sudo
Jurnal Teknik Informatika
https://jurnal.ilmubersama.com/index.php/sudo

Kriptografi

Pengamanan Data Teks Menggunakan Algoritma Kriptografi Elgamal dan XOR dari Serangan Hacker

Khairani, Mhd Zulfansyuri Siambaton

Fakultas Teknik, Program Studi Teknik Informatika, Unversitas Islam Sumatera Utara, Medan, Indonesia

INFORMASI ARTIKEL

Diterima Redaksi: 23 November 2023
Revisi Akhir: 22 Desember 2023
Diterbitkan *Online*: 27 Desember 2023

KATA KUNCI

Kriptografi; Elgamal; XOR; Teks

KORESPONDENSI

Phone: +62 822-6785-6862
E-mail: Khairaniiiii17@gmail.com

A B S T R A K

Keamanan data merupakan hal yang sangat penting dalam menjaga kerahasiaan informasi terutama yang berisi informasi sensitif yang hanya boleh diketahui isinya oleh pihak yang berhak saja, apalagi jika pengirimannya dilakukan melalui jaringan publik, apabila data tersebut tidak diamankan terlebih dahulu, akan sangat mudah disadap dan diketahui isi informasinya oleh pihak-pihak yang tidak bertanggung jawab. Oleh karena itu peneliti akan menerapkan algoritma *ElGamal* dan *XOR* untuk dapat memberikan keamanan data pada text. *Algoritma XOR* algoritma sederhana yang menggunakan prinsip logika. Sedangkan Algoritma ElGamal menitik beratkan kekuatan kuncinya pada pemecahan masalah logaritma diskrit. Dengan memanfaatkan bilangan prima yang besar serta masalah logaritma diskrit yang cukup menyulitkan. pesan teks memiliki keamanan yang berlapis karena memiliki banyak kunci dengan menggabungkan algoritma ElGamal dan XOR.

PENDAHULUAN

Perkembangan zaman membuat teknologi informasi dan komunikasi semakin maju. Proses bertukar pesan atau informasi menjadi semakin mudah dilakukan. Dalam proses bertukar pesan sangat penting menjaga keamanan pesan atau informasi agar pesan tersebut tidak dapat dimengerti oleh pihak lain maupun pihak yang tidak berwenang [1]. Dengan pesatnya perkembangan teknologi di dunia internet, masyarakat dapat mengumpulkan informasi, bertukar informasi dan berita dengan mudah, cepat dan bebas. Bebasnya akses informasi di internet juga dapat menimbulkan dampak negatif khususnya cybercrime seperti akses tidak sah terhadap data berita, penyalahgunaan informasi untuk keuntungan pribadi yang merugikan pengguna internet. Itulah mengapa penting untuk melindungi dan mengamankan pesan. Keberadaan pengamanan pesan bertujuan untuk melindungi pesan dari berbagai kejahatan dunia maya.

Kriptografi adalah seni atau ilmu untuk menghasilkan pesan rahasia. Pesan asli, disebut *plaintext*, disandikan menjadi pesan terenkripsi yang disebut dengan *ciphertext* melalui proses enkripsi, dan *ciphertext* diubah kembali menjadi *plaintext* melalui proses dekripsi. Kriptografi memiliki beberapa algoritma yang banyak digunakan untuk mengamankan informasi [2]. Salah satu algoritma kriptografi yang umum digunakan dalam keamanan adalah algoritma XOR. Algoritma XOR adalah algoritma yang sering digunakan dalam sandi yang menggunakan operasi bit demi bit dan termasuk dalam kriptografi klasik. Algoritma XOR juga merupakan algoritma sederhana yang menggunakan prinsip logika XOR. Untuk proses dimana proses enkripsi dilakukan dengan kunci XOR pada plaintext unntuk mendapatkan cipherteks. Pada proses dekripsi, ciphertext dikodekan dengan XOR dengan kunci untuk mendapatkan teks aslinya (plaintext). Proses enkripsi dan dekripsi tidak sulit dan mudah untuk diimplementasikan [3].

https://doi.org/10.56211/sudo.v2i4.401

Algoritma enkripsi OR atau XOR eksklusif adalah sebuah algoritma Kriptografi yang melakukan logika XOR pada setiap biner dalam teks [4]. Algoritma ElGamal adalah sepasang kunci yang dihasilkan dengan memilih bilangan prima p dan dua bilangan acak g dan x, dengan syarat nilai g dan x kurang dari p, yang memenuhi persamaan [5].

Algoritma ElGamal ini memiliki tingkat keamanan dalam pemecahan masalah logaritma diskret pada group pergandaan bilangan prima yang besar, maka upaya untuk memecahkan pesan yang telah dienkripsi menjadi sangat sulit. Selain tingkat keamanan pada pemecahan logaritma diskret, algoritma ElGamal memiliki kelebihan dalam menghasilkan *ciphertext* (pesan yang telah tersamarkan) yang berbeda untuk *plaintext* (pesan belum disamarkan, masih dapat dibaca dengan jelas) yang sama pada proses enkripsi, tetapi ketika *ciphertext* di dekripsi akan menghasilkan *plaintext* (pesan belum disamarkan, masih dapat dibaca dengan jelas) yang sama pada proses enkripsi, tetapi ketika *ciphertext* di dekripsi akan menghasilkan *plaintext* yang sama. Proses algoritma ElGamal terdiri atas 3 proses yaitu proses pembentukan kunci, proses enkripsi dan proses dekripsi. Setiap proses dalam algoritma ini menggunakan teori bilangan terutama bilangan prima dan modulo bilangan.

