Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 303−311
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1639

Analisis Perbandingan Algoritma NGG dan GGHN pada Frekuensi Hasil Enkripsi

Farid Akbar Siregar[1], Ade Rizka[2], Annisa Fadillah Siregar[3,*]

[1]Ilmu Komputer dan Teknologi Informasi, Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara, Medan, Indonesia
[2]Sains dan Teknologi, Sistem Komputer, Universitas Pembangunan Panca Budi, Medan, Indonesia
[3]Ilmu Komputer dan Teknologi Informasi, Teknik Informatika, Universitas Budi Dharma, Medan, Indonesia
Email: [1]faridakbar@umsu.ac.id, [2]aderizka@dosen.pancabudi.ac.id, [3,*]annisaf@univ-bd.ac.id
Email Penulis Korespondensi: annisaf@univ-bd.ac.id
Submitted: **06/06/2022**; Accepted: **28/06/2022**; Published**: 30/06/2022**

Abstrak−Data dan informasi dalam perkembangan teknologi digital memiliki peranan penting. Setiap kegiatan maupun aktifitas yang menggunakan teknologi digital berkaitan dengan data dan informasi, sehingga keamanan informasi maupun kerahasiaan data sangat penting. Untuk menjaga keamanan informasi dan kerahasiaan data dibutuhkan perlindungan dengan teknik kriptografi. Teknik kriptografi berhubungan dengan enkripsi yaitu dimana dilakukan proses pengacakan data dan menyembunyikan data dengan sistem kunci sedangkan dekripsi yaitu proses perubahan kondisi data ke bentuk aslinya agar mudah dipahami. Terdapat kendala maupun masalah dalam komunikasi digital, sehingga diperlukan teknik kriptografi yang memiliki tingkat keamanan yang lebih dan dapat diterapkan dalam komunikasi digital. Untuk mengetahui tingkat keamanan dalam teknik kriptografi diperlukan analisis frekuensi. Analisis frekuensi pada algoritma NGG dan GGHN dilakukan untuk mengetahui tingkat keamanan informasi berdasarkan hasil enkripsi data. Berdasarkan proses pengujian pada algoritma NGG dan GGHN akan diketahui frekuensi karakter pada teks yang bervariasi. Semakin banyak karakter yang digunakan pada kunci akan mempengaruhi tingkat keamanan informasi. Algoritma NGG memiliki tingkat keamanan yang lebih tinggi dibandingkan dengan algortima GGHN dengan selisih presentasi sebesar 0.000299967%. Jika frekuensi kemunculan karakter pada teks pesan yang telah di enkripsi semakin sering atau semakin tinggi, maka tingkat keamanan informasi pada pesan lebih rendah dan kata kunci lebih mudah dipecahkan.

Kata Kunci: Kriptografi; NGG; GGHN; Enkripsi; Dekripsi

Abstract−Data and information in the development of digital technology have an important role. Every activity or activity that uses digital technology is related to data and information, so information security and data confidentiality are very important. To maintain information security and data confidentiality, protection with cryptographic techniques is needed. Cryptographic techniques are related to encryption, which is where the process of scrambling data is carried out and hiding data with a key system, while decryption is the process of changing the condition of the data to its original form so that it is easy to understand. There are obstacles and problems ini digital communication, so cryptographic techniques are needed that have higher level of security and can be applied in digital communications. To determine the level of security in cryptographic techniques required frequency anlysis. Frequency analysis on the NGG and GGHN algorithms is carried out to determine the level of information security based on the results of data encryption. BAse on the testing process on the NGG and GGHN algorithms, it will be known that the frequency of characters in the text varies. The more characters used in the key will affect the level of information security. The NGG algorithm has a higher level of security than the GGHN algorithm with a precentage difference of 0.000299967%. If the frequency of occurrence of characters in the message text that has been encrypted is more frequent or higher, then the level of information security in the message is lower and the password is easier to crack.

Keywords: Cryptography; NGG; GGHN; Encryption; Decryption

1. PENDAHULUAN

Data dan informasi dalam perkembangan teknologi digital memiliki peranan penting. Setiap kegiatan maupun aktifitas yang menggunakan teknologi digital berkaitan dengan data dan informasi, sehingga keamanan informasi maupun kerahasiaan data sangat penting. Dalam komunikasi digital terdapat proses transaksi dan distribusi data maupun informasi yang menjadi rutinitas. Kendala dan masalah muncul dalam komunikasi digital sehingga diperlukan teknik kriptografi yang memiliki tingkat keamanan yang lebih dan dapat diterapkan untuk menjaga keamanan informasi dan kerahasiaan data. Teknik kriptografi mampu melindungi hak akses keamanan informasi dan kerahasiaan data, proses transaksi dalam pengiriman dan penerimaan data terjamin tidak ada perubahan dan keaslian berdasarkan sumber data. Teknik kriptografi berhubungan dengan enkripsi yaitu dilakukan proses pengacakan data asli atau merubah pesan asli dan menyembunyikan data dengan sistem kunci sedangkan dekripsi yaitu proses perubahan kondisi data ke bentuk aslinya atau mengembalikan pesan asli agar mudah dipahami.

Teknik kriptografi yang dapat digunakan yaitu simetri dan asimetris. Pada simetri, kunci kriptografi saat proses enkripsi dan dekripsi menggunakan kunci yang sama yaitu disebut *Private Key*. Pada asimetris, kunci kriptografi saat proses enkripsi dan dekripsi menggunakan kunci yang berbeda yaitu pada umumnya dalam proses dekripsi menggunakan *Private Key* dan proses enkripsi menggunakan *Public Key*. Algoritma NGG dan GGHN merupakan teknik kriptografi simetris yaitu menggunakan kunci yang sama dalam proses enkripsi dan dekripsi [1]. Pada dasarnya kriptografi bertujuan untuk menyediakan privasi dalam komunikasi dua entitas serta menyediakan otentikasi antara satu entitas dengan yang lainnya [2].

