Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

Penerapan Algoritma Gronsfeld Dengan Pembangkit Kunci Quicksort Untuk Mengamankan Pesan Rahasia

Ria Ramadhani Syahputri, Mesran*, Murdani

Fakultas Ilmu Komputer dan Teknologi Informasi, Prodi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia
Jl.Sisingamangaraja No. 338, Siti Rejo I, Kec. Medan Kota, 20219, Kota Medan, Indonesia
Email: [1]riaramadhanisyahputri@gmail.com, [2,*]mesran.skom.mkom@gmail.com, [3]murdanimkom@gmail.com
Correspondence Author Email: mesran.skom.mkom@gmail.com
Submitted: 15/08/2023; Accepted: 31/08/2023; Published: 31/08/2023

Abstrak−Banyaknya masalah yang sering terjadi dalam mengamankan pesan rahasia adalah bocornya pesan. Bocornya pesan terjadi dalam beberapa cara, seperti penyadapan. Penyadapan terjadi ketika pihak yang tidak berwenang mampu mengakses dan memperoleh informasi dari komunikasi yang sedang berlangsung. Salah satu cara untuk menjaga kerahasiaan pesan rahasia ialah dengan enkripsi dan deskripsi yang dikenal dengan nama kritografi, Dengan menyandingkan pesan asli (plaintext) ke dalam bentuk pesan rahasia (chipertext). Dalam penelitian ini, hasil dari chipertext karakter enkripsi adalah v, v, w, ^, ™, ˇ , ¦ , ¥ , - , ¼ , Ý , ó , ⁒ , x , [] ," , [], # ,S , a , I, p, p hasil ini dapat dijadikan sebagai alternatif solusi dalam menjaga kerahasiaan pesan sehingga hanya dapat diakses oleh pemilik pesan dan seseorang yang mengetahui kuncinya. Metode kriptografi Gronsfeld juga dapat meningkatkan keamanan dari algoritma Quicksort yang merupakan algoritma kriptografi karena akan menghasilkan pesan teks yang lebih acak serta tidak memperlihatkan pola-pola keterhubungannya dengan pesan teks asli.

Kata Kunci: *Kriptografi; Gronsfeld Cipher, Quicksort;* Pesan Rahasia; Enkripsi

Abstract−The numerous issues often encountered in securing confidential messages revolve around message leaks. Message leaks occur through various means, such as eavesdropping. Eavesdropping happens when unauthorized parties manage to access and obtain information from ongoing communications. One way to maintain the confidentiality of secret messages is through encryption and decryption, known as cryptography, achieved by transforming the original message (plaintext) into a secret message (ciphertext). In this study, the results of the encrypted character ciphertext are v, v, w, ^, ™, ˇ, ¦, ¥, -, ¼, Ý, ó, ⁒, x, [], '', [], #, S, a, I, p, p. These findings can serve as an alternative solution to safeguard message confidentiality, ensuring only the message owner and those with knowledge of the key can access it. The Gronsfeld Cipher cryptographic method can also enhance the security of the Quicksort algorithm, a cryptographic algorithm, by generating a more randomized text message that does not reveal patterns connecting it to the original text message.

Keywords: Cryptography; Gronsfeld Cipher; Quicksort; Secret Message; Encryption

1. PENDAHULUAN

Pesan rahasia merupakan sebuah pesan atau komunikasi yang dirancang untuk tetap terjaga kerahasiaannya antara pengirim dan penerima. Tujuan utama dari pesan rahasia merupakan langkah-langkah untuk melindungi data sensitif agar tidak dapat diakses oleh entitas yang tidak memiliki otorisasi atau yang tidak diinginkan[1]. Hal ini menyebabkan ketakutan bagi pengirim pesan. Pesan yang dicuri akan dimanfaatkan secara tidak sah untuk mendapatkan keuntungan dari pemilik pesan tersebut. Mengamankan pesan memerlukan metode yang memastikan perlindungan terhadap pesan yang dikirim melalui jaringan komputer dari potensi serangan oleh pihak yang tidak bertanggung jawab.

Sekarang ini, banyaknya masalah yang sering terjadi dalam mengamankan pesan rahasia adalah bocornya pesan. Bocornya pesan terjadi dalam beberapa cara, seperti penyadapan. Penyadapan terjadi ketika pihak yang tidak berwenang mampu mengakses dan memperoleh informasi dari komunikasi yang sedang berlangsung. Penyadap dapat mencuri atau memperoleh pesan rahasia saat dikirimkan melalui jalur komunikasi yang tidak terlindungi., seperti jaringan wi-fi publik atau kabel yang tidak terlindungi. Salah satu metode untuk mengamankan kerahasiaan pesan yang bersifat rahasia adalah melalui enkripsi dan dekripsi yang dikenal dengan istilah kriptografi. Dengan mengubah pesan asli (plaintext) menjadi bentuk pesan rahasia (chipertext), proses ini melibatkan penggunaan algoritma dan kunci keamanan. Salah satu teknik kriptografi yang relevan untuk situasi ini adalah menggunakan algoritma Quicksort dengan algoritma Gronsfeld[2].