Namun di sisi lain, algoritma ElGamal juga mempunyai kekurangan yaitu membutuhkan resource yang besar dan processor yang mampu melakukan perhitungan besar. Meskipun memiliki kelemahan tersebut, namun algoritma ElGamal memiliki kelebihan yang jauh lebih banyak, sehingga dalam paper ini menggunakan algoritma ElGamal dalam meningkatkan keamanan data.

TINJAUAN PUSTAKA

Kriptografi

Kriptografi adalah studi tentang metode komunikasi yang aman antara dua belah pihak. Biasanya ada dua pihak yang saling mengirim pesan, tetapi mereka ingin menghindari kemungkinan pihak ketiga memahami isi dari pesan mereka jatuh kepada pihak yang salah [6].

Kriptografi adalah sebuah seni yang berfokus pada menyembunyikan dan mengirimkan pesan secara diam-diam. Banyak sandi yang digunakan sepanjang sejarah, banyak diantaranya sekarang dianggap tidak aman menurut standar modern. Nyatanya, baru pada pertengahan abad ke-20 kriptografi berubah dari seni menjadi sebuah sains [7].

Kriptografi telah digunakan selama ribuan tahun untuk membantu menyediakan rahasia komunikasi antara pihak-pihak yang saling percaya. Dalam bentuknya yang paling dasar, dua orang, sering dilambangkan sebagai Alice dan Bob, telah menyepakati kunci rahasia tertentu. Di lain waktu, Alice mungkin ingin mengirim pesan rahasia ke Bob (atau Bob mungkin ingin mengirim pesan ke Alice). Kunci digunakan untuk mengubah pesan asli (yang biasanya kita sebut dengan plaintext) menjadi bentuk acak yang tidak dapat dipahami kepada siapa saja yang tidak memiliki kunci. Proses ini disebut enkripsi, dan pesan yang diacak disebut ciphertext. Ketika Bob menerima ciphertext, dia dapat menggunakan kunci untuk mengubah ciphertext kembali menjadi plaintext atau teks asli, ini disebut dengan proses dekripsi.[8]

El-Gamal

El-Gamal adalah sistem enkripsi kunci asimetris yang ditemukan Taher El-Gamal pada tahun 1985. Algoritma ini merepresentasikan metode alternatif untuk cipher kunci publik RSA. Perbedaan utama antara algoritma El Gamal dan RSA adalah bahwa keamanan RSA bergantung pada kesulitan faktorisasi bilangan prima besar, sementara El-Gamal bergantung pada kesulitan dalam menghitung modulus logaritmik diskrit dari bilangan prima besar. Masalah logaritma diskrit adalah masalah sulit dalam matematika karena itu penting terutama pada konjungtur untuk mendapatkan semua solusi yang mungkin. Jadi sistem crypto ini hampir rusak tidak tersedia atau membutuhkan waktu lama. Terutama Keunggulan teknologi El Gamal adalah pesan teks yang sama menghasilkan pesan teks rahasia yang berbeda setiap saat jika dienkripsi [9].

Algoritma ElGamal adalah sepasang kunci yang dihasilkan dengan memilih bilangan prima p dan dua bilangan acak g dan x, dengan syarat nilai g dan x kurang dari p, yang memenuhi persamaan [5]. ElGamal dapat digunakan untuk tanda tangan digital dan enkripsi, keamanannya bergantung pada kesulitan menghitung logaritma diskrit dalam bidang

yang terbatas.

Untuk menghasilkan pasangan kunci, pertama pilih bilangan prima, p, dan dua bilangan acak, g dan x, sehingga g dan x keduanya lebih kecil dari p, lalu hitung = $g^X \, mod \, p$.

Kunci publiknya adalah y, g dan p. Baik g dan p dapat dibagikan oleh sekelompok pengguna. Kunci pribadinya adalah X [10].

Proses Pembentukan Kunci El-Gamal

Proses pembentukan kunci merupakan proses penentuan suatu bilangan yang kemudian akan digunakan sebagai kunci pada proses enkripsi dan dekripsi pesan. Kunci untuk enkripsi dibangkitkan dari nilai p, g, y sedangkan kunci untuk dekripsi terdiri dari nilai x, p. Masing-masing nilai mempunyai persyaratan yang harus dipenuhi.
Langkah-langkah dalam pembuatan kunci adalah sebagai berikut:

1. Pilih sembarang bilangan prima p, dengan syarat $p > 255$.
2. Pilih bilangan acak g dengan syarat $g < p$.
3. Pilih bilangan acak x dengan syarat $1 = x = p - 2$.
4. Hitung $y = g^p mod \, p$.

Kunci *public* adalah y, g, p sedangkan kunci *private* adalah x. Nilai y, g, dan p tidak dirahasiakan sedangkan nilai x harus dirahasiakan karena merupakan kunci *private* untuk mendekripsi *plaintext.*

Algoritma XOR

Teknik XOR melakukan enkripsi dan dekripsi terhadap sebuah informasi dengan menggunakan kunci tunggal dan operasi bit XOR.[11] Tabel logika dari operasi XOR adalah sebagai berikut:

Tabel 1. Tabel Logika Operasi XOR

A	B	A ⊕B
0	0	0
0	1	1
1	0	1
1	1	0

Proses Enkripsi XOR

Proses enkripsi atau dekripsi di awali dengan merubah setiap nilai plaintext ke biner Formula untuk melakukan proses enkripsi dan dekripsi adalah :

Enkripsi : $\boldsymbol{C_i = P_i \, XOR \, K_i}$

Deskripsi : $\boldsymbol{P_i = C_i \, XOR \, k_o}$

METODOLOGI

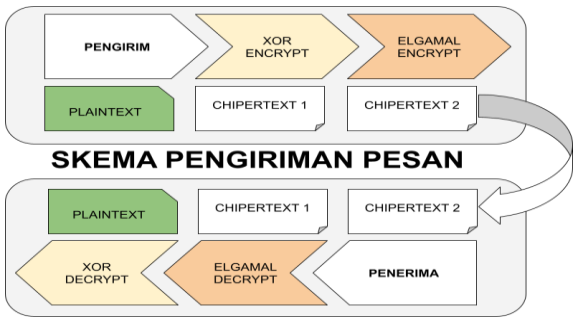
Pengertian Perancangan

Perancangan adalah Proses untuk mendefinisikan sesuatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta didalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya[12]. Perancangan adalah suatu proses untuk membuat dan mendesain sistem yang baru [13].

