KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)
Volume 6, Nomor 1, November 2022
DOI: 10.30865/komik.v6i1.5784
ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 20-23

Penerapan Kode Golay Diperpanjang pada Kriptosistem McEliece

Ilham Setyawan*, Putranto Hadi Utomo

Fakultas Matematika dan Ilmu Pengetahuan Alam, Prodi Matematika, Universitas Sebelas Maret, Surakarta, Indonesia
Email: *ilhams161200@student.uns.ac.id, putranto@staff.uns.ac.id

Abstrak-Pada era modern ini, muncul komputer kuantum yang dapat menghitung operasi matematika lebih cepat dibandingkan komputer klasik. Beberapa kriptosistem dapat dipecahkan dengan komputer tersebut, akan tetapi terdapat kriptosistem yang belum terpecahkan menggunakan komputer tersebut salah satunya kriptosistem McEliece. Kriptosistem McEliece menggunakan kode koreksi kesalahan sehingga lebih aman dibandingkan kriptosistem yang menggunakan faktorisasi. Ide dasar kriptosistem ini yaitu seseorang dapat memberikan kesalahan pesannya sehingga pesannya tidak dapat dibaca oleh penyerang. Kode Golay merupakan kode dengan panjang 24-bit yang dapat mengoreksi kesalahan hingga 3 bit kesalahan. Pada artikel ini akan dijelaskan penerapan kode Golay pada kriptosistem McEliece.

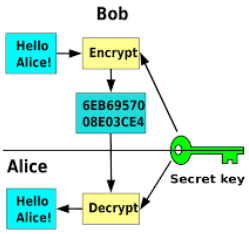
Kata Kunci: Kriptografi, Komputer Kuantum, Kriptosistem McEliece, Kode Koreksi Kesalahan, Kode Golay

Abstract-In this modern era, quantum computers appear that can calculate mathematical operations faster than classical computers. Some cryptosystems can be solved with the computer, but there are cryptosystems that have not been solved using the computer, one of which is the McEliece cryptosystem. The McEliece cryptosystem uses error correction codes so that it is more secure than cryptosystems that use factorization. The basic idea of this cryptosystem is that someone can give an error message so that the message cannot be read by an attacker. Golay code is a code with a length of 24-bits that can correct errors up to 3 bit errors. This article describes the implementation of Golay code on the McEliece cryptosystem.

Keywords: Cryptography, Quantum Computers, McEliece Cryptosystem, Error-Correction Code, Golay Code

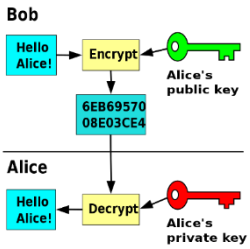
1. PENDAHULUAN

Berdasarkan Matius [1] kriptografi dapat dibedakan menjadi kriptografi kunci simetri dan kriptografi kunci asimetri (publik). Skema kriptografi kunci simetri diilustrasikan pada Gambar 1.



Gambar 1. Skema Kriptografi Kunci Simetris

Berdasarkan Gambar 1, kriptografi kunci simetri menggunakan kunci yang sama pada proses enkripsi dan dekripsi. Keamanan menggunakan sistem ini terletak pada kerahasiaan kunci yang akan digunakan. Sedangkan skema kriptografi kunci public diilustrasikan pada Gambar 2.



Gambar 2. Skema Kriptografi Kunci Asimetris

Berdasarkan Gambar 2, terlihat bahwa skema kriptografi kunci publik digunakan kunci yang berbeda dalam proses enkripsi dan dekripsi. Kunci publik digunakan untuk proses enkripsi, dan kunci privat digunakan untuk mendekripsikan pesan. Kunci publik bersifat tidak rahasia, sedangkan kunci privat hanya boleh diketahui oleh penerima pesan.

Komputer kuantum adalah komputer yang operasinya diatur oleh hukum mekanika kuantum[2]. Komputer kuantum dikembangkan agar dapat membantu peneliti dalam perhitungan. Jika komputer klasik menghitung suatu operasi tertentu memerlukan beberapa tahun atau bahkan berabad-abad maka komputer kuantum hanya butuh waktu beberapa jam saja atau beberapa hari saja.Setelah era pasca komputer kuantum, beberapa kriptosistem menjadi lemah. Tabel 1 merupakan dampak dari komputer kuantum pada beberapa kriptosistem[2]. Algoritma Shor merupakan algoritma yang berpotensi dapat meretas beberapa kriptosistem, salah satunya yaitu algoritma RSA dan kriptografi kurva eliptik[3]. Semua kriptosistem kunci publik dapat diserang dalam waktu polinomial menggunakan algoritma Shor.

ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 20-23

Tabel 1. Jenis jenis database

Kriptosistem	Pasca Komputer Kuantum
Diffie-Hellman *key-exchange* [4]	Rusak
RSA *public-key encryption* [5]	Rusak
Algebraically Homomorphic [6]	Rusak
Buchmann-Williams *key-exchange [7]*	Rusak
Elliptic curve cryptosystems [8]	Rusak
NTRU *public key encryption* [9]	Belum Rusak
McEliece *public key encryption* [10]	Belum Rusak
Lattice-*based public key encryption* [11]	Belum Rusak

Kriptosistem McEliece adalah algoritma kunci asimetris yang dibangun pada tahun 1978 oleh Robert McEliece. Kriptosistem McEliece memanfaatkan kode koreksi kesalahan sebagai mekanisme enkripsi. Ide dasarnya yaitu seseorang dapat dengan sengaja menambahkan kesalahan ke codeword untuk mengaburkan/mengenkripsi pesan dan mengoreksi kesalahannya pada saat mendekripsi pesan. Dengan penambahan kesalahan tersebut, kriptosistem ini tidak dapat diserang dengan menggunakan komputer kuantum.

Kode Golay pertama kali ditemukan oleh Golay [12] di 1949. Kode Golay 23-bit adalah kode yang sangat berguna, terutama untuk aplikasi tersebut ketika bit paritas ditambahkan untuk setiap kata untuk menghasilkan kode setengah tingkat. Diantaranya Kode Golay (24, 12, 8) yang digunakan untuk memberikan kontrol kesalahan pada misi voyager[13]. Algoritma decoding aljabar untuk kode Golay (23, 12, 7) diberikan oleh Elia [14] untuk mengoreksi tiga kemungkinan kesalahan. Pada tahun 1990, pendekatan decoding lain dikembangkan di[15], disebut shift-search prosedur decoding. Seperti yang ditunjukkan pada[15], pencarian shift prosedur ini membandingkan kompleksitas dan kecepatan dengan metode decoding Elia. Aljabar teknik ini sedikit lebih cepat daripada shift-search prosedur.

Kriptosistem McEliece menggunakan matriks generator dan matriks cek-paritas pada proses enkripsi maupun dekripsinya. Banyak sekali kode yang dapat digunakan sebagai matriks generator dan cek-paritas, seperti Kode Goppa, Kode Golay, Kode Hamming, dll. Pada penelitian ini akan diterapakan Kode Golay pada Kriptosistem McEliece.

2. METODOLOGI PENELITIAN

2.1 Proses Pembentukan Kunci

Proses pembentukan kunci merupakan proses penting yang diperlukan pada proses *decoding* dan *encoding*. Terdapat dua kunci pada kriptosistem ini yaitu kunci pribadi dan kunci publik. Kunci pribadi terdiri dari matriks S, matriks P, dan proses *decoding* pada matriks generator yang digunakan. Matriks S adalah matriks pengacak yang digunakan untuk menyamarkan matriks generatornya. Matriks S merupakan matriks non-singular dengan panjang kxk dimana k merupakan dimensi dari kode linier. Matriks P merupakan matriks permutasi yang digunakan untuk menambah tingkat ketidakjelasan pada matriks generatornya. Matriks P memiliki panjang nxn, dimana n merupaka panjang kode linier. Matriks permutasi merupakan matriks dengan tepat satu bit bukan nol di setiap baris dan kolomnya. Matriks permutasi dapat diperoleh dengan mengubah kolom dari matriks identitas. Proses *decoding* yang digunakan adalah proses *decoding* untuk kode Golay diperpanjang. Sedangkan kunci publiknya merupakan perkalian matriks dari matriks S, matriks G, dan matriks P. Matriks G merupakan matriks generator dari kode golay. Kemudian, kalikan matriks G dengan matriks P sehingga menjadi matriks G_1 dan kalikan matriks G_1 dengan matriks S sehingga menjadi G'. G' tersebut yang merupakan kunci public dari kriptosistem ini. Algoritma pembentukan kunci:

a. Ambil generator matriks dari kode Golay(24,12,8) yang dapat mengkodekan $(k = 12)$ bit data dalam satu *word* dengan panjang $(n = 24)$ dan $(t = 3)$ kesalahan yang dapat diperbaiki.
b. Buat matriks P dengan panjang nxn yang merupakan matriks permutasi
c. Buat matriks S dengan panjang kxk yang merupakan matriks non-singular yang mempunyai invers
d. Kalikan matriks G dengan matriks P:

a. $G_1 = GP$ (1)

e. Kalikan matriks G_1 dengan matriks S:

a. $G' = SG_1$ (2)

b. G' tersebut merupakan kunci public dari kriptosistem ini

2.2 Proses Enkripsi

Pada proses enkripsi diperlukan kunci publik dan kesalahan yang akan ditambahkan dengan bobot maksimal 3 karena kode Golay hanya bisa mengoreksi 3 kesalahan. Untuk melakukan proses enkripsi, pengirim membuat pesan m kemudian mengubah pesan tersebut ke biner, setelah menjadi biner, dilakukan pembagian panjang kode biner menjadi panjang dari k. Setelah itu kode tersebut dikalikan dengan G' sehingga menjadi *codeword* dengan panjang n. kemudian ditambahkan kesalahan pada *codeword* tersebut sehingga menjadi *ciphertext*. Algoritma *encoding*:

a. Hitung *codeword*:

$c = mG'$ (3)

ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 20-23

b. Tambahkan kesalahan pada *codeword*:

$$y = c + e \quad (4)$$

2.3 Proses Dekripsi

Pada proses dekripsi diperlukan invers matriks P, invers matriks S, dan proses *decoding* untuk kode Golay diperpanjang. Untuk melakukan proses dekripsi penerima melakukan perkalian vektor c terhadap invers matriks P, kemudian melakukan proses *decoding* untuk kode Golay diperpanjang. Langkah pertama proses *decoding yaitu* menghitung sindrom menggunakan kunci pribadi G_1 untuk memeriksa apakah bobot sindrom s_1 kurang dari atau sama dengan 3. Jika ya, maka mengembalikan vektor kesalahan $e = [s1, 000000000000]$. Jika tidak, ia memeriksa bobot $(s_1 + A_i)$ kurang dari atau sama dengan 2, maka vektor kesalahannya adalah $e = [s_1 + A_i, j_i]$. Jika tidak memenuhi kondisi pertama, maka selanjutnya menghitung sindrom kedua s_2 dan memeriksa apakah bobot sindrom s2 kurang dari atau sama dengan 3. Jika ya, maka kembali ke vektor kesalahan $e = [000000000000, s_2]$. Jika tidak, ia memeriksa bobot $(s_2 + A_i)$ kurang dari atau sama dengan 2, maka vektor kesalahannya adalah $e = [j_i, s_2 + A_i]$. Bagaimanapun, jika kedua kondisi tidak memenuhi dan pola kesalahan *e* belum ditentukan, maka diperlukan pengiriman ulang. Selanjutnya, mS ditemukan dan pesan asli dihitung dengan mengalikan mS dengan S^{-1}. Algoritma *decoding*:

a. Hitung sindrom pertama:
b. $s_1 = cG_1$ (5)
c. Jika $wt(s_1) \leq 3$, maka $e = [s1, 000000000000]$
d. Jika $wt(s_1 + A_i) \leq 2$, maka $e = [s_1 + A_i, j_i]$, dimana j_i *word* dengan panjang 12 dengan 1 di posisi i^{th} dan 0 di tempat lain dalam matriks identitas I_{12}
e. Jika tidak, hitung sindrom kedua:
f. $s_2 = s_1 A$ (6)
g. Jika $wt(s_2) \leq 3$, maka $e = [000000000000, s_2]$
h. Jika $wt(s_2 + A_i) \leq 2$, maka $e = [j_i, s_2 + A_i]$
i. Jika tidak, kesalahan belum bisa ditentukan, maka diperlukan pengiriman ulang

3. HASIL DAN PEMBAHASAN

3.1 Contoh perhitungan Kriptosistem McEliece

Berikut diberikan contoh perhitungan dari Kriptosistem McEliece menggunakan kode Golay. Pada penelitian ini perhitungan dilakukan dengan bantuan komputer. Dimisalkan sebuah pesan akan dikirimkan, sebelum pesan dikirimkan, dilakukan proses pembangkitan kunci.