Cipher Gronsfeld dinamai sesuai dengan penciptanya, yaitu *Johann Franz Graf Gronsfeld-Bronkhorst,* yang juga dikenal sebagai seorang panglima kekaisaran selama pemberontakan nasional Bavaria pada tahun 1705-1706. Teknik ini memiliki kesamaan dengan metode *Vigenere*[3]. Gronsfeld cipher adalah sebuah metode kriptografi yang memanfaatkan kunci berupa angka dan tabel dalam tahap enkripsi dan dekripsi. Kunci yang dipakai akan berulang dari sisi kiri untuk memproses enkripsi atau dekripsi, bahkan saat panjang pesan yang akan dienkripsi atau didekripsi lebih besar daripada panjang kunci yang digunakan[4]. Gronsfeld ciper merupakan sebuah algoritma kriptografi yang menggunakan suatu kunci numerik dan table dalam proses enkripsi dan deskripsi, kunci yang digunkan akan diulang dari kiri yang akan di enkripsi atau deskripsi lebih panjang dari kunci yang digunakan[5]. Untuk meningkatkan efektivitas algoritma tersebut, langkah pertama adalah melakukan modifikasi pada kunci yang akan digunakan dalam proses enkripsi dan dekripsi melalui penerapan algoritma pengacakan. [6]. Oleh karena itu, setiap individu yang memiliki akses kepada kunci publik

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

dapat menjalankan langkah enkripsi, tetapi hasil enkripsi tersebut hanya dapat terbaca oleh mereka yang memiliki akses kunci pribadi. Dalam algoritma gronsfeld, pendekatan yang digunakan adalah kunci simetris, yang menyebabkan kunci yang sama digunakan baik pada tahap enkripsi maupun dekripsi. Algoritma Quicksort merupakan metode untuk mengurutkan data secara rekursif. Proses pengurutan dilakukkan dengan membagi dataset menjadi dua bagian berdasarkan nilai pivot yang telah ditentukan [7].

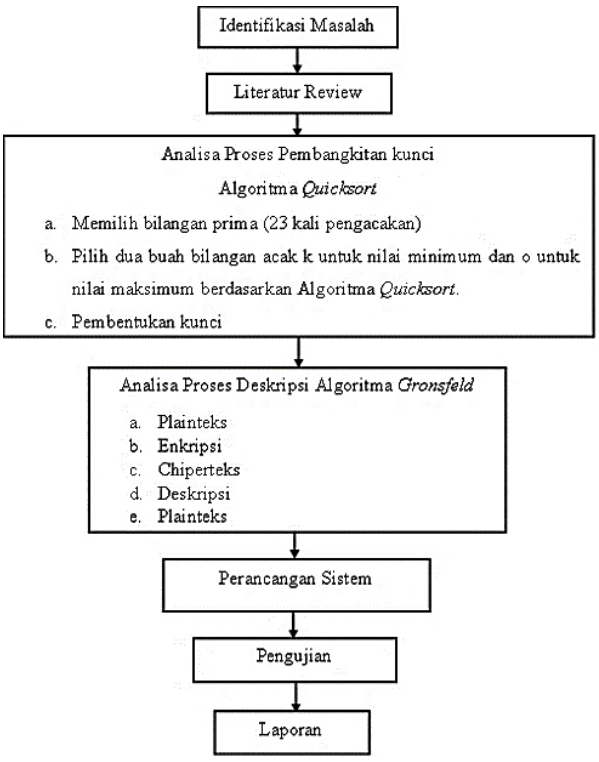
Proses operasi Algoritma Quicksort dilakukan melalui perbandingan antara suatu elemen (yang disebut pivot) dengan elemen-elemen lainnya, dan mengatur posisi elemen-elemen tersebut sedemikian rupa sehingga elemen-elemen yang memiliki nilai lebih kecil daripada pivot diletakkan pada sisi kiri, sementara elemen-elemen yang memiliki nilai lebih besar daripada pivot ditempatkan pada sisi kanan. [8]. Penelitian terkait yang relevan dengan judul yang penulis ambil ini dijadikan sebagai pendukung. Sehingga, penelitian memerlukan reverensi terhadap hasil yang diteliti oleh seseorang terlebih dahulu berupa karya tulis ilmiah yang bersumber dari *internet*. Seperti yang dilakukan oleh A.miftahul,I Al Amin Husni pada tahun 2019 [9], output yang dihasilkan dari penelitian ini adalah pemeriksaan pasien menggunakan metode *quicksort* jauh lebih efektif sehingga dapat meningkatkan pelayanan klik tersebut. Penelitian yang dilakukan oleh Indiyani Angelina,2022[10], dalam pembahasannya mengenai algoritma *Gronsfeld* dengan mengkompresi menggunakan algoritma *Golbach Code* hasil *size* lebih kecil dari teks aslinya sehingga penyimpanan jadi lebih efektif sehingga metode ini cocok untuk penelitian ini. Hasil penelitian yang dirancang oleh Nuri David Maria Veronika dan Yulia Darnita dengan menggunakan algoritma quicksort untuk merancang aplikasi toelf dapat disimpulkan bahwa dapat mengetahui hasil nilai tes dengan mudah dan dapat menentukan perangkingan dari aplikasi tersebut tanpa harus mengitung manual persentase nilainnya [8]. Sedangkan penelitian yang dilakukan oleh Panny Agustia Rahayuningsih,2016 dengan judul perbandingan algoritma pengurutan nilai *(sorting),* hasilnnya adalah nilai dengan algoritma *Quicksort* lebih cepat apabila dibandingkan dengan metode pengurutan shell sort, insertion sort, selection sort, dan bubble sort [11]. Pada penelitiann A. L. Noviani and I.Yuliani ,2019 [5], menggunakan teknik kriptografi memang masih layak untuk penyandikan mengamankan pesan sehingga pesan tetap aman dari pihak lain.