Berdasarkan pengertian diatas dapat disimpulkan bahwa perancangan sistem adalah sebuah proses setelah analisis dari siklus pengembangan sistem untuk merancang suatu sistem.

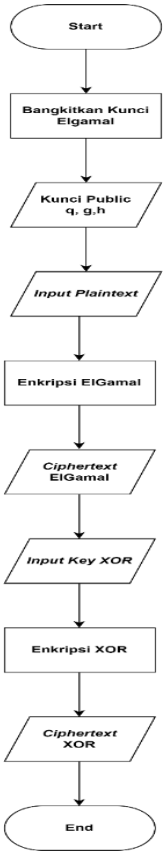
 https://doi.org/10.56211/sudo.v2i4.401

Rancangan Penelitian



Gambar 1. Skema Umum Pengiriman Pesan Menggunakan Algoritma Kriptografi El Gamal dan XOR

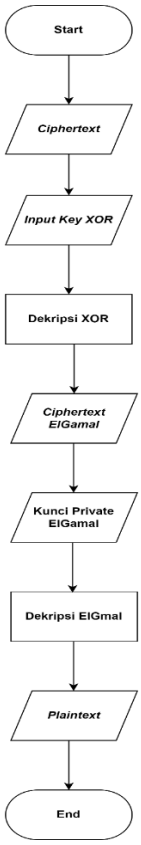
Pada gambar diatas dapat dilihat skema kriptografi hybrid yang mengimplementasikan penggabungan dua teknik kriptografi yang berbeda, yaitu ElGamal dan algoritma XOR, untuk pengiriman pesan. Pada tahap pertama, pesan asli dienkripsi menggunakan algoritma XOR dengan menggunakan kunci XOR yang diinputkan oleh pengguna. Proses ini bertujuan untuk memastikan kerahasiaan dan integritas pesan saat pengiriman. Setelah itu, pesan terenkripsi pertama (chipertext 1) yang dihasilkan dari tahap XOR akan dienkripsi kembali menggunakan skema ElGamal yang akan menghasilkan pesan terenkripsi kedua (chipertext 2) yang selanjutnya akan dikirimkan pada penerima. Pada bagian penerima dilakukan proses dekripsi dengan cara melakukan dekripsi elgamal pada pesan terenkripsi (chipertext 2) yang diterima sehingga menghasilkan pesan terdekripsi pertama (chipertext 1). Kemudian pesan terdekripsi pertama akan didekripsi kembali menggunakan proses dekripsi xor yang kemudian menghasilkan pesan aslinya. Untuk lebih mengetahui bagaimana kedua metode tersebut dapat mengamankan pesan dengan baik dapat dilihat melalui gambar dibawah ini.



Gambar 2. *Flowchart* Enkripsi Elgamal dan XOR

Pada gambar 2 dapat kita lihat *flowchart* enkripsi ElGamal dan XOR, dimana *flowchart* dimulai dari *start* kemudian bangkitkan (*Generate*) kunci elgamal yang nantinya akan menghasilkan nilai *q*, *g*, *h*. Kemudian setelah

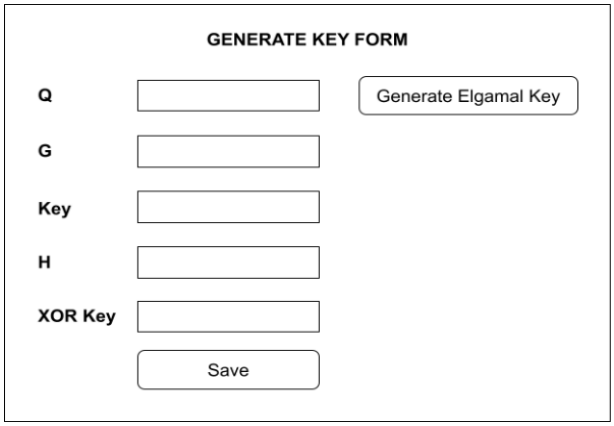
membangkitkan kunci, maka kita masukkan *plaintext* yang akan kita enkripsi. Kemudian enkrip *plaintext* dengan menggunakan algoritma ElGamal. Kemudian hasil enkripsi akan menghasilkan *ciphertext* dari algoritma ElGamal. Kemudian *ciphertext* ElGamal ini kita enkripsi kembali dengan menggunakan algoritma XOR dengan menggunakan kunci XOR. Kemudian *ciphertext* ElGamal tersebut sudah berhasil kita rubah menjadi *ciphertext* algoritm XOR. Kemudian untuk mendekripsi pesan yang telah di enkripsi dapat kita lihat pada gambar 3. berikut ini.



