Journal of Information System Research (JOSH)
Volume 6, No. 1, Oktober 2024, pp 679−687
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v6i1.6112

Identifikasi Nilai Keacakan Berdasarkan Reposisi Fungsi XOR Pada Blok Kedua LFSR A5/1

Jorghie Theodore Kenzo Pallangan*, Alz Danny Wowor

Fakultas Teknologi Informasi, Teknik Informatika, Universitas Kristen Satya Wacana, Salatiga
Jl. Diponegoro No.52-60, Salatiga, Kec. Sidorejo, Kota Salatiga, Jawa Tengah, Indonesia
Email: [1,*]672019265@student.uksw.edu, [2]alzdanny.wowor@uksw.edu
Email Penulis Korespondensi: 672019265@student.uksw.edu
Submitted: **21/10/2024;** Accepted: **31/10/2024;** Published: **31/10/2024**

Abstrak−Penelitian ini merencanakan metode pembangkit bilangan acak dengan metode Linear Feedback Shift Register (LFSR) menggunakan skema A5/1 yang melibatkan tiga fungsi umpan balik. XOR digunakan sebagai penentu Nilai bit keluaran yang baru pada iterasi selanjutnya dalam mekanisme umpan balik. Bahan uji dalam menghasilkan keluaran acakan terhadap suatu inputan menggunakan RunTest, MonoBit, dan Blockbit. Pengujian menggunakan tiga fungsi umpan balik dilakukan untuk dibandingkan dengan penelitian sebelumnya yang menghasilkan bilangan acak. Pengujian enkripsi plainteks serta Cipherteks menunjukan tingkat korelasi "sangat kecil" dengan nilai rata-rata yang mendekati nilai 0. Penggunaan LFSR dengan skema A5/1 yang melibatkan tiga fungsi XOR, menciptakan luaran yang Acak serta dapat digunakan terhadap Stream Chiper.

Kata Kunci: Kriptografi; Linear Feedback Shift Register; Skema A5/1; XOR

Abstract−This research plans a random number generation method using the Linear Feedback Shift Register (LFSR) method with the A5/1 scheme which involves three feedback functions. XOR is used to determine the new output bit value in the next iteration in the feedback mechanism. The test material produces random output for an input using Run Test, Mono Bit, and Block bit. Tests using three feedback functions were carried out to compare with previous research which generated random numbers. Testing of plaintext and ciphertext encryption shows a very small level of correlation with an average value close to 0. The use of LFSR with the A5/1 scheme which involves three XOR functions, creates random output and can be used against Stream Chipers.

Keywords: Cryptography; Linear Feedback Shift Register; A5/1 Scheme; XOR

1. PENDAHULUAN

Kriptografi merupakan salah satu bidang penting dalam ilmu komputer dan teknologi informasi yang berperan besar dalam menjaga keamanan data [1][2]. Keamanan data sangat mempengaruhi kepercayaan pengguna dalam memanfaatkan algoritma atau aplikasi yang digunakan. Algoritma yang baik harus memperhitungkan efisiensi waktu dan ruang agar proses enkripsi-dekripsi dapat berjalan secara optimal. Optimalisasi algoritma kriptografi dapat dimulai dengan memastikan proses pembangkitan kunci yang mampu menerima input secara acak dan menghasilkan output yang acak, sehingga cipherteks dapat dengan efektif menyembunyikan informasi penting dari plainteks dalam proses enkripsi.teks [3][4][5].

Linear Feedback Shift Register (LFSR) bekerja dengan cara menggeser bit-bit dalam register dan memanfaatkan operasi XOR untuk menentukan bit keluaran berikutnya. Proses ini dilakukan secara berulang sehingga menghasilkan bit-bit yang acak [6][7]. Dengan melakukan pergeseran bit masukan, exclusive-or (XOR) digunakan sebagai penentu bit acak periode maksimal [8][9]. Bit masukan dapat digunakan untuk menghasilkan bit pada seluruh register geser atau sebagai fungsi umpan balik. [10][11] Untuk mencapai periode maksimal, konfigurasi tap atau titik-titik dalam register yang digunakan dalam operasi XOR harus dipilih secara hati-hati berdasarkan polinomial primitif tertentu. Bit masukan ini kemudian menjadi pembangkit bagi seluruh bit dalam register atau dapat juga berfungsi sebagai bagian dari mekanisme umpan balik, di mana nilai bit keluaran diumpankan kembali ke dalam register untuk terus menghasilkan deret bit acak.

Skema A5/1 adalah algoritma stream cipher yang banyak diterapkan dalam sistem komunikasi GSM untuk melindungi transmisi data [12]. Dalam konteks skema A5/1, terdapat tiga blok utama yang digunakan, dimana setiap blok terdapat proses fungsi XOR sebagai umpan balik untuk melakukan proses iterasi berikutnya. Setiap blok memiliki peran dalam menghasilkan luaran bit yang acak, begitu juga dengan pemilihan setiap entri bit pada masing-masing blok. Kajian dalam penelitian yang dilakukan sekarang ini adalah menggunakan entri bit dalam blok kedua, atau melakukan reposisi bit yang lain dalam menghasilkan A2 yang lain. Pengujian keacakan menjadi acuan untuk membedakan setiap reposisi yang menghasilkan nilai keacakan terbaik. Selain itu pengujian enkripsi juga digunakan untuk menguji setiap luaran bit yang dihasilkan dari proses reposisi pada A5/1, sehingga proses perbandingan dilakukan untuk melihat apakah ada reposisi yang lebih baik dibandingkan dengan A2. Secara keseluruhan terdapat 22-bit dan yang diambil 2-bit, karena fungsi XOR tunduk pada hukum komutatif sehingga jumlahan 2-bit di posisi yang berbeda akan menghasilkan luaran yang sama.

