. 1-10, 2021.

[11] M. Wildan, W. M. Ashari, "Text Data Security Using LCG and CBC with Steganography Technique on Digital Image", *Journal of Applied Informatics and* Computing, vol. 8, no. 2, pp. 400-407, 2024.

[12] Ekojono, R. Cahyaningrum, K. S. Batubulan, "Implementasi Metode Fisher-Yates Shuffle Dan Fuzzy Tsukamoto Pada Game 2D Gopoh Berbasis Android", Jurnal Informatika Polinema, vol 4, no.3, pp. 174-180, 2018.

[13] A. Najib, B. S. W. Poetro, and S. Mulyono, "Penerapan Algoritma Fisher Yates Shuffle Pada Game Edukasi Belajar Matematika," *Seminar Riset Mahasiswa Computer & Electrical (SERIMA-CE)*, vol. 1, no. 1, pp. 90–95, 2023

[14] Rosen, K. H. (2019). *Discrete Mathematics and Its Applications* (8th ed.). McGraw-Hill Education.

[15] Stallings, W. (2017). *Cryptography and network security: Principles and practice* (7th ed.). Pearson.

[16] N. Bhavsar, T. Shinde, S. Narwade, dan A. Shinde, "Performance Analysis on Different Permutation Algorithms for Security Applications", *International Journal of Advance Research in Computer and Communication Engineering,* vol. 6, no. 4, pp. 495-499,2017.

[17] Axler, S. (2015). Linear Algebra Done Right (3rd ed.). New York: Springer.

[18] A. Japardi, E. L. Frasetyo, A. Sahertian, C. Umam, dan G. A. Trisnapradika, "Playfair Cipher untuk Kriptografi ASCII menggunakan Matriks 9×10", *Jukompak (Jurnal Komputasi dan Pengembangan Aplikasi)*, vol. 1, no. 1, pp. 40-49, 2025.