Untuk mengetahui tingkat keamanan dalam teknik kriptografi diperlukan analisis frekuensi. Analisis frekuensi pada kriptografi merupakan tentang memprediksi probabilitas huruf ataupun kelompok huruf *ciphertext*. Dalam kriptografi pada setiap rentang pesan berupa tulisan huruf atau kombinasi huruf memiliki frekuensi yang beragam.

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 303−311
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1639

Pada proses distribusi terdapat kemunculan karakter yang sama untuk semua data maupun informasi yang digunakan. Analisis frekuensi akan menghitung jumlah karakter dari tulisan huruf atau kombinasi huruf dalam setiap proses enkripsi agar kemampuan kunci kriptografi dapat dianalisis dalam keamanan informasi.

Teknik kriptografi dengan menggunakan algoritma NGG dan GGHN memiliki kemampuan dan tingkat keamanan yang berbeda. Analisis frekuensi pada kedua algoritma akan menghasilkan nilai kinerja, sehingga dalam proses transaksi dan distribusi data maupun informasi dapat ditentukan algoritma yang memiliki kemampuan dan tingkat keamanan yang lebih dibutuhkan. Pada penelitian ini akan menganalisis frekuensi kedua algoritma tersebut. Hasil proses pengujian dapat dilihat dari jumlah kemunculan karakter yang sama pada proses enkripsi. Diharapkan dapat meningkatkan keamanan informasi dalam teknologi digital.

Hal pertama yang tercatat dari teknik kriptografi secara harfiah ditulis di batu hampir empat milenium yang lalu oleh seorang juru tulis Mesir dengan menggunakan subtitusi simbol *hieroglyphic* dalam tulisannya di dinding batu makam seorang bangsawan [3].

Analisis pada keamanan dapat diusulkan serangan yang dibedakan berdasarkan bias jangka pendek dan jangka panjang pada *keystream*. Pembeda dapat digunakan untuk membedakan *keystream* yang dihasilkan beberapa kunci dan untuk membedakan *keystream* yang dihasilkan oleh pasangan kunci [4]. Pada RC4 *stream cipher*, dalam *stream* awal yang mengeksploitasi fitur array dalam menghasilkan *pseudorandom* bit dengan menggunakan beberapa operasi sederhana [5]. Karena struktur RC4 cukup aman jika *cipher* digunakan untuk tindakan pencegahan yang sesuai, perubahan rancangan yang tidak memiliki izin dapat mengakibatkan kerentanan yang potensial [6]. GGHN merupakan *stream cipher* yang mirip dengan RC4 dirancang untuk menggunakan prosesor 32-bit yaitu 3-5 kali lebih cepat dari RC4. Salah satu sumber keamanan tinggi GGHN yaitu ukuran besar keadaan internal rahasianya dengan total 8240 bit [7]. GGHN yaitu *stream cipher* yang relatif lebih efisien yang terinspirasi dari desain atau rancangan RC4 [8].

Pada penelitian S. An-nissa, H. Mawengkang dan S. Efendi (2022), algoritma RC4 dan GGHN untuk pengamanan pesan, algortima RC4 mengolah unit dan memasukan data dalam satu waktu dengan kombinasi panjang karakter pesan dan kunci yang berbeda-beda. Hasil dalam proses menyatakan bahwa panjang kunci dalam proses enkripsi tidak mempengaruhi kemampuan dalam waktu proses. Panjang karater pesan memiliki pengaruh yang besar dalam lama waktu proses enkripsi dan dekripsi, jika semakin banyak jumlah karakter maka semakin lama waktu proses. Hal tersebut mempengaruhi proses dan tingkat keamanan informasi [9].

Pada penelitian F. Akbar, H. Mawengkang dan S. Efendi (2018), analisis perbandingan algoritma RC4+, RC4 NGG dan RC4 GGHN pada keamanan file gambar, berdasarkan tinjauan pustaka algoritma RC4+, RC4 NGG lebih efisien dan lebih cepat dibandingkan algoritam RC4. Dengan melakukan perbandingan waktu proses dan kompleksitasnya. Dilakukan dua proses pengujian pada setiap algoritma yaitu, mengenkripsi gambar dengan panjang kunci yang berbeda dan ukuran gambar sama serta mengenkripsi gambar dengan panjang kunci yang sama dan ukuran gambar yang berbeda. Hasilnya yaitu algoritma RC4 NGG merupakan algoritma yang lebih cepat sebesar 13,3% dari RC4+ dan sebesar 10,2 % lebih cepat dibandingkan RC4 GGHN. Panjang kunci tidak mempengaruhi waktu proses namun ukuran gambar mempengaruhi waktu proses [10].

Pada penelitian S. Suhandinata, R. A. Rizal, D. O. Wijaya, P. Warren dan S. Srinjiwi (2019), analisis performa kriptografi Hybrid, algoritma Blowfish dan algoritma RSA, dengan beberapa jenis data yaitu dokumen, gambar, audio dan video. Hasilnya yaitu algoritma Hybrid memiliki performa yang tidak terlalu berbeda dengan algoritma Blowfish dan dalam proses enkripsi dan dekripsi lebih terjamin keamanannya dengan kelebihan algoritma RSA. Untuk rata-rata nilai performa pada algoritma Hybrid yaitu proses enkripsi dokumen sebesar 0,85 detik, gambar sebesar 1,06 detik, audio sebesar 3,38 detik dan video sebesar 15,56 detik dan proses dekripsi dokumen sebesar 1,01 detik, gambar sebesar 1,38 detik, audio sebesar 4,3 detik dan video sebesar 27,56 detik. Dengan algoritma Hybrid memiliki proses enkripsi dan dekripsi yang lebih cepat namun tidak secepat algoritma pembentuknya [11].