Dalam penelitian ini, digunakan fungsi polinomial dari studi sebelumnya sebagai dasar untuk menguji proses pembangkitan bilangan acak. Penelitian [13] Dengan menggabungkan fungsi polinomial dan metode iterasi seperti Iterasi Titik Tetap serta Newton-Raphson, penelitian ini bertujuan untuk menghasilkan bilangan acak yang memenuhi uji statistik, termasuk Run test, Mono bit, dan Block bit [14]. Bilangan acak yang diperoleh

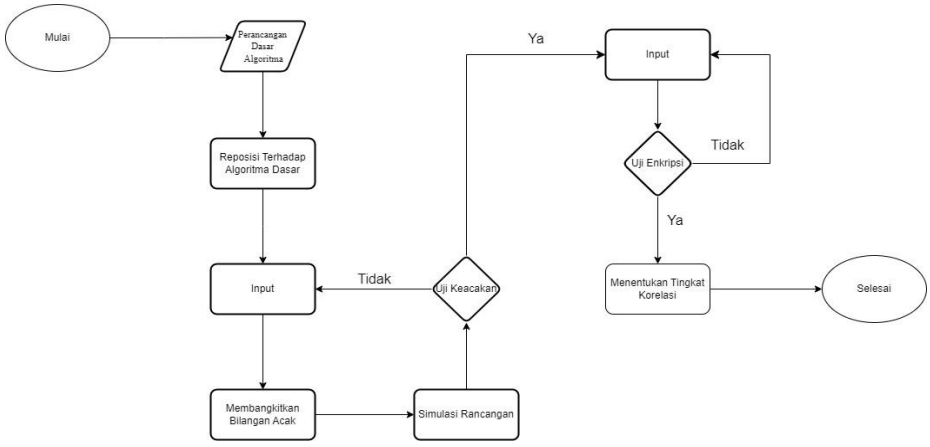
Journal of Information System Research (JOSH)
Volume 6, No. 1, Oktober 2024, pp 679−687
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v6i1.6112

juga diuji dalam proses enkripsi dan terbukti independen secara statistik. Penelitian [15] Penelitian ini memanfaatkan fungsi logaritma sebagai elemen utama dalam proses pembangkitan bilangan acak dalam framework Chaos CSPRNG. Fokus utama dari penelitian ini adalah pada pembuatan kunci kriptografi, mengingat peran penting kunci dalam melindungi informasi sensitif. Dengan menerapkan fungsi matematika, khususnya logaritma, sebagai metode untuk menghasilkan kunci, penelitian ini mengeksplorasi kemungkinan peningkatan ketidakpastian serta kekuatan kunci kriptografi. Penelitian [16] memanfaatkan fungsi $3x^3 + 5x^2+7x-20$ sebagai pembangkit bilangan acak, metode Newton-Raphson digunakan dalam prosesnya. Hasil simulasi menunjukkan bahwa nilai inisialisasi $x_0 = 0.845363$ adalah nilai yang paling optimal, menghasilkan empat set data yang dapat digunakan sebagai kunci. Uji visualisasi mengungkapkan bahwa setiap set data membentuk pola chaos yang dianggap acak berdasarkan uji statistik seperti run test dan mono bit test. Bilangan acak yang dihasilkan berhasil digunakan sebagai kunci dalam proses enkripsi, sehingga tidak ditemukan adanya hubungan statistik antara plainteks dan cipherteks. Fungsi ini terbukti efektif sebagai generator bilangan acak dan dapat menjadi alternatif yang kuat untuk menciptakan bilangan acak berbasis CSPRNG chaos. Penelitian [17] Dengan memanfaatkan fungsi kuadrat x^2- 2x - 4 sebagai generator bilangan acak, penelitian ini menerapkan metode Iterasi Titik Tetap.. Empat fungsi iterasi dihasilkan dan kemudian diuji melalui visualisasi serta uji keacakan menggunakan run test mono bit. Selanjutnya, dilakukan analisis korelasi antara plainteks dan cipherteks untuk menentukan apakah kunci yang digunakan bersifat acak. Hasil penelitian menunjukkan bahwa fungsi x^2-2x-4 dapat menghasilkan bilangan acak yang berbasis pada CSPRNG chaos. Penelitian [18] menggunakan pendekatan komposisi fungsi, di mana fungsi polinomial, f(x), dikombinasikan dengan fungsi trigonometri, g(x). Kombinasi ini menghasilkan enam komponen dari fungsi trigonometri untuk setiap fungsi dalam bentuk polinomial yang dianalisis. Tujuan utama dari pendekatan ini adalah untuk menemukan solusi bagi fungsi polinomial yang sebelumnya tidak mampu membangkitkan bilangan acak secara optimal. Dengan menambahkan fungsi trigonometri ke dalam komposisi, penelitian ini berhasil memperbaiki beberapa fungsi polinomial yang awalnya tidak efektif dalam menghasilkan bilangan acak. Hasilnya menunjukkan bahwa kombinasi fungsi polinomial dan trigonometri dalam bentuk komposisi dapat digunakan sebagai dasar untuk pembangkit bilangan acak yang kuat, seperti CSPRNG Cryptographically Secure Pseudo-Random Number Generator berbasis CHAOS.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Alur rancangan penelitian tidak hanya berfungsi sebagai pedoman untuk mengelola waktu dan sumber daya, tetapi juga sebagai alat untuk memastikan bahwa metodologi yang diterapkan sesuai dengan tujuan yang ingin dicapai. Sebuah alur penelitian yang terstruktur memungkinkan peneliti untuk memastikan bahwa setiap tahap penelitian, mulai dari perumusan masalah, pengumpulan data, hingga analisis hasil, dilakukan secara konsisten dan terintegrasi. Dengan adanya alur yang rinci, peneliti dapat mengevaluasi apakah metode yang digunakan masih relevan dan efektif di setiap tahap, serta memfasilitasi penyesuaian yang diperlukan jika kondisi penelitian berubah. Hal ini penting untuk menjaga validitas dan reliabilitas hasil penelitian, karena sebuah kesalahan kecil dalam satu tahap bisa berdampak besar pada keseluruhan penelitian. Selain itu, alur rancangan yang jelas dapat memudahkan peneliti dalam mendokumentasikan proses penelitian secara rinci. Dokumentasi yang baik sangat penting, terutama untuk penelitian yang memerlukan reproduksibilitas atau ingin dikembangkan lebih lanjut di masa depan. Dengan mengikuti alur yang sistematis, setiap langkah penelitian dapat dicatat dengan baik, memudahkan peneliti lain untuk memahami metodologi yang digunakan dan memungkinkan pengulangan penelitian dengan kondisi yang serupa. Hal ini juga penting dalam menyusun laporan penelitian, karena alur yang rapi dan terdokumentasi dengan baik akan memudahkan penyusunan hasil penelitian yang komprehensif dan transparan.