Penelitian ini bertujuan untuk mengamankan pesan menggunakan algoritma cipher *Gronsfeld*. Berdasarkan latar belakang yang dijelaskan, penulis mengambil penelitian dengan judul "Penerapan Algoritma *Gronsfeld* Dengan Pembangkit Kunci *Quicksort* Untuk Mengamankan Pesan Rahasia".

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Tahapan Penelitian merupakan tahapan yang dilakukan untuk memperoleh hasil optimal, sehingga langkah-langkah yang diambil dalam penelitian ditentukan. Adapun kerangka tahapan dalam proses penyelesaian penelitian ini seperti gambar 1 anatar lain:



Gambar 1. Kerangka Penelitian

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

Adapun penjelasan dari gambar kerangka penelitian diatas yang dilakukan oleh penulis antara lain:

1. Identifikasi Masalah
   Pada langkah ini, pengarang menjelaskan asal mula permasalahan yang timbul pada penyimpanan pesan rahasia yang menyebabkan terjadinya pembobolan pesan rahasia diketahui orang lain.
2. Literatur Review
   Literature review adalah proses sistematik dalam studi ini melibatkan proses mengumpulkan, memilih, dan secara kritis mengevaluasi sumber-sumber literatur yang berkaitan dengan subjek penelitian yang tengah diselidiki.
3. Tahap Analisa Proses Pembangkitan Kunci Menerapkan Algoritma Quicksort
   Pada tahap ini dilakukan proses pembangkitan kunci bilangan acak dengan menerapkan algoritma Quicksort yang memiliki tujuan untuk menentukan kunci dari algoritma gronsfeld.
4. Tahap Analisa Proses Dekripsi Algoritma Gronsfeld
   Pada tahap ini melakukan proses pengembalian pesan teks yang telah dienkripsi dengan algoritma Quicksort.
5. Tahapan Rancangan Sistem
   Pada tahap ini, diberikan gambaran mengenai aplikasi yang telah dirancang oleh penulis dengan tampilan yang simpel dan mudah dimengerti oleh pengguna.
6. Tahapan Pengujian
   Proses ini dilaksanakan untuk menguji hasil dari pesan tersembunyi. Langkah ini bertujuan untuk mendapatkan pemahaman tentang hasil akhir dari studi yang telah dijalankan.
7. Tahap Laporan
   Tahap ini mencakup langkah terakhir dari studi yang dilakukan oleh penulis. Pada tahap ini, akan dijelaskan mengenai aplikasi yang telah dihasilkan, dengan tujuan memberikan panduan yang lebih komprehensif bagi pembaca yang berkeinginan untuk melanjutkan pengembangan aplikasi tersebut.

2.2 Kriptografi

Kriptografi adalah informasi yang ingin Anda kirim harus bersifat rahasia di dalamnya dan tetap asli di tempat tujuan tanpa perlu perubahan[12]. Kriptografi merupakan bidang ilmu yang digunakan untuk mengamankan data melalui proses penyandian[13][14]. Kriptografi merupakan bidang pengetahuan yang memfokuskan pada metode-metode matematika yang terkait dengan perlindungan informasi, seperti keamanan, keutuhan data, autentikasi pengirim/penerima data, dan verifikasi data[15]. Kriptografi juga dikenal sebagai kombinasi ilmu dan seni dalam menjaga kerahasiaan pesan dengan mengubahnya menjadi bentuk yang tidak lagi dapat dipahami maknanya[16].

2.3 Algoritma *Quicksort*

Prinsip yang digunakan oleh algoritma quicksort adalah membagi dan mengatasi *(divide-and-conquer*)[17]. Oleh karena itu, inti utama dari algoritma quicksort adalah pemilihan titik pivot dan penyusunan partisi dengan cara yang sesuai. [18]rupa sehingga elemen dengan nilai kecil ada dibagian kiri poros dan elemen dengan nilai besar ada di bagian kanan poros [5]. Pada tahun 1998,*M.D.Mcllroy* menghasilkan sebuah artikel berjudul *"A Killer Adversary for Quicksort"*[5] yang dijelaskan adalah cara menciptakan pengaturan data khusus (dalam bentuk array) sedemikian rupa sehingga penggunaan algoritma quicksort dalam operasi pengurutan mendekati kompleksitas kuadratik O(n^2)[16]. Metode ini berlaku untuk semua variasi dari quicksort dengan ketentuan spesifik[19].

Waktu yang dibutuhkan oleh quicksort bergantung pada bagaimana partisi dibentuk, apakah seimbang atau tidak, yang juga dipengaruhi oleh elemen yang dijadikan poros[20]. Dalam menghitung kompleksitas ini, penting juga untuk mempertimbangkan perhitungan rekursif, karena terdapat fungsi rekursif untuk memecahkan sub-masalah.