Pada penelitian *Quad -RC4: Merging Four RC4 States towards a 32 bit Stream Cipher* oleh G. Paul dan S. Maitra (2012), dalam menggabungkan yang terbaik dari keduanya dengan mempertahakan struktur dasar RC4 dengan keamanan yang terjamin dan menggabungkan 4 status RC4 untuk membuat aliran *stream cipher* yaitu *Quad-RC4* dengan hasil output 32-bit pada tiap putaran. Ketentuan penyimpanan dalam keadaan internal yaitu 1024 bit. *Cipher* memiliki kinerja yang lebih cepat dibandingkan RC4 normal dan sebanding dengan HC-128, *stream cipher* lebih cepat diantara eSTREAM [12].

Pada penelitian S. Banik, S. Maitra dan S. Sarkar (2011) tentang evolusi GGHN *Cipher*, *cipher* dimotivasi dari RC4 dengan tujuan untuk mendapatkan percepatan melalui pertimbangan output *keystream* berorientasi kata daripada berorientasi *byte*. Dapat dibuktikan bahwa terdapat sejumlah siklus pendek dengan panjang sama yang panjang *array* keadaan digunakan dalam *cipher*. Analisis secara teori, mengenai evolusi tipe GGHN, dipelajari model acak dari kata kunci. Menggunakan proses Markovian, ditunjukan bahwa model berkembang ke seluruh keadaan nol lebih cepat daripada rencana yang diharapkan [13].

Pada penelitian *On the Weak State in GGHN-like Ciphers* Oleh A. Kircanski, E. Al-Zaidy dan A. M. Youssef (2009), GGHN merupakan *stream cipher* yang relatif lebih efisien yang terinspirasi dari RC4. Terdapat aspek yang menunjukan tantangan dari prinsip desain terakhir. Secara khusus dinilai algoritma mirip GGHN dengan status yang lemah, dimana semua karakter status internal dan elemen output genap. GGHN diserap dalam keadaan lemah, bit tidak signifikan dari karakter *plaintext* tapi akan dapat terdeteksi dengan melihat *ciphertext*. Dengan model algoritma *Markov chain* dan menghitung waktu penyerapan, maka jumlah rata-rata langkah yang diperlukan algoritma untuk

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 303−311
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1639

menginput dalam keadaan lemah dapat lebih rendah dari yang diharapkan pada proses awal dan karena hal tersebut harus lebih hati-hati saat memperkirakan nilainya [14].

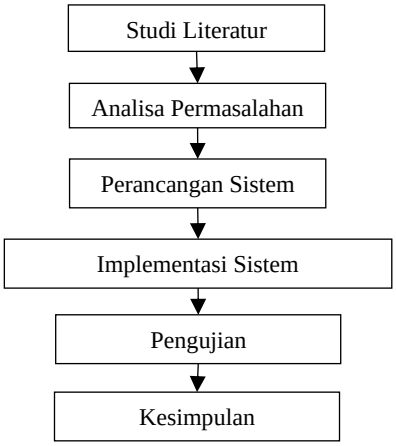
Pada penelitian analisis keamanan RC4+ *Stream Cipher* oleh S. Banik, S. Sarkar dan R. Kacke (2013), diklaim bahwa RC4+ mengatasi sebagian besar kelemahan dari RC4 dan sedikit lebih lambat dari RC4 pada *software*. Pemasangan serangan agar membedakan RC4+ berdasarkan bias byte output awal. Pembeda membutuhkan kisaran 226 sampel yang merupakan hasil kesalahan diferensial pada RC4 yang diusulkan oleh peneliti sebelumnya di RC4+. Hasilnya menunjukan bahwa RC4+ rentan dengan serangan kesalahan diferensial dan kemungkinan untuk pemulihan semua keadaan internal *cipher* pada awal PRGA dengan memperbaiki kisaran 217,2 kesalahan [15].

Dalam penelitian ini, analisis frekuensi pada metode NGG dan GGHN dilakukan untuk mengetahui tingkat keamanan informasi pada proses transaksi maupun distribusi. Tujuan dari pengujian algoritma berdasarkan analisis frekuensi masing-masing agar dapat mengetahui kinerja maksimal masing-masing algoritma dengan hasil probabilitas huruf pada setiap pesan enkripsi sehingga dapat diketahui algoritma yang memiliki kemampuan dan tingkat keamanan yang lebih.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Tahapan yang akan dilakukan dalam pelaksanaan penelitian analisis frekuensi algoritma NGG dan GGHN memiliki langkah-langkah yang terdapat pada Gambar 1.