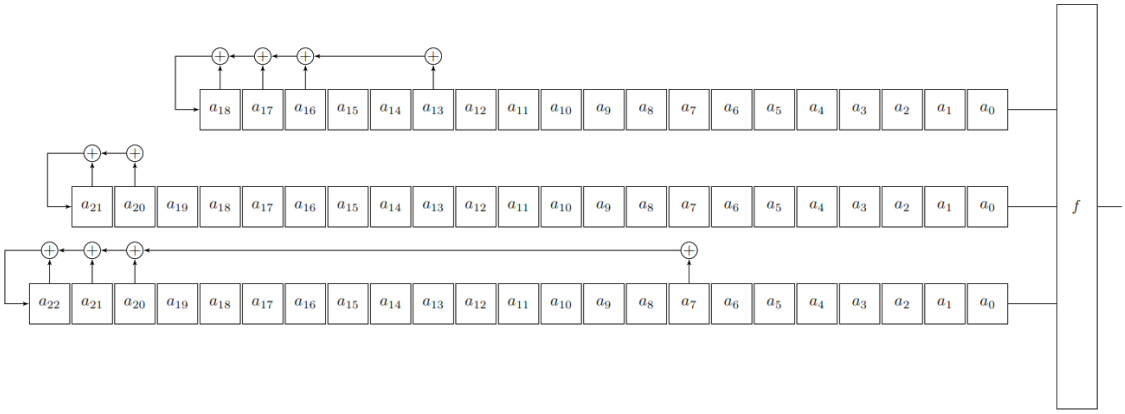
Gambar 1. Desain Pengujian

Journal of Information System Research (JOSH)
Volume 6, No. 1, Oktober 2024, pp 679−687
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v6i1.6112

Pada Gambar (1) dimulai dengan Tahap Per ancangan fungsi umpan balik Ai = 1, 2, 3, dilajutkan dengan melakukan reposisi operasi XOR pada blok A1, kemudian mendesain algoritma setiap reposisi yang telah di rancang. Tahap pembangkitan bit acak dimulai dengan memasukkan plainteks, tidak lupa masukan diubah ke dalam kode ASCII yang kemudian dijadikan ke bilangan biner, setelah didapatkan bilangan biner dilakukan proses padding. Melakukan si- mulasi model, dengan melakukan operasi Runs Test, bl,mb, yang kemudian disimpan. jika telah dilakukan ke semua rancangan reposisi dan data masih kurang, mengulang dengan memasukan plainteks, dan jika data sudah mencukupi akan dilanjutkan dengan mencari rata setiap operasi pembangkit keacakan. Rata-rata tersebut kemudian akan dipilih 10 teratas sebagai bahan pengujian deksripsi. Uji deksripsi dimulai dengan memasukan ciphertext, masukan tersebut akan dirubah ke dalam bilangan ASCII yang digunakan sebagai acuan dalam melakukan tingkat in, dan diklasifikasin berdasarkan tingkatan korelasinya.

2.2 Skema A5/1

Skema A5/1 berfungsi sebagai metode perlindungan yang dirancang untuk mengamankan data dari potensi penyadapan dan pencurian informasi yang dapat terjadi dalam jaringan 2G atau GSM[19]. Dengan menggunakan algoritma ini, komunikasi yang berlangsung antara perangkat dapat terlindungi dari pihak yang tidak berwenang, sehingga meningkatkan keamanan informasi yang dikirimkan. Sistem ini memainkan peran penting dalam menjaga kerahasiaan dan integritas data dalam lingkungan jaringan seluler, terutama di era di mana ancaman terhadap keamanan informasi semakin meningkat. Implementasi skema A5/1 membantu memastikan bahwa data yang ditransmisikan tetap aman dari berbagai risiko yang dapat merugikan pengguna dan penyedia layanan[12]. A5/1 merupakan kumpulan dari beberapa LFSR. Dengan A sebagai fungsi utama dari setiap keluaran fungsi linier Ai. meskipun skema A5/1 dirancang untuk keamanan, kelemahan dalam algoritma ini telah teridentifikasi seiring dengan perkembangan teknologi dan metode serangan yang lebih canggih. Beberapa penelitian telah menunjukkan bahwa skema A5/1 rentan terhadap serangan tertentu seperti time-memory tradeoff attack, di mana penyerang dapat menyimpan sejumlah besar data pra-dihitung untuk memecahkan enkripsi dalam waktu yang relatif singkat. Meskipun demikian, dalam konteks penggunaan sehari-hari di jaringan 2G, skema A5/1 masih memberikan lapisan keamanan yang signifikan dibandingkan dengan tidak adanya enkripsi sama sekali. Inilah sebabnya, meskipun jaringan yang lebih baru seperti 3G dan 4G telah mengadopsi algoritma enkripsi yang lebih kuat, A5/1 tetap digunakan di beberapa wilayah yang masih mengandalkan teknologi GSM.