Ada tiga jenis kompleksitas waktu yang terkait dengan quicksort:

1. Kasus terburuk (worst case), terjadi saat partisi terbentuk dengan pembagian sub-masalah antara n – 1 elemen dan 0 elemen. Ini mengakibatkan pemanggilan fungsi rekursif dengan array berukuran 0 yang akan segera kembali, T (0) = O (1), maka berlaku: T(n) = T(n-1) + cn = O (n^2).
2. Kasus terbaik (best case), terjadi saat partisi terbentuk secara seimbang, dengan ukuran masing-masing tidak lebih dari n/2. Sehingga diperoleh: T(n) = 2T(n/2) + cn = na + cn log n = O (n log n).
3. Kasus rata-rata (average case), terjadi sebagai hasil dari keseimbangan antara kasus terbaik dan terburuk, yang dalam praktiknya lebih mendekati kasus terbaik daripada kasus terburuk. Sehingga diperoleh: T_avg(n) = O (n log n)

2.4 Pembentukan Kunci

Pembentukan kunci adalah tahap krusial dalam kriptografi dimana kunci yang akan digunakan untuk melakukan enkripsi atau dekripsi data diciptakan[21]. Kunci kriptografi merupakan rangkaian bit yang diterapkan oleh algoritma kriptografi untuk mengubah data menjadi bentuk yang tak terbaca (pada proses enkripsi) atau mengembalikan data ke bentuk asalnya (pada proses dekripsi)[4]. Kunci untuk enkripsi dihasilkan melalui nilai

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

p, g, dan y, sementara kunci untuk dekripsi terdiri dari nilai x dan p. Setiap nilai ini wajib mematuhi persyaratan tertentu. Berikut ini adalah langkah-langkah dalam proses pembuatan kunci:

1. Pilih sembarang bilangan prima K, dengan syarat K> 256.
2. Tentukan suatu bilangan acak g yang memenuhi kondisi g < p.
3. Pilih angka acak x sesuai dengan syarat 1 ≤ x ≤ p – 2.
4. Hitung nilai y = g^x mod p.

$$y = g^{\wedge} x \ mod \ p \quad (1)$$

Kunci public nya adalah y, g, p.

2.5 Proses Enkripsi

Berikut adalah langkah-langkah dalam proses enkripsi algoritma Quicksort:
Pilih sebuah karakter dari pesan yang perlu dienkripsi dan ubah karakter tersebut menjadi kode ASCII untuk mendapatkan bilangan bulat B.
Hitung nilai r dan t menggunakan persamaan berikut:

$$r = \alpha k \ (\text{mod } p) \quad (2)$$

$$t = \beta k \ B \ (\text{mod } p) \quad (3)$$

Hasilnya adalah cipherteks untuk karakter B dalam blok (r, t).
Lakukan langkah di atas untuk semua karakter dalam pesan, termasuk karakter spasi.

2.6 Proses deskripsi

Proses untuk mengubah cipherteks kembali menjadi plainteks menggunakan kunci rahasia a, yang dijaga kerahasiaannya oleh penerima pesan. Kunci publik dinyatakan sebagai (p, α, β), sementara kunci rahasia adalah a dalam algoritma Quicksort. Jika diberikan cipherteks dalam bentuk (r, t), maka

$$M = t\,(r^a)^{-1} \text{ mod } p \quad (4)$$

Jika M mewakili plainteks, di mana nilai (ra)-1 = r-a = rp-1-a mod p.
Berikut adalah langkah-langkah dalam proses dekripsi algoritma quicksort [11]:

1. Ambil sebuah blok cipherteks dari pesan yang telah dienkripsi oleh pengirim.
2. Dengan menggunakan nilai a yang dijaga kerahasiaannya oleh penerima, lakukan perhitungan untuk mendapatkan nilai plainteks menggunakan

Persamaan $M = t\,(r^a)^{-1} \text{ mod } p$ dan persamaan $(r^a)^{-1} = r^{-a} = r^{p-1-a} \text{ mod } p$ (5)

2.7 Algoritma Gronsfeld

Gronsfeld adalah sebuah metode pengurutan yang awalnya dikembangkan oleh C.A.R. Hoare pada tahun 1960. Algoritma ini memiliki kompleksitas pengurutan rata-rata O(n log n) untuk mengurutkan sejumlah n item. [22]. Gronsfeld juga dikenal sebagai algoritma kriptografi teks terang modern yang menggunakan kunci simetris dan beroperasi sebagai blok cipher [23]. Gronsfeld mendapat namanya dari Johann Franz Graf Gronsfeld-Bronkhorst, seorang panglima kekaisaran dalam pemberontakan nasional Bavaria pada tahun 1705-1706. Algoritma ini adalah metode enkripsi yang mengandalkan angka sebagai kunci pengkodean[12]. Kuncinya diatur oleh Programmer. Gronsfeld Cipher adalah metode kriptografi yang bekerja seperti Vigenere Cipher. Gronsfeld Cipher menggunakan kunci dari angka desimal bukan huruf, tetapi kadang-kadang dapat menggunakan ASCII sebagai substitusi kunci [3].

Langkah-langkah pengerjaannya terlihat seperti kode Vigenere, di mana ia akan diulang secara berkala untuk kunci untuk tujuan jadi panjang dan kunci plaintek sama. Ada dua model penggunaan karakter dalam algoritma *Gronsfeld Cipher*. Algoritma ini dapat menggunakan 256 ASCII karakter atau hanya menggunakan 26 karakter *alphabet*. Persamaan (4) adalah rumus yang digunakan untuk proses enkripsi dengan menggunakan algoritma *Gronsfeld Ciphe.*

$$C_i = (P_i + K_i) \text{mod } 256 \quad (6)$$

Sedangkan persamaan (2.4) adalah rumus yang digunakan untuk proses dekripsi dengan menggunakan algoritma *Gronsfeld Cipher*.

$$P_i = (C_i - K_i) \text{mod } 256 \quad (7)$$

Keterangan Penjelasan mengenai persamaan (6) dan persamaan (7) adalah sebagai berikut:
C_i = representasi desimal (dalam ASCII) dari karakter ke-i dalam Gronsfeld Cipher.
P_i = representasi desimal (dalam ASCII) dari karakter ke-i dalam Plaintext.
K_i = representasi desimal (dalam ASCII) dari karakter ke-i dalam kunci yang digunakan.