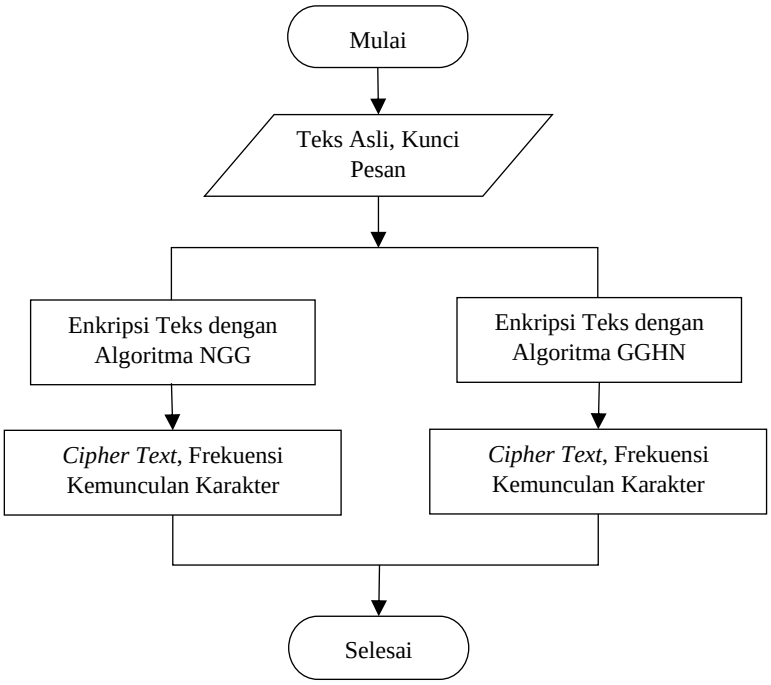
Gambar 1. Kerangka Kerja Penelitian

a. Studi Literatur
   Studi literatur yang dilakukan pada penelitian ini yaitu mengumpulkan bahan refrensi mengenai keamanan informasi dengan menggunakan algoritma NGG dan GGHN dari berbagai sumber yang terpercaya untuk melengkapi informasi sehingga memiliki landasan teori dan ilmu yang sesuai.
b. Analisa Permasalahan
   Pada tahap ini dilakukan analisis terhadap hasil studi literatur untuk mengetahui dan mendapatkan pemahaman mengenai algoritma NGG dan algoritma GGHN.
c. Perancangan Sistem
   Pada tahap perancangan sistem dilakukan perancangan arsitektur, pengumpulan data pelatihan, merancang antarmuka. Proses perancangan dilakukan berdasarkan hasil analisis studi literatur yang telah didapatkan.
d. Implementasi Sistem
   Pada tahap implementasi sistem ini dilakukan penerapan algoritma NGG dan algoritma GGHN pada sistem untuk melakukan enkripsi dan dekripsi pada pesan teks.
e. Pengujian
   Pada tahap ini dilakukan pengujian sistem terhadap hasil enkripsi pesan teks dengan menganalisis frekuensi kemunculan karakter pada pesan teks.
f. Kesimpulan
   Kesimpulan merupakan hasil pembahasan dari proses penerapan dan pengujian sistem berdasarkan analisa permasalahan yang ingin diselesaikan.

2.2 Tahapan Metode Penelitian

Tahapan metode penelitian pada analisis frekuensi algoritma NGG dan GGHN dapat dilihat pada Gambar 2.

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 303−311
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1639



Gambar 2. Tahapan Metode Penelitian

Dalam tahapan metode penelitian ini dimulai dari menginput teks asli dan kunci pesan yang merupakan *random string*, selanjutnya dilakukan proses enkripsi teks asli dengan menggunakan algoritma NGG dan kunci pesan yang sudah ditentukan, dan secara terpisah dilakukan proses enkripsi juga dengan menggunakan algoritma GGHN dan kunci pesan yang sudah ditentukan, hasil output teks dari masing-masing algoritma adalah berupa teks yang telah terenkripsi dan jumlah frekuensi kemunculan masing-masing karakter pada pesan teks.

2.3 Analisis Frekuensi

Pada substitusi sederhana, huruf *plaintext* diganti dengan huruf berbeda dan huruf tertentu *plaintext* akan diubah menjadi huruf yang sama dalam *ciphertext*. Jikan munculnya huruf a menjadi huruf Y maka, pesan *ciphertext* yang terdapat banyak huruf Y akan disarankan ke *cryptanalyst* Y mewakili a.

Dasar dari analisis frekuensi yaitu menghitung frekuensi huruf *ciphertext* dan dikaitkan dengan huruf *plaintext* yang diterka. Dimana lebih banyak Ys di *ciphertext* menunjukan bahwa Y berhubungan dengan a pada *plaintext*, namun tidak pasti karena huruf lain juga sangat umum, maka Y bisa menjadi salah satu bagian huruf. Perubahan menjadi *plaintext* lain juga tidak mungkin. *Cryptanalyst* harus mencoba kombinasi dalam pemetaan antara hururf *ciphertext* dan *plaintext.*

Statistik yang kompleks dapat digunakan agar mudah dipahami, dengan mempertimbangkan jumlah pasangan huruf (bigram), tiga seragam (trigram) dan selanjutnya. Maka hal tersebut dapat dilakukan agar memberikan informasi yang lebih banyak ke *cryptanalist* [16]. Untuk mengukur tingkat fekuensi kemunculan karakter pada masing-masing algoritma yaitu dengan menggunakan persamaan berikut:

$$\text{Frekuensi Relatif} = \frac{\text{Jumlah Karakter}}{\text{Total Seluruh Karakter}} \quad (1)$$

2.4 Algoritma NGG dan GGHN

Terdapat *cipher* baru yang dikembangkan dengan cara memperluas RC4 menjadi 32 bit yaitu Sheet Bend dan Bowline. Generalisasi RC4 diusulkan bertujuan untuk memperluas RC4 menjadi 32 atau 64 bit dimana ukuran state menjadi lebih kecil yaitu 232 atau 264. Algoritma tersebut yaitu RC4 (n, m), dimana N=2n merupakan ukuran array state dalam kata-kata, m yaitu ukuran kata dalam bit, n ≤ m. NGG merupakan adopsi dari inisial para perancang algoritma. Pada NGG KSA dan PRGA diperbaharui indeks i, j dengan cara yang sama dengan RC4 KSA dan PRGA.

Pada NGG KSA, larik S diinisialisasi ke larik acak yang ditentukan sebelumnya a. Selanjutnya, S [i] dan S[j] diganti dan jumlah dari kedua elemen ini (mod M=2m) ditetapkan ke S [i] [1] [17].

Key Scheduling Algorithm (KSA) NGG yaitu :

Input :

a. Secret key array $K[0…N - 1]$.
b. Precomputed random array $a[0…N - 1]$.

Output : Scrambled array $S[0…N - 1]$.