Gambar 2. Skema A5/1 3 Fungsi XOR

Gambar (2) terdiri dari tiga baris LFSR, pada baris pertama (R1) memiliki panjang 19 bit dimulai dari bit 0 hingga 18, dengan fungsi polinomial umpan balik terdapat pada bit 13, 16, 17 dan 18. Baris kedua (R2) memiliki panjang 22 bit dimulai dari bit 0 hingga 21, dengan fungsi polinomial umpan balik terdapat pada bit 20 dan 21. Baris ketiga (R3) memiliki panjang 23 bit dimulai dari bit 0 hingga 22, dengan fungsi polinomial umpan balik terdapat pada bit 7, 20, 21, 22.

2.3 Run Test

Run test adalah pengujian statistik yang mengevaluasi panjang dan jumlah "runs" dalam sebuah urutan bit. Run di sini merujuk pada serangkaian bit yang berurutan yang semuanya memiliki nilai yang sama

1. Fungsi panggil
   Runs(n), n panjang bit string dan ε merupakan urutan bit yang dari hasil uji, berikut struktur fungsi umum

   $$\varepsilon = \varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n. \quad (1)$$

2. Uji statistik

Journal of Information System Research (JOSH)
Volume 6, No. 1, Oktober 2024, pp 679−687
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v6i1.6112

V_n(obs) : Jumlah total proses (jumlah total operasi nol+jumlah total proses sekali) di semua n bit. Distribusi χ^2 adalah Distribusi referensi untuk statistik uji.

3. Uji deskripsi
   Hitung prates proposisi π dari satu urutan bit

$$\pi = \Sigma_j \varepsilon_j / n \quad (2)$$

   Uji frekuensi berhasil jika nilai dari $|\pi - 1/2| \geq \tau$, maka uji run tidak perlu dilakukan (pengujian seharusnya tidak dijalankan karena kegagalan lulus tes Frekuensi (Monobit)). Jika pengujian tidak dapat diterapkan, nilai-P diatur ke 0,0000.

$$\tau = 2 / \sqrt{n} \quad (3)$$

4. Hitung Uji statistic

$$V_n(obs) = \Sigma\_\{k=1\}\^\{n-1\} r(k) + 1 \quad (4)$$

   dimana r(k) = 0 jika $\varepsilon_k = \varepsilon_{k+1}$ dan lainnya r(k) = 1.

5. Hitung P-Value

$$P\text{-alue} = \text{erfc}[(V_n(obs) - 2n\pi(1 - \pi)) / (2\sqrt{2n\pi(1 - \pi)})] \quad (5)$$

   Tolak ukur keacakan dilihat dari P-value < 0,01 maka barisan tersebut tidak acak, sebaliknya jika P-value > 0,01 disimpulkan barisan tersebut adalah acak.

2.4 Block Bit

1. Fungsi Panggil
   Metode uji ini menentukan keacakan, berdasarkan frekuensi 1 pada M-bit blok merupakan bagian M/2. Blok frekuensi (M,n) dengan M sebagai panjang blok, n sebagai panjang string bit. $\varepsilon = \varepsilon_1, \varepsilon_2, ..., \varepsilon_n$ sebagai hasil dari setiap uji.
2. Uji Statistik
   Uji χ^2(obs) sebagai tolak ukur rasio kecocokan dengan M-bit yang dihasilkan (1/2), dengan referensi distribusi untuk uji statistiknya ialah distribusi χ^2.
3. Uji Deskripsi
   Deskripsi pengujian, N sebagai partisi urutan N = ⌊1/2⌋ blok tidak tumpang tindih serta membuang bit yang tidak digunakan.
   Persamaan proporsi π_i di setiap blok M-bit:

$$\pi = (1/M) \Sigma\_\{j=1\}\^\{i\} ((i-1)M + j) \quad (6)$$

   dimana $1 \leq i \leq N$

4. Uji statistik χ^2:

$$\chi^2\_obs = 4M (\Sigma\_\{i=1\}\^M (\pi\_i - 1/2)) \quad (7)$$

5. p-value:

$$p\text{-value} = \text{igamc}(N/2, \chi^2\_obs / 2) \quad (8)$$

$$\text{igamc}(x) = \int\_x\^\infty e\^\{-t\} t\^\{x-1\} dt \quad (9)$$

   olak ukur keacakan dilihat dari p-value < 0.01, maka barisan tersebut tidak acak, sedangkan p-value > 0.01, maka barisan tersebut adalah acak.

2.5 Mono Bit

Mono bit test mengukur proporsi bit 1 dalam urutan bit. Pengujian ini digunakan untuk memastikan bahwa dalam urutan bit acak, jumlah bit 1 mendekati setengah dari total bit dalam urutan tersebut.

1. Fungsi Panggil
   Pengujian frekuensi (Monobit) dapat menentukan keacakan pada barisan biner.

$$S\_n = X_1 + X_2 + ... + X_n \quad (10)$$

   dimana, X_i = 2ε_i - 1

2. Uji Statistik

$$s\_obs = |S\_n| / \sqrt{n} \quad (11)$$

3. Uji Deskripsi
   Menghitung p-value = erfc(s_obs / √2), erfc merupakan komponen error pada hasil definisinya:

$$\text{erfc}(x) = 2/\sqrt{\pi} \int\_x\^\infty e\^\{-u^2\} du \quad (12)$$

Journal of Information System Research (JOSH)
Volume 6, No. 1, Oktober 2024, pp 679−687
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v6i1.6112

Tolak ukur keacakan dilihat dari p-value < 0.01, maka barisan tersebut tidak acak, sedangkan p-value > 0.01, maka barisan tersebut acak.