Gambar 3. *Flowchart* Dekripsi ElGamal dan XOR

Pada gambar 3 dapat kita lihat *flowchart* dekripsi algoritma ElGamal dan XOR dimana *flowchart* dimulai dari *Start.* Kemudian masukkan *ciphertext* yang akan didekripsi. Kemudian kita masukkan kunci XOR, kemudian proses dekripsi XOR dilakukan. Kemudian setelah proses XOR berhasil dilakukan, maka *ciphertext* akan berubah menjadi *ciphertext* algoritma ElGamal. *Ciphertext* dari algoritma ElGamal ini akan kita dekripsi Kembali dengan menggunakan kunci *private* algoritma ElGamal. Kemudian proses dekripsi algoritma ElGamal dilakukan yang nantinya *ciphertext* algoritma ElGamal akan menghasilkan *plaintext* seutuhnya atau menjadi pesan yang sempurna.

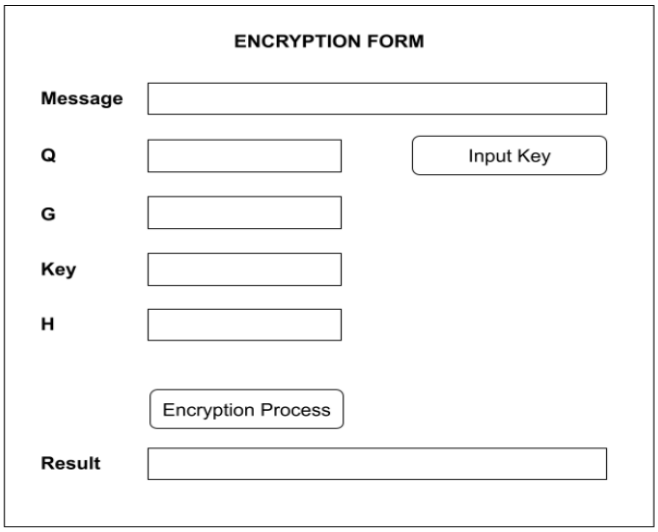
Perancangan Antarmuka



Gambar 4. Rancangan Antarmuka Pembangkit Kunci

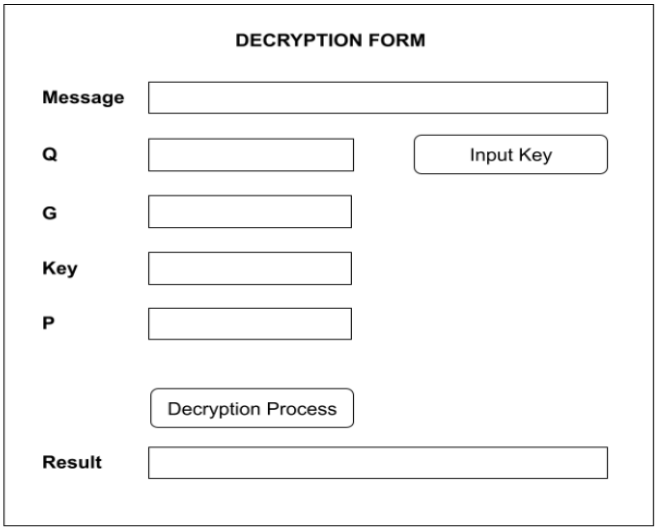
 https://doi.org/10.56211/sudo.v2i4.401

Pada gambar perancangan diatas dapat dilihat bahwa terdapat beberapa variabel yang akan dibangkitkan yaitu variabel *Q, G, H, Key* yang dapat dimasukkan secara manual atau atau otomatis melalui tombol *generate* dan *XOR Key* yang hanya dapat dimasukkan secara langsung oleh pengguna. Kemudian setelah pengguna menetapkan variabel-variabel yang dibutuhkan maka selanjutnya seluruh variabel tersebut akan secara otomatis disimpan kedalam sebuah file teks dengan nama *kunci_enkrip.txt.*



Gambar 5. Rancangan Antarmuka Enkripsi

Dari gambar diatas dapat kita lihat bahwa dalam antarmuka enkripsi memuat beberapa fungsi seperti fungsi untuk memasukkan pesan di kotak teks "Message" dan terdapat juga tempat untuk menampung kunci-kunci variabel elgamal seperti *Q, G, Key, H* yang bagian-bagian tersebut dapat diunggah dari sebuah file yang diperoleh dari proses pembangkitan kunci sebelumnya. Untuk melakukan proses enkripsi pengguna dapat menekan tombol "Encryption Process" untuk menghasilkan ciphertext yang bersamaan dengan itu juga menghasilkan file teks dengan nama kunci_dekrip.txt untuk proses dekrip nantinya



Gambar 6. Rancangan Antarmuka Dekripsi

Pada gambar diatas dapat dilihat bahwa rancangan antarmuka dekripsi memiliki kemiripan dengan enkripsi namun memiliki perbedaan pada variabel *P* dimana pada proses dekripsi menggunakan parameter P sebagai private key untuk melakukan dekripsi pada chipertext yang telah dimasukkan pada kotak teks pesan "Message".