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 303−311
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1639

Initialization:
for $i = 0,\ldots, N - 1$ do
    $S[i] = a_i$;
    $j = 0$;
end
Scrambling :
for $i = 0,\ldots, N - 1$ do
    $j = (j + S[i] + K[i]) \bmod N$;
    Swap($S[i]$, $S[j]$);
    $S[i] = (S[i] + S[j]) \bmod M$;
end

Pada fase *Pseudo Random Generation Algorithm* (PRGA), elemen *pseudo-random* dikirim ke output dan setelah itu segera diubah oleh tambahan (mod M) dari dua elemen lain dari array S.
Pseudo Random Generation Algorithm (PRGA) NGG yaitu :
Input: Key-dependent scrambled array $S[0\ldots N - 1]$.
Output: Pseudo-random keystream bytes z.
Initialization:
$i = j = 0$;
Output Keystream Generation Loop:
$i = (i + 1) \bmod N$;
$j = (j + S[i]) \bmod N$;
Swap($S[i]$, $S[j]$);
Output $z = S\,[(S[i] + S[j]) \bmod M\,) \bmod N]$;
$S\,[(S[i] + S[j]) \bmod M\,) \bmod N] = (S[i] + S[j]) \bmod M$;

GGHN *cipher* merupakan versi lain dari NGG yang diperkenalkan. Dalam algoritma GGHN terdapat tiga variabel yaitu i, j, dan k untuk meningkatkan keamanan *cipher*. k merupakan inisialisasi dari KSA dan bergantung pada kunci. Jumlah perulangan loop KSA r bergantung pada parameter n, m.
Key Scheduling Algorithm (KSA) GGHN yaitu :
Input :
1. Secret key array $K[0\ldots N - 1]$.
2. Precomputed random array $a[0\ldots N - 1]$.
Output :
1. Scrambled array $S[0\ldots N - 1]$.
2. Key-dependent secret variable k.

Initialization:
for $i = 0,\ldots, N - 1$ do
    $S[i] = a_i$;
    $j = k = 0$;
end

Scrambling :
for $i = 0,\ldots, N - 1$ do
    $j = (j + S[i] + K[i \bmod l]) \bmod N$;
    Swap($S[i]$, $S[j]$);
    $S[i] = (S[i] + S[j]) \bmod M$;
    $k = (k + S[i]) \bmod M$;
end

Dalam *Pseudo Random Generation Algorithm* (PRGA), k digunakan untuk memperbarui S serta untuk menyamarkan output. *Pseudo Random Generation Algorithm* (PRGA) GGHN yaitu :
Input: Key-dependent scrambled array $S[0\ldots N - 1]$.
Output: Pseudo-random keystream bytes z.
Initialization:
$i = j = 0$;
Output Keystream Generation Loop:
$i = (i + 1) \bmod N$;
$j = (j + S[i]) \bmod N$;
$k = (k + S[j]) \bmod M$;
Output $z = S\,[(S[i] + S[j]) \bmod N + k) \bmod M]$;
$S\,[(S[i] + S[j]) \bmod N] = (k + S[i]) \bmod M$;

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 303−311
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1639

3. HASIL DAN PEMBAHASAN

Pada bab ini akan diuraikan hasil dan pembahasan yang dilakukan meliputi penerapan algoritma NGG dan algoritma GGHN dan hasil pengujian berdasarkan analisis frekuensi masing-masing algoritma. Data pada penelitian ini adalah pesan teks sejumlah 3000 karakter yang akan dienkripsi dengan menggunakan algoritma NGG dan GGHN. Analisis frekuensi dilakukan dengan menghitung kemunculan karakter pada teks pesan yang telah dienkripsi. Kunci yang digunakan yaitu teks *random string* seperti pada Tabel 1.

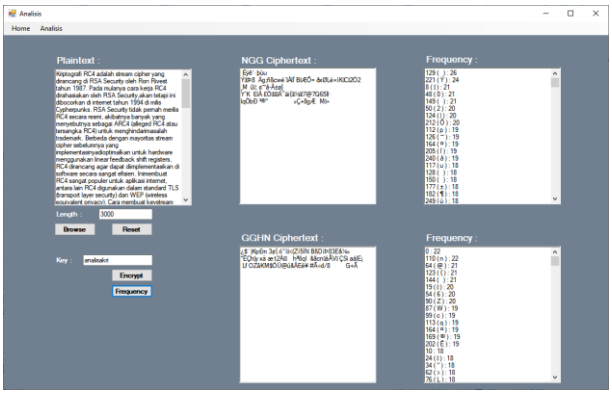
Tabel 1. Tabel Data Kunci

No	Percobaan Ke-	Panjang Kunci	Isi Kunci
1.	1	10 karakter	analisakri
2.	2	20 karakter	analisakriptografial
3.	3	30 karakter	analisakriptografialgortimangg
4.	4	40 karakter	analisakriptografialgoritmanggdangghndal
5.	5	50 karakter	analisakriptografialgoritmanggdangghndalamkeamanan
6.	6	60 karakter	analisakriptografialgoritmanggdangghndalamkeamananpesanteksd
7.	7	70 karakter	analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarak
8.	8	80 karakter	analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarakterpadakun
9.	9	90 karakter	analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarakterpadakuncipesantek
10.	10	100 karakter	analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarakterpadakuncipesanteks1karakter

3.1 Penerapan Algoritma NGG dan GGHN

Pada proses penerapan algoritma NGG dan algoritma GGHN dilakukan 10 percobaan terhadap 3000 karakter pesan teks untuk melihat hasil enkripsi dan jumlah kemunculan tiap karakter pada pesan teks dengan menggunakan panjang karakter kunci yang berbeda-beda. Kode karakter yang digunakan yaitu kode ASCII 8 bit. Informasi yang dapat diambil yaitu, jumlah karakter pesan teks yang digunakan, jumlah kemunculan masing-masing karakter pada pesan teks dan presentasi kemunculan masing-masing karakter.