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

3. HASIL DAN PEMBAHASAN

3.1 Penerapan Algoritma Quicksort

Proses pembangkitan kunci dilakukan melalui penerapan algoritma yang memanfaatkan angka acak[24]. Dengan memasukkan angka-angka acak ini, dianggap mampu mengurangi peluang bagi pihak yang tidak sah untuk menebak hasilnya dengan memiliki pengetahuan tentang algoritma yang digunakan. Berikut adalah tahap-tahap dalam memodifikasi kunci algoritma Gronsfeld menggunakan algoritma Quicksort:

1. Tentukan sebuah bilangan prima secara acak melalui 23 iterasi pengacakan.
2. Pilih dua angka acak, misalnya k untuk nilai minimum dan o untuk nilai maksimum berdasarkan metode Quicksort.
3. Kunci dihasilkan dari bilangan prima K dan O. Diberikan informasi berikut:

Diketahui:

B=1 Kn=53 O=103 Mod=300

K_1 = (1*(53+103) Mod 300
= 156 Mod 300 = 156

K_2 = (1*(156+103) Mod 300
= 259 Mod 300 = 259

K_3 = (1*(259+103) Mod 300
= 362 Mod 300 = 62

K_4 = (1*(62+103) Mod 300
= 165 Mod 300 = 165

K_5 = (1*(165+103) Mod 300
= 268 Mod 300 = 268

K_6 = (1*(268+103) Mod 300
= 371 Mod 300 = 71

K_7 = (1*(71+103) Mod 300
= 174 Mod 300 = 174

K_8 = (1*(174+103) Mod 300
= 277 Mod 300 = 277

K_9 = (1*(277+103) Mod 300
= 380 Mod 300 = 80

K_{10} = (1*(80+103) Mod 300
= 183 Mod 300 = 183

K_{11} = (1*(183+103) Mod 300
= 286 Mod 300 = 286

K_{12} = (1*(386+103) Mod 300
= 389 Mod 300 = 89

K_{13} = (1*(89+103) Mod 300
= 192 Mod 300 = 192

K_{14} = (1*(192+103) Mod 300
= 295 Mod 300 = 295

K_{15} = (1*(295+103) Mod 300
= 398 Mod 300 = 98

K_{16} = (1*(98+103) Mod 300

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

= 201 Mod 300 = 201

K_{17} = (1*(201+103) Mod 300

= 304 Mod 300 = 4

K_{18} = (1*(4+103) Mod 300

= 107 Mod 300 = 107

K_{19} = (1*(107+103) Mod 300

= 210 Mod 300 = 210

K_{20} = (1*(210+103) Mod 300

= 313 Mod 300 = 13

K_{21} = (1*(13+103) Mod 300

= 116 Mod 300 = 116

K_{22} = (1*(116+103) Mod 300

= 219 Mod 300 = 219

K_{23} = (1*(219+103) Mod 300

= 322 Mod 300 = 22

Sehingga didapatkan nilai acak adalah 156, 259, 62, 165, 268, 71, 174, 277, 80, 183, 286, 89, 192, 295, 98, 201, 4, 107, 210, 13, 116, 219, 22. Ada beberapa cara untuk menentukan nilai bilangan acak, antara lain:

a. Langkah pertama yaitu menentukan pivotnya, penulis memilih angka dari sebelah kanan kolom

Tabel 1. Nilai bilangan acak

156	259	62	165	268	71	174	277	80	183	286	89	192	295	98	201	4	107	210	13	116	219	22

b. Setelah itu, lakukan pembagian menjadi dua bagian di sebelah kiri dan kanan dengan mengurutkan angka dari yang terkecil hingga yang terbesar.
Sehingga didapat nilai acak dari algoritma *Quicksort* adalah :4, 13, 22, 62, 71, 80, 89, 98, 107, 116, 156, 174, 183, 192, 201, 210, 219, 259, 268, 277, 286, 295.

3.2 Penerapan Algoritma Gronsfeld dalam Mengamankan Pesan Rahasia

Untuk menerapkan Dalam algoritma Gronsfeld Cipher, penulis perlu mengkonversikannya ke dalam bentuk bilangan ASCII.

Plaintext	R	I	A	< >	R	A	M	A	D	H	A	N	I	< >	S	Y	A	H	P	U	T	R	I
Desimal	8	7	6		8	6	7	6	6	7	6	7	7		8	8	6	7	8	8	8	8	7
	2	3	5	32	2	5	7	5	8	2	5	8	3	32	3	9	5	2	0	5	4	2	3

Berikut ini adalah persamaan yang mendasari Algoritma Gronsfeld Cipher:

Cn = (Pn+Ki) mod 256

Keterangan:
Cn = nilai desimal (ASCII) karakter cipher ke-n
Pn = nilai desimal (ASCII) karakter plainteks ke-n
Kn = nilai desimal (ASCII) karakter kunci ke-n
Dalam sebuah pesan rahasia terdapat isi sebagai berikut:
Plaintext: RIA RAMADHANI SYAHPUTRI

Tabel 2. Bilangan ASCII nilai desimal *plaintext*

Plaintext	R	I	A	< >	R	A	M	A	D	H	A	N	I	< >	S	Y	A	H	P	U	T	R	I
Desimal	8	7	6		8	6	7	6	6	7	6	7	7		8	8	6	7	8	8	8	8	7
	2	3	5	32	2	5	7	5	8	2	5	8	3	32	3	9	5	2	0	5	4	2	3