HASIL DAN PEMBAHASAN

Hasil

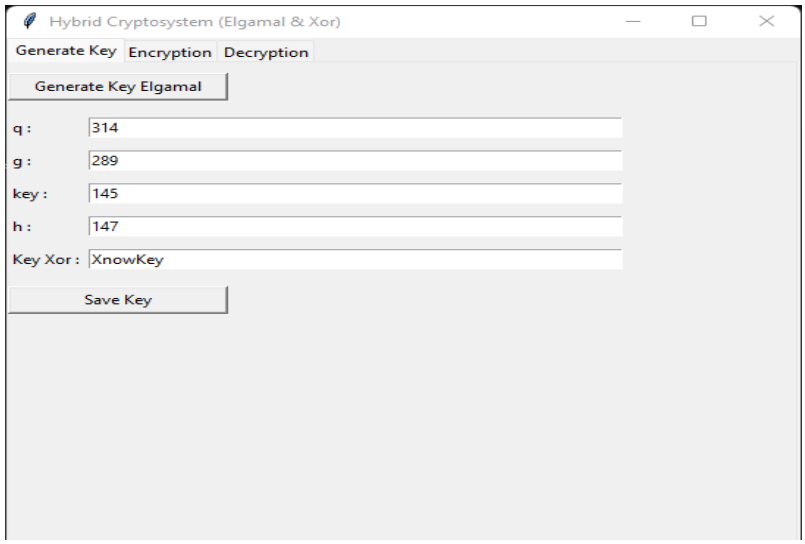
Dalam penelitian ini telah antarmuka telah berhasil dibuat antarmuka sistem yang berfungsi untuk mengamankan pesan menggunakan dua algoritma, yaitu ElGamal dan XOR. Antarmuka ini memungkinkan pengguna untuk memasukkan pesan langsung secara manual atau membangkitkan nilai variabel secara otomatis dengan tombol "generate". Variabel q, g, key, h, p dan Key XOR digunakan dalam proses enkripsi dan dekripsi.

Hasil dari percobaan menunjukkan bahwa antarmuka ini dapat berfungsi dengan baik. Pengguna dapat dengan mudah memasukkan pesan dan menghasilkan kunci enkripsi dan dekripsi. Setelah pesan dimasukkan, pengguna dapat menekan tombol "save" untuk menyimpan kunci enkripsi dan dekripsi dalam file teks.

Percobaan ini membuktikan bahwa antarmuka dan sistem yang dibuat berhasil mengimplementasikan algoritma ElGamal dan XOR untuk pengamanan pesan. Dengan demikian, antarmuka ini dapat menjadi solusi yang efektif untuk keamanan pesan dalam skenario tertentu. Namun, lebih lanjut evaluasi dan pengujian lebih mendalam mungkin diperlukan untuk memastikan keandalan dan keamanan sistem secara menyeluruh.

Pembangkitan Kunci Enhanced El-Gamal dan XOR

Tampilan halaman menu login ada pada gambar nomor 7 ini.

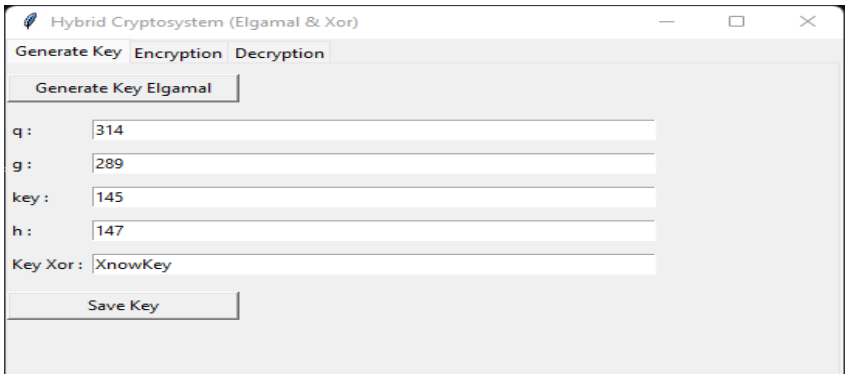


Gambar 7. Antarmuka Pembangkit Kunci ElGamal Dan Xor

Gambar diatas menunjukkan antarmuka berupa antarmuka yang memungkinkan pengguna memasukkan pesan secara manual. Form ini memiliki 5 variabel yaitu q, g, key, h, dan Key Xor, yang merupakan bilangan variabel yang relevan dengan penelitian ini. Nilai-nilai variabel tersebut dapat dimasukkan manual atau dihasilkan secara otomatis dengan menggunakan tombol generate. Setelah pengguna memasukkan semua pesan yang diinginkan, mereka dapat menekan tombol 'save' untuk menyimpan kunci enkripsi dalam sebuah file teks.

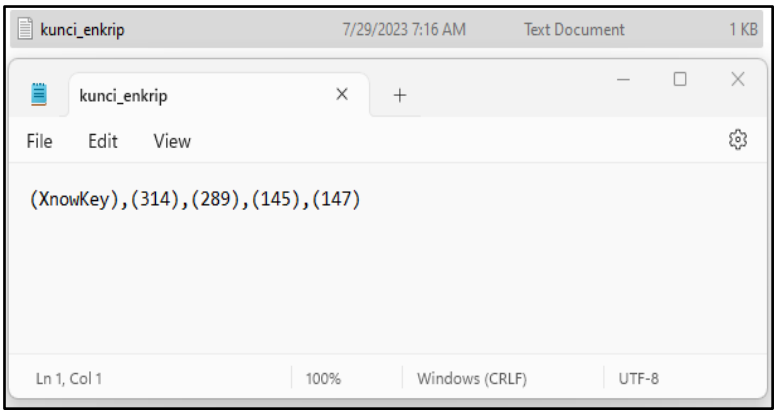
Sebuah percobaan telah dilakukan untuk menguji sistem yang dijalankan. Berikut ini adalah gambar hasil dari percobaan antarmuka pembangkitan kunci Elgamal dan XOR.