Pada percobaan ke-1 dapat dilihat pada Gambar 5, dilakukan penerapan algortima NGG dan GGHN dengan menggunakan 10 karakter kunci yaitu “analisakri”. Dapat dilihat hasil enkripsi dan analisis frekuensi pada algoritma NGG yaitu karakter “ ” kode ASCII 129 sebagai karakter yang kemunculannya paling banyak 26 kali dengan presentasi kemunculan sebesar 0.86666667%. Pada algoritma GGHN yaitu karakter “NULL” kode ASCII 0 sebagai karakter yang kemunculannya paling banyak 22 kali dengan presentasi kemunculan sebesar 0.7333% pada pesan teks.



Gambar 2. Hasil Enkripsi Percobaan ke-1

Pada percobaan ke-2, dilakukan penerapan algortima NGG dan GGHN dengan menggunakan 20 karakter kunci yaitu “analisakriptografial”. Hasil enkripsi dan analisis frekuensi pada algoritma NGG yaitu karakter “ACK” kode ASCII 6 sebagai karakter yang kemunculannya paling banyak 22 kali dengan presentasi kemunculan sebesar 0.7333%. Sedangkan pada algoritma GGHN yaitu karakter “q” kode ASCII 113 sebagai karakter yang kemunculannya paling banyak 23 kali dengan presentasi kemunculan sebesar 0.76666667%.

Pada percobaan ke-3, dilakukan penerapan algortima NGG dan GGHN dengan menggunakan 30 karakter kunci yaitu “analisakriptografialgortimangg”. Hasil enkripsi dan analisis frekuensi pada algoritma NGG yaitu karakter “Æ” kode ASCII 198 sebagai karakter yang kemunculannya paling banyak 25 kali dengan presentasi kemunculan sebesar 0.8333%. Sedangkan pada algoritma GGHN yaitu karakter “EM” kode ASCII 25 sebagai karakter yang kemunculannya paling banyak 22 kali dengan presentasi kemunculan sebesar 0.7333%.

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 303−311
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1639

Pada percobaan ke-4, dilakukan penerapan algortima NGG dan GGHN dengan menggunakan 40 karakter kunci yaitu "analisakriptografialgoritmanggdangghndal". Hasil enkripsi dan analisis frekuensi pada algoritma NGG yaitu karakter "BS" kode ASCII 8 sebagai karakter yang kemunculannya paling banyak 26 kali dengan presentasi kemunculan sebesar 0.86666667%. Sedangkan pada algoritma GGHN yaitu karakter "#" kode ASCII 35 sebagai karakter yang kemunculannya paling banyak 24 kali dengan presentasi kemunculan sebesar 0.8%.

Pada percobaan ke-5, dilakukan penerapan algortima NGG dan GGHN dengan menggunakan 50 karakter kunci yaitu "analisakriptografialgoritmanggdangghndalamkeamanan". Hasil enkripsi dan analisis frekuensi pada algoritma NGG yaitu karakter ")" kode ASCII 41 sebagai karakter yang kemunculannya paling banyak 20 kali dengan presentasi kemunculan sebesar 0.66666667%. Sedangkan pada algoritma GGHN yaitu karakter """ kode ASCII 132 sebagai karakter yang kemunculannya paling banyak 22 kali dengan presentasi kemunculan sebesar 0.7333%.

Pada percobaan ke-6, dilakukan penerapan algortima NGG dan GGHN dengan menggunakan 60 karakter kunci yaitu "analisakriptografialgoritmanggdangghndalamkeamananpesanteksd". Hasil enkripsi dan analisis frekuensi pada algoritma NGG yaitu karakter "L" kode ASCII 76 sebagai karakter yang kemunculannya paling banyak 22 kali dengan presentasi kemunculan sebesar 0.7333%. Sedangkan pada algoritma GGHN yaitu karakter "a" kode ASCII 97 sebagai karakter yang kemunculannya paling banyak 28 kali dengan presentasi kemunculan sebesar 0.9333%.

Pada percobaan ke-7, dilakukan penerapan algortima NGG dan GGHN dengan menggunakan 70 karakter kunci yaitu "analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarak". Hasil enkripsi dan analisis frekuensi pada algoritma NGG yaitu karakter "CAN" kode ASCII 24 sebagai karakter yang kemunculannya paling banyak 22 kali dengan presentasi kemunculan sebesar 0.7333%. Sedangkan pada algoritma GGHN yaitu karakter "¿" kode ASCII 191 sebagai karakter yang kemunculannya paling banyak 23 kali dengan presentasi kemunculan sebesar 0.76666667%.

Pada percobaan ke-8, dilakukan penerapan algortima NGG dan GGHN dengan menggunakan 80 karakter kunci yaitu "analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarakterpadakun". Hasil enkripsi dan analisis frekuensi pada algoritma NGG yaitu karakter "ä" kode ASCII 228 sebagai karakter yang kemunculannya paling banyak 23 kali dengan presentasi kemunculan sebesar 0.76666667%. Sedangkan pada algoritma GGHN yaitu karakter "r" kode ASCII 114 sebagai karakter yang kemunculannya paling banyak 22 kali dengan presentasi kemunculan sebesar 0.7333%.

Pada percobaan ke-9, dilakukan penerapan algortima NGG dan GGHN dengan menggunakan 90 karakter kunci yaitu "analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarakterpadakuncipesantek". Hasil enkripsi dan analisis frekuensi pada algoritma NGG yaitu karakter "Ã" kode ASCII 195 sebagai karakter yang kemunculannya paling banyak 23 kali dengan presentasi kemunculan sebesar 0.76666667%. Sedangkan pada algoritma GGHN yaitu karakter "M" kode ASCII 77 sebagai karakter yang kemunculannya paling banyak 29 kali dengan presentasi kemunculan sebesar 0.96666667%.