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

Tabel 3. Kunci bilangan random

									10	11	11	11	11	11	11	20	20	20	20	20	20	20	20
Kunci	4	13	22	62	71	80	89	98	07	16	56	65	74	83	92	01	10	19	59	68	77	86	95

3.3 Proses Enkripsi

Dalam proses Enkripsi, kunci yang digunakan ialah kunci publik yang sudah dirahasiakan oleh pengirim pesan agar tidak diketahui oleh banyak orang. Proses enkripsi melibatkan penggunaan bilangan bulat unik untuk setiap karakter dalam pesan, di mana setiap karakter direpresentasikan oleh nilai ASCII integer yang menghasilkan kode. Berikut adalah langkah-langkah dalam enkripsi plainteks menggunakan kunci bilangan acak:

R $C_1 = (82+4) \bmod 256$

$=86 \bmod 256$

$= 86$ (v)

I $C_2 = (73+13) \bmod 256$

$= 86 \bmod 256$

$= 86$ (v)

A $C_3 = (65+22) \bmod 256$

$=87 \bmod 256$

$= 87$ (w)

Spasi $C_1 = (32+62) \bmod 256$

$=94 \bmod 256$

$= 94$ (^)

R $C_5 = (82+71) \bmod 256$

$=153 \bmod 256$

$= 153$ (™)

A $C_6 = (65+80) \bmod 256$

$=145 \bmod 256$

$= 145$ (‘)

M $C_7 = (77+89) \bmod 256$

$= 166 \bmod 256$

$= 166$ (¦)

A $C_8 = (65+98) \bmod 256$

$= 163 \bmod 256$

$= 163$ (¥)

D $C_9 = (68+107) \bmod 256$

$= 175 \bmod 256$

$=175$ (-)

H $C_{10} = (72+116) \bmod 256$

$= 188 \bmod 256$

$= 188$ (¼)

A $C_{11} = (65+156) \bmod 256$

$= 221 \bmod 256$

$= 221$ (Ý)

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

N $C_{12} = (78+165) \bmod 256$
$= 243 \bmod 256$
$= 243$ (ó)

I $C_{13} = (73+174) \bmod 256$
$= 247 \bmod 256 = 247$ (⁒)

Spasi $C_{14} = (32+183) \bmod 256$
$= 215 \bmod 256$
$= 215$ (x)

S $C_{15} = (83+192) \bmod 256$
$= 275 \bmod 256$
$= 19$ ([])

Y $C_{16} = (89+201) \bmod 256$
$=290 \bmod 256$
$= 290$ (")

A $C_{17} = (65+210) \bmod 256$
$= 275 \bmod 256$
$= 19$ ([])

H $C_{18} = (72+219) \bmod 256$
$=291 \bmod 256$
$= 35$ (#)

P $C_{18} = (80+259) \bmod 256$
$= 339 \bmod 256$
$= 97$ (S)

U $C_{19} = (85+268) \bmod 256$
$= 353 \bmod 256$
$= 97$ (a)

T $C_{21} = (84+277) \bmod 256$
$= 361 \bmod 256$
$= 105$ (i)

R $C_{22} = (82+286) \bmod 256$
$= 368 \bmod 256$
$= 112$ (p)

I $C_{23} = (73+286) \bmod 256$
$= 368 \bmod 256$
$= 112$ (p)

Nilai desimal *chipertext* yang didapat adalah: 86, 86, 87, 94, 153, 145, 166, 163, 175, 188, 221, 243, 247, 215, 19, 34, 19, 35, 83, 97, 105, 112, 112.

3.4 Proses Dekripsi

Proses dekripsi dalam bidang kriptografi adalah langkah berlawanan dari proses enkripsi. Hal ini melibatkan mengembalikan pesan atau data yang telah diubah menggunakan algoritma kunci kriptografi yang identik dengan yang digunakan dalam proses enkripsi. Dalam proses dekripsi, pesan yang dienkripsi dikonversi kembali ke bentuk aslinya. Proses dekripsi dalam kriptografi memainkan peran penting dalam mengembalikan pesan

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

yang terenkripsi menjadi bentuk aslinya. Menjaga kerahasiaan kunci dan memverifikasi penggunaan kunci yang konsisten saat proses enkripsi dan dekripsi menjadi faktor penting untuk berhasil menguraikan pesan.

Tabel 4. *Chipertext Enkripsi*

														[		[					
v	v	w	^	™	‘	¦	¥	-	¼	Ý	ó	⁒	x	]	"	]	#	S	a	i	p	p
86	86	87	94	153	145	166	163	175	188	221	243	247	215	19	34	19	35	83	97	105	112	112

Tabel 5. Kunci Dekripsi

										10	11	15	16	17	18	19	20	21	22	25	26	27	28	29
kunci	4	13	22	62	71	80	89	98	107	116	156	165	174	183	192	201	210	219	259	268	277	286	295

Pi = (Ci-Ki) mod 256

P_1= (86-4) mod 256

=82 mod 256

= 82 (R)

P_2 = (86-13) mod 256

= 73 mod 256

= 73 (I)

P_3 = (87-22) mod 256

= 65 mod 256

= 65 (A)

P_4 = (94-62) mod 256

= 32 mod 256

= 32 (Spasi)