 https://doi.org/10.56211/sudo.v2i4.401



Gambar 8. Hasil Pembangkitan Kunci Elgamal Dan Xor

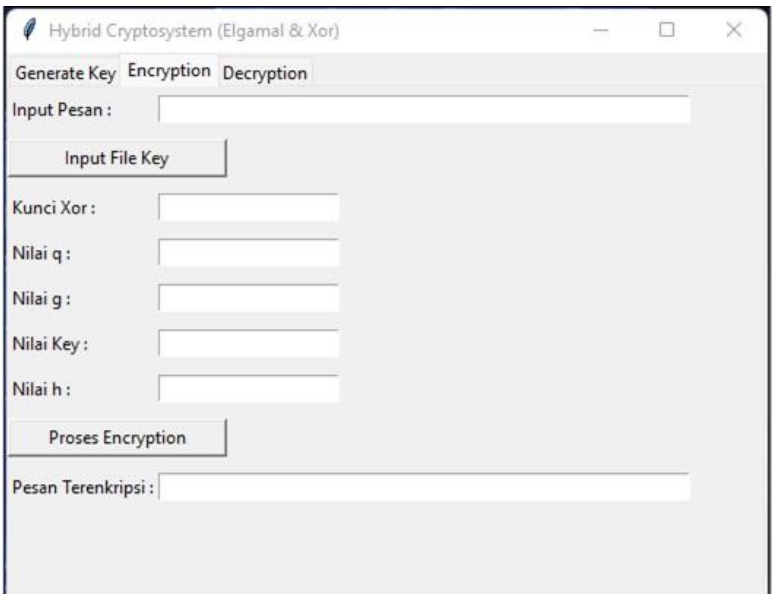
Pada gambar diatas dapat dilihat bahwa setiap variabel yang dibutuhkan telah dibangkitkan menggunakan tombol generate dan pengguna secara manual memasukkan *KeyXor*. Kemudian seluruh variabel tersebut akan disimpan sebagai kunci enkripsi pada sebuah file teks dengan nama *kunci_dekrip.txt*.



Gambar 9. Kunci Enkripsi

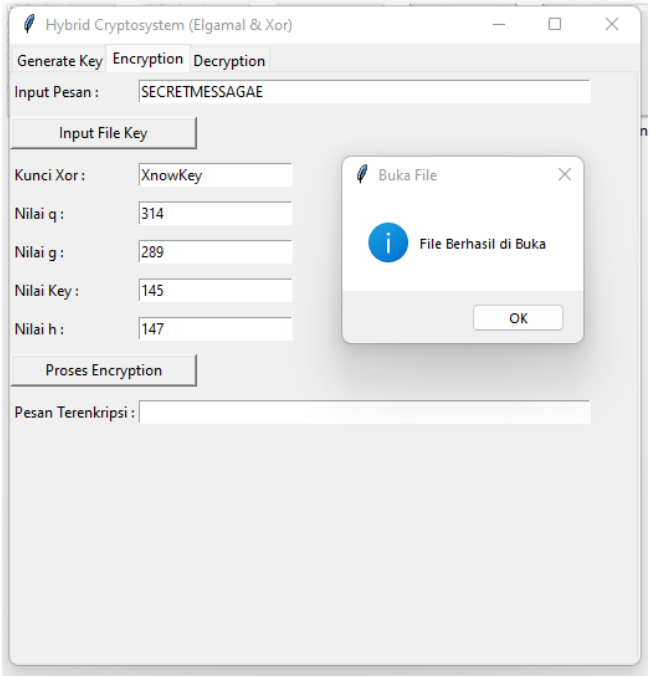
Dari gambar diatas dapat diketahui isi konten dari file *kunci_dekripsi.txt* memuat nilai-nilai dari proses pembangkitan kunci yang dilakukan sebelumnya.

Proses Enkripsi



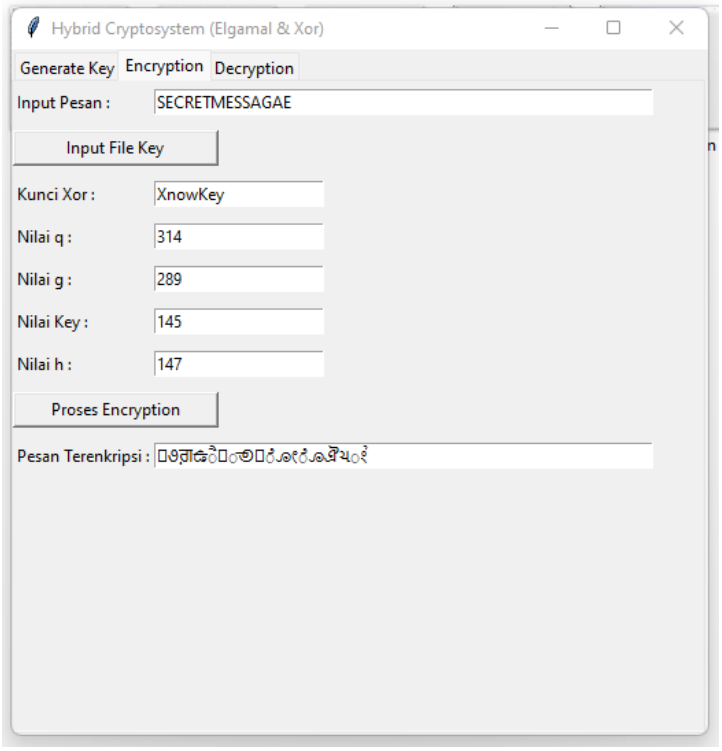
Gambar 10. Antarmuka Enkripsi

Pada gambar antarmuka di atas pesan nantinya akan dimasukkan ke dalam kotak “input pesan”. Kemudian kunci enkripsi akan diunggah melalui file yang telah dibuat sebelumnya yang selanjutnya akan dilakukan enkripsi untuk menghasilkan ciphertext pada kotak “Pesan Terenkripsi”. Untuk melakukan percobaan pada antarmuka ini kita lakukan proses kita akan memasukkan pesan dan kunci untuk menguji sistem yang telah dibuat.