Pada percobaan ke-10 atau percobaan terakhir, dilakukan penerapan algortima NGG dan GGHN dengan menggunakan 100 karakter kunci yaitu "analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankar akterpadakuncipesanteks1karakter". Hasil enkripsi dan analisis frekuensi pada algoritma NGG yaitu karakter "GS" kode ASCII 29 sebagai karakter yang kemunculannya paling banyak 24 kali dengan presentasi kemunculan sebesar 0.8%. Sedangkan pada algoritma GGHN yaitu karakter "i" kode ASCII 105 sebagai karakter yang kemunculannya paling banyak 27 kali dengan presentasi kemunculan sebesar 0.9%.

Hasil dari 10 percobaan dengan menggunakan 10 panjang kunci yang berbeda-beda memiliki hasil frekuensi yang bervariasi. Rincian frekuensi tertinggi pada setiap percobaan dapat dilihat pada Tabel 2. Semakin banyak jumlah karakter yang digunakan pada kunci maka akan mempengaruhi frekuensi kemunculan karakter pada proses enkripsi.

Tabel 2. Tabel Frekuensi

No	Percobaan Ke-	Panjang Kunci	Frekuensi Tertinggi NGG	Frekuensi Tertinggi GGHN
1.	1	10 karakter	26 kali	22 kali
2.	2	20 karakter	22 kali	23 kali
3.	3	30 karakter	25 kali	22 kali
4.	4	40 karakter	26 kali	24 kali
5.	5	50 karakter	20 kali	22 kali
6.	6	60 karakter	22 kali	28 kali
7.	7	70 karakter	22 kali	23 kali
8.	8	80 karakter	23 kali	22 kali
9.	9	90 karakter	23 kali	29 kali
10.	10	100 karakter	24 kali	27 kali

3.2 Hasil Pengujian

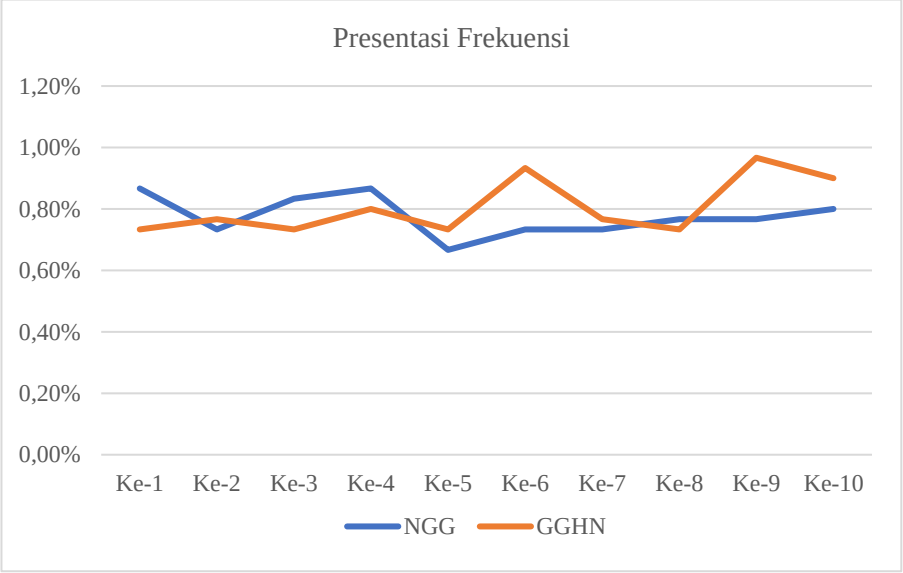
Pada bagian ini akan dilakukan pengujian terhadap analisis frekuensi dari hasil enkripsi pada algoritma NGG dan GGHN untuk mengetahui tingkat keamanan informasi. Proses pengujian kedua algoritma menggunakan pesan teks sejumlah 3000 karakter yang diinput ke sistem dan akan dienkripsi menggunakan masing-masing algoritma yaitu

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 303−311
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1639

algoritma NGG dan algortima GGHN. Selanjutnya berdasarkan hasil enkripsi akan dianalisis frekuensi karakter yang muncul memiliki presenasi yang dapat lihat pada tabel 3.

Tabel 3. Tabel Presentasi Frekuensi

No	Percobaan Ke-	Panjang Kunci	Isi Kunci	Presentasi Kemuncualan Tertinggi NGG	Presentasi Kemuncualan Tertinggi GGHN
1.	1	10 karakter	analisakri	0.86666667%	0.7333%
2.	2	20 karakter	analisakriptografial	0.7333%	0.76666667%
3.	3	30 karakter	analisakriptografialgortimangg	0.8333%	0.7333%
4.	4	40 karakter	analisakriptografialgoritmanggdangghndal	0.86666667%	0.8%
5.	5	50 karakter	analisakriptografialgoritmanggdangghndalamkeamanan	0.66666667%	0.7333%
6.	6	60 karakter	analisakriptografialgoritmanggdangghndalamkeamananpesanteksd	0.7333%	0.9333%
7.	7	70 karakter	analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarak	0.7333%	0.76666667%
8.	8	80 karakter	analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarakterpadakun	0.76666667%	0.7333%
9.	9	90 karakter	analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarakterpadakuncipesantek	0.76666667%	0.96666667%
10.	10	100 karakter	analisakriptografialgoritmanggdangghndalamkeamananpesanteksdengankarakterpadakuncipesanteks1karakter	0.8%	0.9%



Gambar 3. Grafik Hasil Presentasi Frekuensi

Berdasarkan dari Tabel 3 dan Gambar 3 dapat dilihat pada grafik hasil presentasi frekuensi kemunculan karakter terhadap panjang kunci yang berbeda-beda pada algoritma NGG dan GGHN memiliki hasil yang berbeda beda pada setiap percobaan. Pada 10 percobaan analisis frekuensi algoritma NGG didapat rata-rata presentasi kemunculan sebesar 0.007766533 % dan algoritma GGHN didapat rata-rata presentasi kemunculan sebesar 0.0080665 %.