Setelah mendapatkan kunci pribadi dan kunci publik, selanjutnya dilakukan proses pengenkripsian pesan. Pertama menginput pesan yang akan dikirim. Dimisalkan pesan yang akan dikirim berisi kata ”Pesan”, maka $m =$ Pesan. Sebelum perhitungan dilakukan, isi pesan terlebih dahulu diubah ke dalam sistem bilangan biner dengan memanfaatkan standar UNICODE. Setiap karakter pada kata ’Pesan’ diubah ke dalam standar UNICODE dihasilkan bilangan $[80, 101, 115, 97, 110]$ kemudian dari bilangan-bilangan tersebut diubah ke dalam biner dan dihasilkan $[01010000, 01100101, 01110011, 01100001, 01101110]$, selanjutnya bilangan biner digabungkan menjadi $[01010000011001010111001101100000101101110]$. Setelah isi pesan diubah, perhitungan dapat dilanjutkan.

Berdasarkan proses enkripsi diatas dari kata ”Pesan” didapatkan ciphertext sebagai berikut. Keluaran pada proses enkripsi dalam bentuk UNICODE. Setelah proses enkripsi selesai, langkah selanjutnya mengembalikan pesan tersebut ke bentuk awal. Untuk mengembalikan pesan tersebut, dilakukan proses dekripsi. Menggabungkan append, Kemudian pesan diatas digabungkan menjadi satu bagian. Hasil dari penggabungannya yaitu m=0000000010000101000001100101011100110110000101101111011111111. Unpadding bit, Selanjutnya, dilakukan unpadding agar pesan tersebut dapat diubah dari biner ke UNICODE. hasil dari unpaddingnya adalah m=01010000011001010111001101100000101101110. Parsing pesan, Pesan yang telah diterima kemudian diparsing menjadi 8 bit agar dapat dibaca oleh bilangan biner. Hasilnya menjadi [01010000,01100101,01110011,01100001,01101110] Mengubah ke UNICODE Kemudian dilanjutkan mengubah ke bentuk UNICODE menjadi bilangan [80,101,115,97,110], dan diubah ke huruf menjadi [P,e,s,a,n]. Output,Output dari proses dekripsi ini yaitu kata “Pesan”, karena kata “Pesan” merupakan kata yang diinputkan tadi, maka proses dekripsi berjalan dengan lancar.

3.2 Implementasi McEliece menggunakan python

Pada penelitian ini dirancang sebuah program sederhana yang dapat menjelaskan secara singkat bagaimana cara kerja kriptosistem McEliece menggunakan kode Golay. Program yang dirancang terbagi menjadi dua program, yaitu program pembangkitan kunci dan program enkripsi-dekripsi. SageMath merupakan software dan library matematika untuk menangani fungsi matematika yang rumit. SageMath menggunakan lisensi GPL, dan termasuk open source. Didalamnya terdapat kumpulan library matematika dan statistika seperti NumPy, SciPy, matplotlib, Sympy, Maxima, GA, FLINT, R dan masih banyak lainnya. Kombinasi library tersebut bila dikombinasikan akan membuat mudah pekerjaan analisis data pembaca seperti kriptografi, matematika, dan statistika.

3.2.1 Pembangkitan Kunci

ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 20-23

Pada program ini di impor modul random untuk mengacak karakter pada program. Pertama, buat matriks generator untuk kode Golay dengan *library* yang ada pada *Sagemath*, yaitu dengan sintaksis $\boldsymbol{golay = codes.GolayCode(GF(2), extended = True)}$. Selanjutnya membuat matriks S dengan fungsi tersebut dengan menjalankan sintaksis $\boldsymbol{s = non\_singular\_matrix(GF(2), 12)}$. Selanjutnya membuat matriks S dengan fungsi tersebut dengan menjalankan sintaksis $\boldsymbol{p = rand\_perm(GF(2), 24)}$. Setelah mendapatkan ketiga matriks tersebut, kemudian membuat fungsi *error* vektor.Selanjutnya, membuat *error* vector berbobot tiga dengan perintah sintaksis $\boldsymbol{er = error\_vector(24, 3)}$.