2.6 Fungsi Pemotongan

Dalam proses ini, pemisah digunakan untuk membedakan antara nilai integer dalam konteks bilangan chaos, yang terkadang dapat ditemukan di antara bilangan riil, khususnya dalam rentang antara 0 dan 1. Setiap nilai chaos yang dihasilkan akan diolah secara berulang-ulang hingga mencapai nilai keluaran yang diinginkan. Setelah mendapatkan nilai tersebut, langkah selanjutnya adalah memisahkan hasil keluaran dan fokus pada bagian integer dari nilai yang telah diolah. Dengan cara ini, proses pemisahan tidak hanya memastikan bahwa kita mendapatkan komponen integer yang relevan, tetapi juga memfasilitasi analisis yang lebih mendalam terhadap karakteristik dan pola yang ada dalam data chaos tersebut. Hal ini penting untuk mengoptimalkan penggunaan bilangan chaos dalam berbagai aplikasi yang memerlukan presisi dan ketepatan dalam pengolahan data.

$$T(x, size) = |\lfloor x * 10\text{^}count \rfloor| \ , x \neq 0 \quad (13)$$

Nilai chaos x dikonversi menggunakan persamaan (13), dimana count dimulai dari 1 dan bertambah 1 sampai x * 10^count ≥ 10^{size-1}. Hasil akhirnya hanya diambil bagian integer-nya saja.

2.7 Nilai Korelasi

Relasi antara variabel bebas (x) dengan variabel tidak bebas (y) dipakai uji korelasi yang persamaannya:

$$r = (n\Sigma xy - \Sigma x\Sigma y) / \sqrt{[(n\Sigma x^2 - (\Sigma x)^2)(n\Sigma y^2 - (\Sigma y)^2)]} \quad (14)$$

Berikut adalah acuan koefisien korelasi:

Tabel 1. Nilai tingkat korelasi interval

No	Koefisien	Tingkat Hub
1	0.00-0.19	Sangat Kecil
2	0.20-0.39	Kecil
3	0.40-0.59	Cukup
4	0.60-0.79	Kuat
5	0.80-1.00	Sangat Kuat

Tabel 1 merupakan acuan sebuah algoritma dalam menyembunyikan data. Batas antara interval pada korelasi $-1 \leq r \leq 1$. Interval akan lebih optimal jika nilainya mendekati 0, entah interval bernilai positif maupun negatif. Korelasi yang bernilai negatif dapat langsung dimutlakkan dan tidak terlalu memberikan pengaruh, dikarenakan penilaian nilai korelasi dilihat sejauh mana nilai interval terhadap 0 sesuai persamaan korelasi.

3. HASIL DAN PEMBAHASAN

Algoritma A5/1 merupakan algoritma berbasis Linear Feedback Shift Register (LFSR) yang berperan penting dalam proses enkripsi dengan menggunakan tiga blok fungsi XOR. Algoritma ini bekerja dengan mengombinasikan keluaran dari beberapa register untuk membentuk aliran bit pseudo-acak yang berfungsi sebagai kunci dalam enkripsi data. Salah satu karakteristik utama dari A5/1 adalah adanya beberapa reposisi pada fungsi A1, di mana fungsinya diintegrasikan ke dalam operasi utama f pada model operasi XOR. Reposisi ini dimaksudkan untuk memperkuat struktur enkripsi, meningkatkan keamanan, dan membuat proses dekripsi menjadi lebih sulit tanpa mengetahui kunci enkripsinya. Dalam implementasinya, ketiga blok XOR pada algoritma ini bekerja secara sinkron untuk menghasilkan urutan bit acak yang digunakan dalam setiap sesi komunikasi. Fungsi A1 yang telah diposisikan ulang dalam operasi utama f bertujuan untuk menambah tingkat kompleksitas pada aliran keluaran, sehingga pola bit yang dihasilkan tidak dapat dengan mudah diprediksi. Pada dasarnya, setiap perubahan kecil pada masukan dapat menghasilkan perubahan besar pada keluaran, menjadikan algoritma ini cukup handal dalam menjaga kerahasiaan data yang ditransmisikan melalui jaringan. Reposisi pada fungsi A1 juga memberikan fleksibilitas yang lebih besar pada algoritma ini dalam hal penyesuaian terhadap dinamika masukan dan keluaran selama proses enkripsi berlangsung. Dengan demikian, reposisi ini tidak hanya meningkatkan efisiensi tetapi juga memberikan tingkat keamanan yang lebih tinggi dibandingkan dengan algoritma yang menggunakan pendekatan statis. Algoritma A5/1 dengan struktur fungsional yang dinamis ini memastikan bahwa komunikasi yang terjadi tetap terlindungi dari ancaman eksternal, khususnya dalam lingkungan jaringan yang rentan terhadap penyadapan atau intersepsi data. Selain itu, kombinasi dari fungsi XOR yang dikombinasikan dengan LFSR memungkinkan algoritma ini untuk memaksimalkan pengacakan dalam penyusunan bit. Hal ini menambah kerumitan enkripsi sehingga lebih sulit untuk diretas menggunakan serangan brute-force atau serangan kriptanalisis lainnya.

$$A1 = a_{13} \oplus a_{16} \oplus a_{17} \oplus a_{18} \quad (15)$$

Journal of Information System Research (JOSH)
Volume 6, No. 1, Oktober 2024, pp 679−687
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v6i1.6112

$$A2 = a_{20} \oplus a_{21} \quad (16)$$

$$A3 = a_7 \oplus a_{20} \oplus a_{21} \oplus a_{22} \quad (17)$$

Persamaan (15-17) menjelaskan operasi XOR yang terjadi pada setiap blok, dengan Ai dinyatakan pada fungsi umpan balik, dengan Ai = {1, 2, 3} untuk menghasilkan keluaran bit acak pada setiap bloknya. Dengan menerapkan operasi XOR pada elemen-elemen Ai, kita dapat menciptakan kombinasi yang menghasilkan variasi bit yang tidak terduga dalam setiap blok. Proses ini membantu memastikan bahwa keluaran yang dihasilkan memiliki sifat acak yang tinggi, yang sangat diperlukan dalam banyak aplikasi, termasuk kriptografi dan pengolahan sinyal. Setiap blok yang melalui proses ini akan memiliki karakteristik unik yang dihasilkan dari kombinasi bit yang berbeda, berkat fungsi umpan balik yang melibatkan nilai-nilai Ai tersebut. Dengan demikian, hasil akhirnya adalah keluaran bit acak yang dapat meningkatkan keamanan dan keandalan sistem yang bergantung pada data tersebut.

$$f = A1 \oplus A2 \oplus A3 \quad (18)$$

Persamaan (18) menyatakan sebagai fungsi utama dalam menggabungkan setiap fungsi umpan balik Ai.