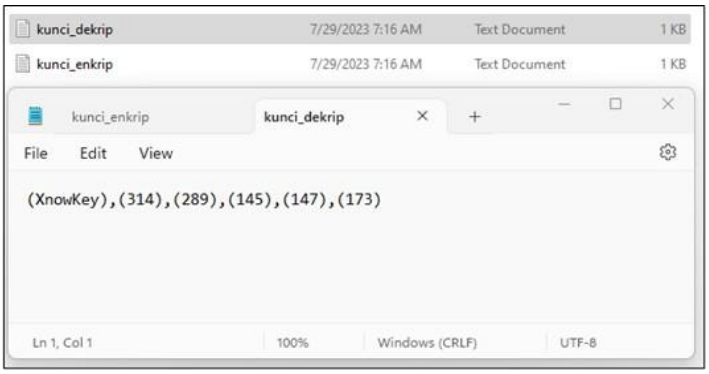
Gambar 11. Proses Unggah Kunci Enkripsi

Dari gambar diatas diketahui bahwa prose pengunggahan kunci berhasil dan memuat kunci-kunci yang sebelumnya dari file teks *kunci_enkrip.txt* yang sebelumnya. Setelah kunci berhasil diunggah dan pesan dimasukkan maka selanjutnya dilakukan proses enkripsi dengan menekan tombol “Proses *Encryption*” untuk menghasilkan ciphertext dari pesan yang dimasukkan. Berikut hasil dari proses enkripsi yang dilakukan.



Gambar 12. Hasil Enkripsi

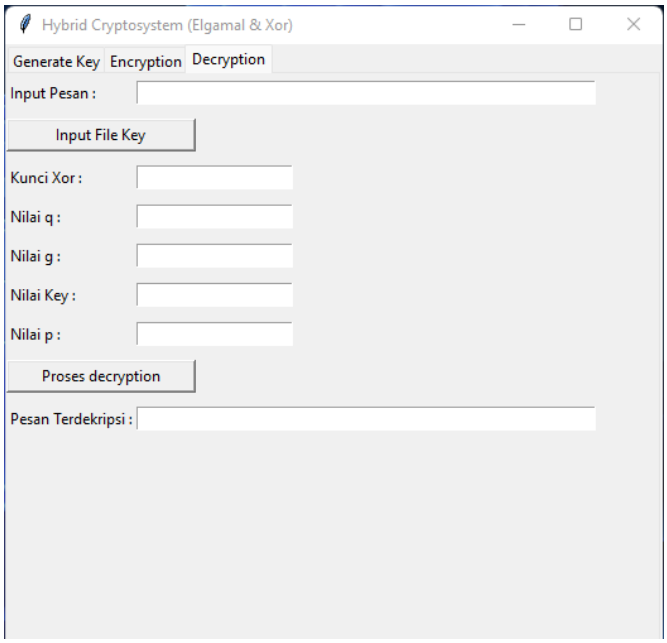
 https://doi.org/10.56211/sudo.v2i4.401



Gambar 13. Kunci Dekripsi

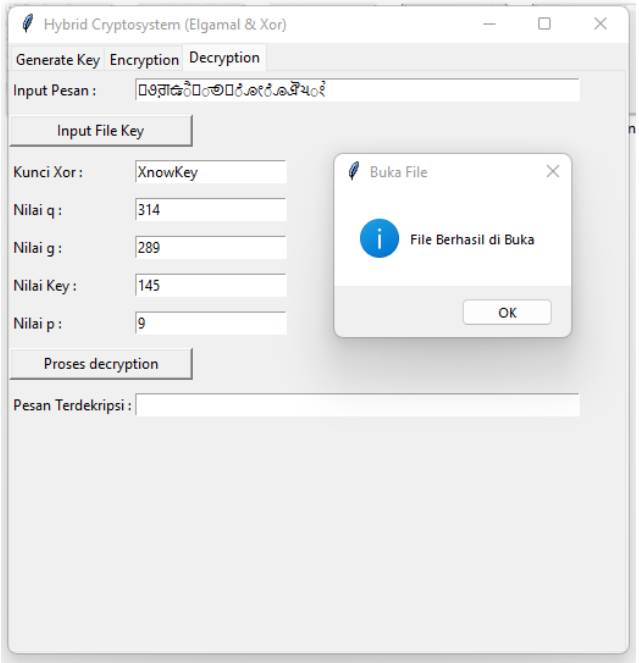
Dari gambar diatas dapat diketahui isi konten dari file *kunci_dekripsi.txt* memuat nilai-nilai dari proses pembangkitan kunci yang dilakukan sebelumnya.

Proses Deskripsi



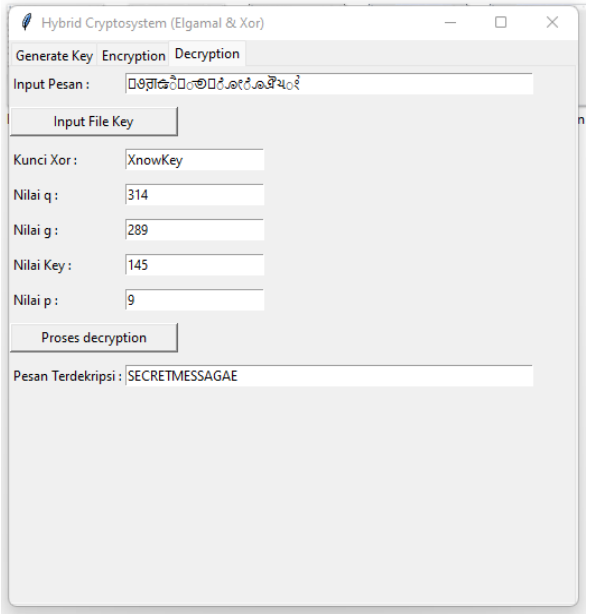
Gambar 14. Antarmuka Deskripsi

Pada gambar antarmuka di atas pesan nantinya akan dimasukkan ke dalam kotak “input pesan”. Kemudian kunci enkripsi akan diunggah melalui file yang telah dibuat sebelumnya yang selanjutnya akan dilakukan enkripsi untuk menghasilkan ciphertext pada kotak “Pesan Terenkripsi”. Untuk melakukan percobaan pada antarmuka ini kita lakukan proses kita akan memasukkan pesan dan kunci untuk menguji sistem yang telah dibuat.