Building of Informatics, Technology and Science (BITS)
Volume 4, No 1, Juni 2022 Page: 303−311
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v4i1.1639

4. KESIMPULAN

Pada penelitian ini, analisis frekuensi terhadap algoritma NGG dan GGHN untuk mengetahui tingkat keamanan informasi berdasarkan hasil enkripsi data. Dalam proses enkripsi pesan tergantung pada kunci yang telah ditentukan. Hal tersebut memiliki pengaruh yang signifikan terhadap hasil enkripsi. Berdasarkan proses pengujian pada algoritma NGG dan GGHN menghasilkan frekuensi karakter pada teks yang bervariasi. Pada 10 percobaan didapat rata-rata presentasi kemunculan tertinggi pada algoritma NGG yaitu 0.007766533% dan rata-rata presentasi kemunculan tertinggi pada algoritma GGHN 0.0080665%. Berdasarkan pada penelitian ini dapat diketahui bahwa algoritma NGG memiliki tingkat keamanan yang lebih tinggi dibandingkan algoritma GGHN dengan selisih presentasi sebesar 0.000299967%. Jika semakin banyak jumlah karakter yang digunakan pada kunci maka akan mempengaruhi tingkat keamanan informasi. Diharapkan dapat dilakukan pengujian terhadap lebih banyak pesan dan kunci agar menghasilkan akurasi dalam proses analisis frekuensi yang lebih spesifik. Jika frekuensi kemunculan karakter pada teks pesan yang telah dienkripsi semakin sering atau semakin tinggi, maka tingkat keamanan informasi pada pesan lebih rendah dan kata kunci lebih mudah dipecahkan.

REFERENCES

[1] A. Khalid, G. Paul, and A. Chattopadhyay, *Domain specific high-level synthesis for cryptographic workloads*.
[2] S. D. Galbraith, *Mathematics of Public Key Cryptography*, 1st ed. United State of America: Cambridge University Press, 2012.
[3] D. Chaudhuri, G. Fink, and K. Liu, *Discrete subaortic stenosis in a patient with a history of repaired AV canal defect*, vol. 28, no. 9. 2011. doi: 10.1111/j.1540-8175.2011.01507.x.
[4] S. Banik and T. Isobe, "Cryptanalysis of the full Spritz stream cipher," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vol. 9783, pp. 63–77. doi: 10.1007/978-3-662-52993-5_4.
[5] D. Liestyowati, *Public Key Cryptography*, vol. 1477, no. 5. 2020. doi: 10.1088/1742-6596/1477/5/052062.
[6] G. Paul, S. Maitra, and A. Chattopadhyay, "Quad-RC4: Merging Four RC4 States towards a 32-bit Stream Cipher.," *IACR Cryptol. ePrint Arch.*, vol. 2013, no. May, p. 572, 2013, [Online]. Available: http://dblp.uni-trier.de/db/journals/iacr/iacr2013.html#PaulMC13
[7] A. Kircanski and A. M. Youssef, "On the structural weakness of the GGHN stream cipher," *Cryptogr. Commun.*, vol. 2, no. 1, pp. 1–17, Sep. 2010, doi: 10.1007/s12095-009-0013-3.
[8] A. Kircanski and A. M. Youssef, "On the weak state in GGHN-like ciphers," in *Proceedings - 2012 7th International Conference on Availability, Reliability and Security, ARES 2012*, 2012, pp. 397–401. doi: 10.1109/ARES.2012.32.
[9] I. Print, S. An-nissa, H. Mawengkang, and S. Efendi, "InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan RC4 GGHN Cryptography Algorithm for Message Security," vol. 2, pp. 6–10, 2022.
[10] F. Akbar, H. Mawengkang, and S. Efendi, "Comparative analysis of RC4+ algorithm, RC4 NGG algorithm and RC4 GGHN algorithm on image file security," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 420, no. 1, 2018, doi: 10.1088/1757-899X/420/1/012131.
[11] S. Suhandinata, R. A. Rizal, D. O. Wijaya, P. Warren, and S. Srinjiwi, "Analisis Performa Kriptografi Hybrid Algoritma Blowfish Dan Algoritma Rsa," *JURTEKSI (Jurnal Teknol. dan Sist. Informasi)*, vol. 6, no. 1, pp. 1–10, 2019, doi: 10.33330/jurteksi.v6i1.395.
[12] G. Paul and S. Maitra, *RC4 stream cipher and its variants*. CRC Press, 2012.
[13] S. Banik, S. Maitra, and S. Sarkar, "On the evolution of GGHN cipher," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7107 LNCS, pp. 181–195, 2011, doi: 10.1007/978-3-642-25578-6_15.
[14] A. Kircanski, R. Al-Zaidy, and A. M. Youssef, "A new distinguishing and key recovery attack on NGG stream cipher," *Cryptogr. Commun.*, vol. 1, no. 2, pp. 269–282, May 2009, doi: 10.1007/s12095-009-0012-4.
[15] S. Banik, S. Sarkar, and R. Kacker, "Security analysis of the RC4+ stream cipher," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8250 LNCS, pp. 297–307, 2013, doi: 10.1007/978-3-319-03515-4_20.
[16] F. Riza, N. Sridewi, A. M. Husein, and M. K. Harahap, "Analisa Frekuensi Hasil Enkripsi Pada Algoritma Kriptografi Blowfish Terhadap Keamanan Informasi," *J. Teknol. dan Ilmu Komput. Prima*, vol. 1, no. 1, pp. 11–15, 2018, doi: 10.34012/jutikomp.v1i1.233.
[17] P. Singh, D. Mishra, and K. Seth, *Transformation in healthcare with emerging technologies*.