3.1 Pembangkit Bilangan Acak

Proses perhitungan nilai inisialisasi dimulai dengan langkah konversi masukan yang berupa plaintext menjadi kode ASCII. Dalam tahap ini, setiap karakter dari plaintext diubah menjadi representasi angka yang sesuai dengan standar ASCII, yang memudahkan pemrosesan selanjutnya. Setelah kode ASCII dihasilkan, langkah berikutnya adalah mengonversi kode tersebut menjadi bilangan biner. Proses ini sangat penting karena bilangan biner adalah format yang umum digunakan dalam berbagai operasi komputasi, termasuk dalam algoritma kriptografi. Untuk memastikan bahwa setiap fungsi umpan balik f_i dapat dipisahkan dengan jelas, digunakan teknik padding yang berfungsi sebagai pembatas. Padding ini membantu menjaga struktur blok data agar tetap teratur, sehingga setiap blok dapat dilakukan pemilihan bit yang akan dijadikan sebagai pembangkit bit acak dengan menggunakan operasi XOR. Setelah bit acak dihasilkan untuk masing-masing blok, proses selanjutnya mengikuti langkah-langkah yang ditentukan oleh Persamaan (13). Pada tahap ini, bit acak yang telah diperoleh digunakan dalam perhitungan lebih lanjut, yang berkontribusi pada kompleksitas dan keamanan dari proses kriptografi yang sedang berlangsung. Dengan metode ini, setiap blok tidak hanya mempertahankan elemen keacakan, tetapi juga memastikan bahwa hasil akhir dari proses perhitungan tetap aman dan sulit untuk diprediksi. Penggunaan padding dan operasi XOR dalam pembangkitan bit acak menjadi kunci dalam menciptakan algoritma yang kuat dan efisien.

3.2 Pengujian Keacakan

Microsoft Office Excel digunakan sebagai alat bantu proses uji, terdapat batas pada masukan plainteks dengan panjang 8 karakter atau 64 bit. Apabila panjang karakter kurang dari 8 secara otomatis panjang kekurangan karakter akan di isi ”Spasi”, akan tetapi jika lebih dari 8 karakter secara otomatis karakter ke 9 dan seterusnya dilupakan. Pengujian ini menggunakan metode statistik, antara lain Mono Bit, Block Bit, dan Runs Test. Nilai keluaran dalam dinyatakan ”tidak acak” apabila ≤ 0.01 dan untuk ≥ 0.01 dinyatakan ”Acak” (AC).

Tabel 2. 10 Terbaik

No	Algoritma	p-value	Hasil
1)	L1105	0.69725037	“AC”
2)	L401	0.671025191	“AC”
3)	L2015	0.669059667	“AC”
4)	L2018	0.656004083	“AC”
5)	L2116	0.653523574	“AC”
6)	L907	0.651809705	“AC”
7)	L1707	0.651462567	“AC”
8)	L1916	0.647039165	“AC”
9)	L1600	0.643445309	“AC”
10)	L1716	0.637366173	“AC”
	L2120	0.555686361	“AC”

Tabel (2) merupakan rata-rata p value dari pengujian Runs Test, Mono Bit, dan Block Bit. Kemudian diambil 10 terbaik yang akan digunakan sebagai penguji enkripsi, pada hasil 10 terbaik didapatkan nilai keluaran lebih tinggi dari algoritma acuan (L2120). pada tabel 2, 3, 4 akan menjadikan tabel (1) sebagai acuan.

Tabel 3. 10 Hasil Pengujian Runs Test

No	Algoritma	p-value	Hasil
1)	L1105	0.655152055	“AC”
2)	L401	0.660147117	“AC”

Journal of Information System Research (JOSH)
Volume 6, No. 1, Oktober 2024, pp 679−687
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v6i1.6112

No	Algoritma	p-value	Hasil
3)	L2015	0.67912007	“AC”
4)	L2018	0.574181903	“AC”
5)	L2116	0.619626264	“AC”
6)	L907	0.608093271	“AC”
7)	L1707	0.547159438	“AC”
8)	L1916	0.63468669	“AC”
9)	L1600	0.62261367	“AC”
10)	L1716	0.583004744	“AC”
	rata-rata	0.618378522	“AC”

Tabel (3) menyatakan p value pada hasil uji Runs Test dengan nilai rata-rata 0.618378522.

Tabel 4. Hasil dari 10 Pengujian Mono Bit

No	Algoritma	p-value	Hasil
1)	L1105	0.826500041	“AC”
2)	L401	0.757100057	“AC”
3)	L2015	0.700920004	“AC”
4)	L2018	0.80501998	“AC”
5)	L2116	0.757100057	“AC”
6)	L907	0.791800049	“AC”
7)	L1707	0.839719972	“AC”
8)	L1916	0.679439943	“AC”
9)	L1600	0.649732925	“AC”
10)	L1716	0.748839927	“AC”
	rata-rata	0.755617295	“AC”

Table (4) menghasilkan rata-rata Mono Bit dengan besaran p value 0.755617295.