Gambar 15. Proses Unggah Kunci Deskripsi

Dari gambar diatas diketahui bahwa prose pengunggahan kunci berhasil dan memuat kunci-kunci yang sebelumnya dari file teks *kunci_enkrip.txt* yang sebelumnya. Setelah kunci berhasil diunggah dan pesan dimasukkan maka selanjutnya dilakukan proses enkripsi dengan menekan tombol "Proses Encryption" untuk menghasilkan ciphertext dari pesan yang dimasukkan. Berikut hasil dari proses enkripsi yang dilakukan.



Ganbar 16. Hasil Deskripsi

Dari gambar diatas dapat dilihat bahwa pesan yang berupa chipertext berhasil dikonversi kembali menjadi pesan aslinya menggunakan kunci-kunci yang diunggah dari proses enkripsi sebelumnya.

KESIMPULAN DAN SARAN

Hasil proses enkripsi ini adalah Antarmuka dan sistem yang dibuat berhasil mengimplementasikan kedua algoritma dengan baik. Pengguna dapat dengan mudah memasukkan pesan, menghasilkan kunci enkripsi, dan menyimpan kunci tersebut dalam file teks. Dengan menggunakan aplikasi ini, pesan teks memiliki keamanan yang berlapis karena

 https://doi.org/10.56211/sudo.v2i4.401

memiliki banyak kunci dengan menggabungkan algoritma ElGamal dan XOR. Program masih berbasis desktop dan tidak online, sebaiknya dikembangkan sehingga dapat diakses secara online. Untuk tetap menjaga keamanan ciphertext hasil enkripsi dengan algoritma elgamal,kunci rahasia harus selalu dilindungi dari upaya manipulasi oleh pihak-pihak yang tidak bertanggung jawab. Algoritma XOR sangat sederhana meskipun efektif dalam beberapa konteks, disarankan agar dapat lebih kompleks dalam perhitungan matematiknya.

DAFTAR PUSTAKA

[1] Makhomah, R., Santoso, K. A., & Kamsyakawuni, A. (2021). Pengkodean Teks Menggunakan Kombinasi Hill Cipher dan Operasi XOR. PRISMA, Prosiding Seminar Nasional Matematika, 4, 548–552.

[2] Yusfrizal. (2019). Rancang Bangun Aplikasi Kriptografi Pada Teks Menggunakan Metode Reverse Cipher Dan Rsa Berbasis Android. Jurnal Teknik Informatika Kaputama (JTIK), 3(2), 29–3.

[3] Saputro, Pujo, H. (2023). Implementasi Algoritma Exclusive OR (XOR) Dalam Pengembangan Aplikasi Chat Berbasis Android. Informatika Fakultas Sains & Teknologi Universitas Labuhan Batu, 11(1), 71–76.

[4] Sulaiman, O. K., Nasution, K., & Siambaton, M. Z. (2020). Three Pass Protocol untuk Keamanan Kunci Berbasis Base64 pada XOR Cipher. Jurnal Sains Komputer & Informatika (J-SAKTI), 4(September), 721–727.

[5] Alfiah, F., Sudarji, R., & Taqiyyuddin Al Fatah, D. (2020). Aplikasi Kriptografi Dengan Menggunakan Algoritma Elgamal Berbasis Java Desktop Pada Pt. Wahana Indo Trada Nissan Jatake. 12260.

[6] Rubinstein_Salzedo, S. (2018). Cryptography. Springer Cham. https://doi.org/10.1007/978-3-319-94818-8.

[7] Iqbal, H., & Krawec, W. O. (2020). Semi-quantum cryptography. In Quantum Information Processing (Vol. 19,
Issue 3). Springer US. https://doi.org/10.1007/s11128-020-2595-9

[8] Stinson, D., & Paterson, M. (2019). Cryptography Theory and Practice Fourth Edition (Fourth Edi). Chapman & Hall. https://www.ptonline.com/articles/how-to-get-better-mfi-results

[9] Yousif, S. F., Abboud, A. J., & Radhi, H. Y. (2020). Robust Image Encryption with Scanning Technology, the El- Gamal Algorithm and Chaos Theory. IEEE Access, 8, 155184–155209. https://doi.org/10.1109/ACCESS.2020.3019216

[10] Jorgensen, P. (2003). Applied cryptography: Protocols, algorithm, and source code in C. Government Information Quarterly, 13(3), 336. https://doi.org/10.1016/s0740-624x(96)90083-0

[11] Sidik, A. P., Komputer, S., Sains, F., Pembangunan, U., Budi, P., Gatot, J. J., Km, S., Sikambing, S., Medan, K., & Utara, S. (2019). Teknik Xor Pada Mode Operasi Algoritma Cipher Block Chaining (CBC) Dengan Kunci Acak Blum Blum Shub Dalam Meningkatkan Keamanan Data.

[12] Analisa dan Perancangan Aplikasi Pembelajaran Bahasa Inggris Dasar Berbasis Android

[13] Nur, R., & Suyuti, M. A. 2018. Perancangan Mesin-Mesin Industri. Deepublish.