P_5 = (153-71) mod 256

= 82 mod 256

= 82 (R)

P_6 = (145-80) mod 256

= 65 mod 256

= 65 (A)

P_7 = (166-89) mod 256

= 77 mod 256

= 77 (M)

P_8 = (163-98) mod 256

= 65 mod 256

= 65 (A)

P_9 = (175-107) mod 256

= 68 mod 256

= 68 (D)

P_{10} = (188-116) mod 256

= 72 mod 256

= 72 (H)

P_{11} = (221-156) mod 256

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

= 65 mod 256

= 65 (A)

P_{12} = (243-165) mod 256

= 78 mod 256

= 78 (N)

P_{13} = (247-174) mod 256

= 73 mod 256

= 73 (I)

P_{14} = (215-183) mod 256

= 32 mod 256

= 32 (Spasi)

P_{15} = (87-22) mod 256

= 65 mod 256

= 65 (S)

P_{16} = (34-201) mod 256

= -167 mod 256

= 89 (Y)

P_{17} = (19-210) mod 256

= -191 mod 256

= 65 (A)

P_{18} = (35-219) mod 256

= -184 mod 256

= 74 (H)

P_{19} = (97-268) mod 256

= -176 mod 256

= 80 (P)

P_{20} = (97-68) mod 256

= -171 mod 256

= 85 (U)

P_{21} = (105-272) mod 256

= -167 mod 256

= 89 (T)

P_{22} = (112-286) mod 256

= -174 mod 256

= 82 (R)

P_{23} = (112-295) mod 256

= -183 mod 256

= 73 (I)

Berdasarkan dari perhitungan diatas, dapat disimpulkan bahwa setiap hasil mod yang bernilai desimal diubah kedalam bentuk karakter, sehingga nilai desimal *plaintext* yang diperoleh ialah: 82, 73, 65, 32, 82, 65, 77, 65, 68, 72, 65, 78, 73, 32, 83, 89, 65, 74, 80, 85, 89, 82, 73. Dari nilai desimal tersebut, dapat diketahui bahawa

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

nilai 82 merupakan sebuah karakter yang di beri huruf "R", begitu juga seterusnya sehingga dapat diketahui bahwa karakteristik *plaintext* dari proses dekripsi menghasilkan pesan: RIA RAMADHANI SYAHPUTRI.

4. KESIMPULAN

Setelah dilakukan penelitian, dapat diperoleh kesimpulan yaitu enggunaan algoritma Gronsfeld dapat diterapkan dengan mudah karena melibatkan hanya nilai-nilai plainteks dan dalam penelitian ini teks pesan yang digunakan berjumlah 23 kata dengan menggunakan nama RIA RAMADHANI SYAHPUTRI dan diubah dalam bentuk desimal dengan nomor desimal ialah 82, 73, 65, 32, 82, 65, 77, 65, 68, 72, 65, 78, 73, 32, 83, 89, 65, 72, 80, 85, 84, 82, 73. Sedangkan kunci pada proses enkripsi dan dekripsi yaitu 4, 13, 22, 62, 71, 80, 89, 98, 107, 116, 156,165,174,183,192, 201, 219, 259, 268, 277, 286, 295. Pada algoritma gronsfeld dengan proses enkripsi menggunakan perulangan 23 kali perhitungan dengan hasil akhir ialah 86, 86, 87, 94, 153, 145, 166, 163, 175, 188, 221, 243, 247, 215, 19, 34, 19, 35, 83, 97, 105, 112, 112 dilanjut proses dekripsi mengubah bentuk karakter. Pada proses dekripsi didapat hasil 82, 73, 65, 32, 82, 65, 77, 65, 68, 72, 65, 78, 73, 32, 83, 89, 65, 74, 80, 85, 89, 82, 73 dan diubah menjadi bentuk karakter menjadi bentuk semula *chipertext* RIA RAMADHANI SYAHPUTRI.

REFERENCES

[1] Y. Tumanggor, R. Maya, F. Lubis, M. P. Sianturi, and R. G. Purba, "Metode Algoritma Bubble Sort , Algoritma Merge Sort Dan Algoritma Quick Sort Dalam Pengujian Pengujian Perbandingan Proses Penelitian Kualitatif," *J. Tek. Inform. Komput. Univers.*, vol. 2, no. 2, pp. 47–58, 2022.

[2] I. V. Firmandia, M. Hannats, H. Ichsan, and R. Primananda, "Implementasi Quick Sort pada Cluster Computer menggunakan Raspberry Pi 3," vol. 5, no. 4, pp. 1630–1636, 2021.

[3] Z. Ramadhan and A. P. U. Siahaan, "Protection of Important Data and Information using Gronsfeld Cipher," *Int. J. Innov. Res. Multidiscip. F.*, vol. 4, no. 10, pp. 6–10, 2018, doi: 10.31227/osf.io/uq7nt.

[4] M. Luthfi Zulfa, B. Nurina Sari, and U. Singaperbangsa Karawang Abstract, "Analisis Perbandingan Algoritma Bubble Sort, Shell Sort, dan Quick Sort dalam Mengurutkan Baris Angka Acak menggunakan Bahasa Java," *J. Ilm. Wahana Pendidik.*, vol. 2022, no. 13, pp. 237–246, 2022.

[5] A. L. Noviani and I. Yuliani, "Perancangan Perangkat Lunak Kriptografi Menggunakan Gronsfeld Cipher, Vernam Cipher dan Ron Code 4 Stream Cipher," *Enter*, vol. 2, pp. 549–559, 2019.