3.2.2 Proses Enkripsi-Dekripsi

Pada program ini dibuat beberapa fungsi untuk mempermudah proses enkripsi, Selanjutnya untuk menjalankan enkripsi dan dekripsinya, diperlukan *input* berupa *string*. Diberikan *input* kata berupa "Pesan" sehingga $\boldsymbol{msg = "Pesan"}$. Dengan menjalankan sintaksis $\boldsymbol{mceliece(msg)}$, dihasilkan *output*]" □ > ! Ú□çð Ò = Å

4. KESIMPULAN

Berdasarkan hasil dan pembahasan dapat disimpulkan bahwa kode Golay diperpanjang dapat digunakan pada kriptosistem McEliece, sehingga bit *error* yang bisa ditambahkan hingga 3 bit *error*. Pada penelitian sebelumnya, kriptosistem McEliece menggunakan kode Goppa, dimana kode Goppa hanya dapat mengoreksi *error* hingga 2 bit saja dan panjang kata yang dienkripsi hanya 4 bit. Sedangkan pada kode Golay diperpanjang, *error* yang dapat dikoreksi hingga 3 bit dan panjang kata yang dienkripsi yaitu 12 bit. Contoh perhitungan algoritma kriptosistem McEliece dengan kode Golay diperpanjang menggunakan komputer dapat diselesaikan dengan baik dan dijelaskan setiap tahapan dalam pembangkitan kunci, proses enkripsi dan proses dekripsi. Selain itu konsep kriptosistem McEliece dengan kode Golay diperpanjang berhasil dipraktikkan ke dalam bahasa pemrograman python dengan memanfaatkan library Sagemath. Program tersebut berhasil dijalankan dan berhasil mengenkripsi pesan maupun mendekripsi pesan ke semula. Kriptosistem ini dapat diterapkan di era pasca komputer kuantum. Kriptosistem ini aman karena setiap pesan yang akan dikirim diberikan beberapa bit *error* agar pesan sulit diserang. Akan tetapi kriptosistem ini memerlukan penyimpanan yang besar karena memerlukan beberapa kunci.

REFERENCES

[1] M. C. Sinaga, "Kriptografi Python," *INA-Rxiv*, 2017.
[2] D. J. Bernstein, "Introduction to post-quantum cryptography," in *Post-quantum cryptography*, 2009.
[3] L. Chen *et al.*, "Report on post-quantum cryptography," National Institute of Standard and Technology, US, 2016.
[4] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE transactions on Information Theory*, 1976.
[5] R. L. Rivest, L. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," *Foundations of secure computation*, 1978.
[6] R. L. Rivest, A. Shamir, and L. Adleman, " A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, 1978.
[7] J. Buchmann and H. C. Williams, "A key-exchange system based on imaginary quadratic fields," *Journal of Cryptology*, 1988.
[8] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of computation*, 1987.
[9] J. Hoffstein, J. Pipher, and J. H. Silverman, "ntru: A ring-based public key cryptosystem," in *International Algorithmic Number Theory Symposium*: Springer, 1998.
[10] R. J. McEliece, "A public-key cryptosystem based on algebraic coding theory," *JPL DSN Progress Report*, 1978.
[11] J.-Y. Cai and T. W. Cusick, " A lattice-based public-key cryptosystem," in *International Workshop on Selected Areas in Cryptography*: Springer, 1998.
[12] M. J. E. Golay, "Notes on digital coding," *Proc. IRE*, p. 67, 1949.
[13] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*. Prentice Hall, 1995.
[14] M. Elia, "Algebraic decoding of the (23, 12, 7) Golay codes," *y, IEEE transactions on Information Theory*, vol. 33, pp. 150-151, 1987.
[15] I. S. Reed, X. Yin, T. K. Truong, and J. K. Holmes, "Decoding the (24,12,8) Golay code," *IEE Processings*, vol. 137, pp. 202-206, 1990.