Tabel 5. Hasil dari 10 Pengujian Block Bit

No	Algoritma	p-value	Hasil
1)	L1105	0.610099016	“AC”
2)	L401	0.595828401	“AC”
3)	L2015	0.627138927	“AC”
4)	L2018	0.588810367	“AC”
5)	L2116	0.583844399	“AC”
6)	L907	0.555535794	“AC”
7)	L1707	0.567508292	“AC”
8)	L1916	0.626990863	“AC”
9)	L1600	0.657989331	“AC”
10)	L1716	0.580253848	“AC”
	rata-rata	0.599399924	“AC”

Tabel (5) pada percobaan Block Bit nilai p value menghasilkan keluaran dengan rata-rata 0.599399924.

3.3 Pengujian Keacakan

Proses uji enkripsi memiliki batas 32 karakter atau 256 bit, yang akan diubah ke dalam kode ASCII, operasikali ini menggunakan modulus 256 sebagai domainnya dalam pengujian Enkripsi E_k dan Deskripsi D_k.

$$E_k : P \oplus K = C, \qquad D_k : C \oplus K = P \tag{19}$$

Dalam proses uji enkripsi digunakan tiga masukan ciphertext yang diharapkan dapat mewakili variasi plainteks yang biasanya digunakan oleh user. Antara lain,ciphertext pertama yaitu Ibu memasak nasi goreng di dapur, untuk mewakili tulisan biasa yang hanya terdiri dari kombinasi huruf. ciphertext kedua Xg7l2yvkR_mzWbˆ ˆoQf1Jh8aTc*5dPn digunakan kombinasi huruf, simbol, dan angka. ciphertext terakhir adalah GGGGGGGGGGGGGGGGGGGGGGGGGGGGG- GT yang mempunyai huruf yang sama.

Hasil uji korelasi digunakan untuk mengevaluasi hubungan antara cipherteks dan plainteks[20]. Ini dilakukan untuk mengetahui seberapa baik kunci dapat menyamarkan plainteks dan menghilangkan hubungan antara keduanya. Hasil pengujian untuk ketiga plainteks dengan setiap data set diberikan pada Tabel 5.

Tabel 6. 10 Hasil Uji Korelasi

No	Algoritma	Uji Plainteks			Rata- Rata	Tingkat Korelasi
		Plainteks 1	Plainteks 2	Plainteks 3		
1	L1105	-0.0791065573	0.0536663475	0.0160611218	-0.00312636265	“Sangat Kecil”
2	L401	-0.00552138204	-0.0479740015	-0.0583653367	-0.0372869067	“Sangat Kecil”

Journal of Information System Research (JOSH)
Volume 6, No. 1, Oktober 2024, pp 679−687
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v6i1.6112

No	Algoritma	Uji Plainteks			Rata- Rata	Tingkat Korelasi
		Plainteks 1	Plainteks 2	Plainteks 3		
3	L2015	0.109137513	0.0117155058	0.0178103923	0.0462211373	“Sangat Kecil”
4	L2018	0.0842099542	0.0565328818	-0.0590977002	0.0272150452	“Sangat Kecil”
5	L2116	0.0104899774	0.0390152729	-0.0512419746	-0.000578908098	“Sangat Kecil”
6	L907	-0.124469115	0.0139834417	-0.0315269695	-0.0473375478	“Sangat Kecil”
7	L1707	-0.137821672	0.000831865725	0.046683191	-0.0301022051	“Sangat Kecil”
8	L1916	0.00357317658	0.0277879610	-0.0266656567	0.0015651602	“Sangat Kecil”
9	L1600	0.0359920778	0.0452246544	-0.0141379171	0.0223596050	“Sangat Kecil”
10	L1716	0.00607874040	0.0325805648	-0.0559050021	-0.0057485656	“Sangat Kecil”

Tabel (6) Menjelaskan bahwa rata-rata interval dari algoritma L1105 sampai dengan L1716 pada tabel (6) mendekati 0 dengan tingkat korelasi yang “Sangat Kecil”.

4. KESIMPULAN

Penelitian ini berfokus pada pengembangan metode reposisi yang bertujuan untuk menghasilkan kunci dan keluaran "acak" yang lebih aman dalam konteks kriptografi. Pendekatan ini dirancang untuk meningkatkan keamanan data dengan menerapkan teknik pengacakan yang lebih efektif, sehingga dapat memberikan perlindungan yang lebih baik terhadap informasi sensitif. Salah satu pencapaian utama dari penelitian ini adalah pengurangan korelasi antara kunci dan keluaran yang dihasilkan, berkat penerapan proses reposisi pada setiap algoritma yang diuji. Proses ini menghasilkan bilangan "acak" dengan tingkat korelasi yang sangat rendah, yang merupakan indikator penting dalam menilai kualitas bilangan acak. Dalam penelitian ini, pengujian terhadap bilangan acak yang dihasilkan dilakukan dengan menggunakan Microsoft Excel sebagai alat bantu analisis. Berbagai metode pengujian diterapkan, termasuk Runtest, Monobit, dan Blockbit, yang semuanya dirancang untuk mengukur efektivitas dari teknik pengacakan yang digunakan. Hasil pengujian menunjukkan bahwa rata-rata p-value dari setiap algoritma yang diuji kurang dari 0,01, menandakan bahwa bilangan acak yang dihasilkan memiliki kualitas yang tinggi. Setelah memperoleh hasil dari metode pengacakan ini, penelitian melanjutkan dengan menghitung rata-rata dari setiap metode dan memilih sepuluh hasil terbaik untuk diuji lebih lanjut dalam proses enkripsi. Proses uji enkripsi ini melibatkan tiga plainteks yang berbeda, dengan harapan bahwa tingkat korelasi yang dihasilkan dari enkripsi tersebut akan sangat rendah. Berdasarkan analisis yang dilakukan, penelitian menyimpulkan bahwa metode reposisi yang diterapkan berhasil menunjukkan potensi signifikan dalam meningkatkan keamanan sistem informasi. Dengan menerapkan pengacakan yang lebih efektif, metode ini mampu menghasilkan keluaran yang lebih aman, sehingga mengurangi risiko potensi ancaman terhadap data sensitif. Peningkatan keamanan ini sangat penting dalam era digital saat ini, di mana perlindungan informasi menjadi semakin krusial. Dengan hasil yang menunjukkan kualitas tinggi dan korelasi rendah antara kunci dan keluaran, penelitian ini berkontribusi pada pengembangan solusi kriptografi yang lebih canggih dan dapat diandalkan.