[6] M. R. Hasibuan, "Implementasi Algoritma Quicksort Untuk Pembangkitan Kunci Algoitma RSA Pada Pengamanan Data Audio," vol. 2, no. 1, pp. 18–25, 2022.

[7] dkk 2018) richard oliver (dalam Zeithml., "d.richard oliver," *Angew. Chemie Int. Ed. 6(11), 951–952.*, pp. 2013–2015, 2021.

[8] N. D. M. Veronica and Y. Darnita, "Rancang Bangun Aplikasi Tes Toefl Menggunakan Algoritma Quick Sort Berbasis Komputer," *Pseudocode*, vol. 2, no. 2, pp. 89–97, 2015, doi: 10.33369/pseudocode.2.2.89-97.

[9] A. Miftachul, I. Al Amin Husni, and J. Wibowo Sasongko, "Implementasi Metode Quick Sort Dalam Sistem Pemeriksaan Pasien Klinik Fisioterapi Widya Husada Semarang," vol. 3, no. 2019, pp. 326–331, 2019.

[10] S. Kriptokompresi, M. Metode, G. Code, D. A. N. Algoritma, G. Cipher, and P. File, "Indriyani Angelina, Sistem kriptokompresi menggunakan metode goldbach code dan algoritma simetris gronsfeld cipher pada file text," 2022.

[11] M. KE, "Panny Agustia Rahayuningsih,"analisa perbandingan kompleksitas algoritma pengurutan nilai (sorting)" jurnal Vol.4, No.2,2016," □□□ □□□□□□□□□ □□□□ □□□□□□□ □□□□□□□ □□□□□ □□□□□ □□□□, vol. 147, no. March, pp. 11–40, 2016.

[12] I. Algoritma, Q. Untuk, P. Kunci, A. Rsa, and P. D. Audio, "PENGAMANAN DATA AUDIO," 2020.

[13] A. Rohmanu, "METODE ALGORITMA DES DAN METODE END OF FILE Ajar Rohmanu," *J. Inform.*, vol. 2, no. 1, pp. 1–11, 2017.

[14] R. Fadillah, A. S. Idris, D. Marlina, G. Lubis, and R. Meisisri, "Implementasi Algoritma Fast Encryption Algorithm (FEAL) Dan Algoritma Fibonacci Mengamankan File Teks," pp. 295–300, 2022.

[15] A. Ginting, R. R. Isnanto, and I. P. Windasari, "Implementasi Algoritma Kriptografi RSA untuk Enkripsi dan Dekripsi Email," *J. Teknol. dan Sist. Komput.*, vol. 3, no. 2, p. 253, 2015, doi: 10.14710/jtsiskom.3.2.2015.253-258.

[16] T. Limbong, "Pengujian Kriptografi Klasik Caesar Chipper Menggunakan Matlab," *Semin. Nas. Inov. dan Teknol. Inf. Sept.*, no. September 2015, pp. 77–80, 2015.

[17] D. R. Poetra, "Performa Algoritma Bubble Sort dan Quick Sort pada Framework Flutter dan Dart SDK(Studi Kasus Aplikasi E-Commerce)," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 2, pp. 806–816, 2022, doi: 10.35957/jatisi.v9i2.1886.

[18] P. T. Informatika, U. B. Darma, and P. Teks, "Implementasi Algoritma Quicksort Untuk Pembangkitan Kunci Algoritma Elgamal Pada Pengamanan Data File Dokumen," vol. 1, no. 1, pp. 17–24, 2022.

[19] K. V. Sedana, I. K. A. Mogi, I. B. G. Dwidasmara, I. G. A. Wibawa, C. R. A. Pramartha, and L. G. Astuti, "Implementasi Algoritma Gronsfeld Cipher dan Steganografi End Of File Untuk Pengamanan Data,"

ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 4, No. 4, August 2023, Page 985-996
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v4i4.4090

JELIKU (Jurnal Elektron. Ilmu Komput. Udayana), vol. 11, no. 1, p. 129, 2022, doi: 10.24843/jlk.2022.v11.i01.p14.

[20] M. A. Pratama, A. Desiani, and Irmeilyana, "Analisis Kebutuhan Waktu Algoritma InsertionSort, Merge Sort, dan Quick Sort denganKompleksitas Waktu," *Comput. Sci. ICT*, vol. ISBN, no. 1, pp. 31–34, 2017.

[21] B. O. Sinaga, D. Almahera, S. Wahyuni, I. Saputra, A. Gronsfeld, and A. D. A. N. Pembahasan, "Pengamanan File Docx Menerapkan Algoritma Gronsfeld," pp. 415–419, 2020.

[22] D. Iqbal, A. R. Panggabean, I. W. Sinaga, and ..., "Implementasi Algoritma RC4+ Untuk Mengamankan Pesan Teks Pada Aplikasi Chatting," *... Teknol. Komput. ...*, pp. 916–920, 2019.

[23] S. Maulidin, "Rancangan Sistem Keamanan Algoritma Gronsfeld Cipher dengan Pembangkit Bilangan Acak LCG (Linear Congruential Generator)," *J. Panca Budi*, vol. 1, no. 1, pp. 20–22, 2021.

[24] Y. Y. P. Rumapea, "Analisis Perbandingan Metode Algoritma Quick Sort Dan Merge Sort Dalam Pengurutan Data Terhadap Jumlah Langkah Dan Waktu," *J. Method.*, vol. 3, no. 2, pp. 2442–7861, 2017.