REFERENCES

[1] A. D. Wowor and B. Susanto, “One to many (new scheme for symmetric cryptography),” TELKOMNIKA (Telecommunication Comput. Electron. Control., vol. 21, no. 4, pp. 762–770, 2023.
[2] R. Rahman et al., Buku Ajar Keamanan Jaringan Komputer. PT. Sonpedia Publishing Indonesia, 2024. [Online]. Available: https://books.google.co.id/books?id=fxUFEQAAQBAJ
[3] A. Ramadhani and A. D. Wowor, “Implementasi Variasi Fungsi XOR dalam Pembangkitan Kunci LFSR pada Skema A5/1 dengan Tiga Blok Bit,” JIKO (Jurnal Inform. dan Komputer), vol. 8, no. 1, pp. 161–173, 2024.
[4] M. Azhari, D. I. Mulyana, F. J. Perwitosari, and F. Ali, “Implementasi Pengamanan Data pada Dokumen Menggunakan Algoritma Kriptografi Advanced Encryption Standard (AES),” J. Pendidik. Sains dan Komput., vol. 2, no. 01, pp. 163–171, 2022.
[5] N. G. Najoan and M. A. I. Pakereng, “Modifikasi Skema Teknik Tanam Padi dan Bajak Sawah Berbasis Square Transposition 64-bit,” J. Tek. Inform. dan Sist. Inf., vol. 7, no. 1, 2021.
[6] A. J. Herman, “Desain Pembangkit Kunci LFSR dengan Skema A5/1 Menggunakan 7 Blok Bit Fungsi XOR,” 2022.
[7] R. V. Manullang, “Desain Lima Fungsi Umpan Balik LFSR dengan Skema A5/1 sebagai Pembangkit Kunci Kriptografi Stream Cipher,” 2022.
[8] O. K. Sulaiman, “Generate Pseudo-Random Numbers Linear-Feedback Shift Register (LSFR) Pada Kunci Algoritma One Time Pad (OTP),” in Seminar Nasional Teknologi Komputer & Sains (SAINTEKS), 2020, pp. 171–175.
[9] B. D. Kurniawan, M. A. Rosid, I. A. Kautsar, and N. E. Pratama, “Rancang Bangun Library Web Token untuk Enkripsi HTTP Data Menggunakan Eksklusif-OR (XOR),” Phys. Sci. Life Sci. Eng., vol. 1, no. 1, p. 14, 2023.
[10] R. S. Durga, C. K. Rashmika, O. N. V Madhumitha, D. G. Suvetha, B. Tanmai, and N. Mohankumar, “Design and synthesis of lfsr based random number generator,” in 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), IEEE, 2020, pp. 438–442.
[11] G. Jonatan, I. Syafalni, N. Sutisna, R. Mulyawan, and T. Adiono, “Gaussian Pseudo-Random Number Generator using LFSR’s Rotation and Split,” in 2021 International Symposium on Electronics and Smart Devices (ISESD), IEEE, 2021, pp. 1–5.
[12] N. A. Padjalo, “Implementasi 3 Blok Bit Fungsi XOR untuk Modifikasi Skema LFSR A5/1 dalam Pembangkitan Kunci

Journal of Information System Research (JOSH)
Volume 6, No. 1, Oktober 2024, pp 679−687
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v6i1.6112

Kriptografi," 2023.
[13] S. Angelina, "Optimasi Pembangkit Bilangan Acak Dengan Fungsi Polinomial Dan Kombinasi Metode Iterasi," 2024.
[14] O. N. E. Choesni, "Implementasi 30x^ 3-19x-6 Sebagai Fungsi Pembangkit Bilangan Acak Menggunakan Metode Newton Raphson," 2022.
[15] H. Nathanael and A. D. Wowor, "Chaos CSPRNG Design As a Key in Symmetric Cryptography Using Logarithmic Functions," Komputasi J. Ilm. Ilmu Komput. dan Mat., vol. 21, no. 1, pp. 83–91, 2024.
[16] R. J. Belora, "Implementasi 3x3+ 5x2+ 7x–20 Sebagai Fungsi Generator Menggunakan Meto De Newton Raphson," 2021.
[17] M. Permadi, "Penggunaan Meto De Fixed Point Ite Ration Dalam Menentukan Pembangkit Bilangan Acak Menggunakan Fungsi Kuadrat X2− 2x–4," 2021.
[18] C. R. Irianto, "Implementasi Polinomial dan Trigonometri Sebagai Fungsi Komposisi dalam Pembangkit Bilangan Acak Berbasis CSPRNG Chaos," 2022.
[19] R. P. Prajapat, R. Bhadada, and G. Sharma, "Implementation of Enhanced A5/1 Stream Cipher and its Randomness Analysis by NIST Test Suite," in 2021 IEEE International Symposium on Smart Electronic Systems (iSES), IEEE, 2021, pp. 426–431.
[20] T. T. M. Bulamey and H. Hendry, "Perancangan Kriptografi Block Cipher Menggunakan Pola Logo Media Sosial," J. Sist. Komput. dan Inform., vol. 2, no. 2, pp. 115–122, 2021.