

Kombinasi Algoritma Elgamal dan Arithmetic Coding Pada Keamanan File Teks

Combination of ElGamal Algorithm and Arithmetic Coding for Text File Security

Samuel Fernando Manurung¹, Ria Eka Sari²

¹ Mahasiswa Universitas Potensi Utama, Jl.K.L Yos Sudarso KM 6.5 Tj.Mulia, Medan, 20241, Indonesia

² Dosen Universitas Potensi Utama, Jl.K.L Yos Sudarso KM 6.5 Tj.Mulia, Medan, 20241, Indonesia

E-mail: samuelfernando453@gmail.com¹, ladiespure@gmail.com²

ABSTRAK

Dalam lingkungan informasi, kerahasiaan data sangat penting untuk diperhatikan. Beberapa contoh kasus pada kebocoran data disebabkan oleh orang tak bertanggung jawab. Untuk mengatasi masalah ini diperlukannya metode yang mampu mengamankan file. Penulis menggunakan algoritma Elgamal dan Arithmetic Coding guna memperkuat keamanan file dan dapat memperkecil ukuran file, hasil dari penelitian ini berupa sebuah sistem yang mampu meningkatkan keamanan file dengan cara mengenkripsi dan mengompresi file. Algoritma ElGamal memberikan keamanan yang kuat dengan mengandalkan enkripsi kunci publik, di mana hanya kunci privat yang dapat digunakan untuk mendekripsi data. Ini membuat serangan brute-force sulit dilakukan. Di sisi lain, pengkodean aritmetika berfungsi mengompresi file teks, mengurangi ukurannya, sekaligus menambah kompleksitas pada data terenkripsi. Meskipun proses kompresi dapat sedikit memperlambat waktu pemrosesan, keuntungannya dalam menghemat ruang penyimpanan dan meningkatkan keamanan menjadikannya solusi yang efisien untuk melindungi file teks. Dari pengujian yang dilakukan diperoleh nilai rata-rata Compression Ratio (CR) adalah 2.12, Ratio of Compression (RC) adalah 48.68%, dan space saving (SS) adalah 51.41%. dari penelitian yang dilakukan sistem dapat mengenkripsi sekaligus mengompresi file dan dapat mengembalikan ke file semula.

Kata Kunci : Kriptografi, Elgamal, Arithmetic Coding, Kompresi

ABSTRACT

In the information environment, data confidentiality is crucial to maintain. Several data breach cases are caused by irresponsible individuals. To address this issue, methods capable of securing files are required. The author employs the ElGamal algorithm and Arithmetic Coding to enhance file security while also reducing file size. The result of this research is a system that improves file security by encrypting and compressing the files. The ElGamal algorithm provides strong security by relying on public key encryption, where only the private key can be used to decrypt the data, making brute-force attacks difficult to execute. On the other hand, arithmetic coding compresses text files, reducing their size while adding complexity to the encrypted data. Although the compression process might slightly slow down processing time, its advantages in saving storage space and enhancing security make it an efficient solution for protecting text files. The testing results showed an average Compression Ratio (CR) of 2.12, a Ratio of Compression (RC) of 48.68%, and a space-saving (SS) of 51.41%. Based on the research conducted, the system is able to encrypt and compress files simultaneously, as well as restore them to their original state.

Keywords : Cryptography, Elgamal, Arithmetic Coding, Compression

Disubmit: 18 Agustus 2024

Info Artikel :
Direview: 19 September 2024

Diterima : 15 Oktober 2024

Copyright © 2024 – senadimu. All rights reserved.

1. PENDAHULUAN

Kriptografi berasal dari bahasa Yunani dan terdiri dari dua suku kata: crypt dan graphia, yang berarti "menyembunyikan", dan graphia berarti "menulis". Kriptografi merupakan ilmu dan seni menjaga kerahasiaan pesan, selain definisi tersebut, terdapat juga definisi ilmu yang mengajarkan teknologi dan matematika yang mengacu pada aspek keamanan informasi seperti kerahasiaan data, validitas data, integritas data, dan otentikasi data [1]. Dalam jenis kunci yang dipakai pada kriptografi dibagi menjadi 2 jenis yaitu kriptografi simetris dan asimetris [2]. Algoritma elgamal adalah Teknik kriptografi kunci publik untuk mengenkripsi dan mendekripsi komunikasi adalah algoritma ElGamal. Algoritma ElGamal ditemukan pada tahun 1985 oleh Taher ElGamal, algoritma ini biasanya digunakan untuk tanda tangan digital, namun kemudian dimodifikasi sehingga dapat digunakan untuk enkripsi dan dekripsi [3].

Kompresi adalah pengubahan data yang berupa sekumpulan karakter menjadi bentuk kode dengan tujuan menghemat ruang penyimpanan dan waktu transfer data, karena data yang tergabung dalam setiap grup dapat diasumsikan memiliki properti yang sama/serupa, maka data dalam grup yang sama dapat dikompresi dengan merepresentasikannya menggunakan indeks prototipe setiap grup [4]. Dalam kompresi jenisnya diklasifikasikan menjadi dua yaitu Lossy Compression, Loseless Compression [5]. Algoritma Arithmetic coding melibatkan penggantian sekumpulan simbol masukan dengan angka floating point. Semakin panjang dan kompleks pesan terenkripsi, semakin banyak bit yang dibutuhkan. Hasil pengkodean aritmatika ini adalah angka yang kurang dari 1 dan lebih besar atau sama dengan 0 [6].

Penelitian yang dilakukan oleh Cathrin Desri Natalia Situmorang, dengan judul Penerapan "Algoritma Arithmetic Coding Pada Kompresi File Video Streaming Berbasis Android" penelitian ini menunjukkan bahwa penerapan algoritma arithmetic coding memberikan kinerja yang baik, dengan mencapai Ratio Of Compression sebesar (2,5), Compression Ratio sebesar (40%), penghematan ruang penyimpanan sebesar (60%), dan Redundancy sebesar (60%) [7].

Penelitian yang dilakukan oleh fahri husaini dkk. dengan judul "Penerapan Enkripsi Menggunakan Metode Elgamal guna Meningkatkan Keamanan Data Teks dan Gambar". penelitian ini melakukan penerapan elgamal pada teks dan gambar untuk teks mengambil sampel yaitu ABCDEF dan diubah kedalam kode ASCII untuk kunci privatnya 243 dan kunci publik 461. Penelitian ini berhasil memproses file dengan baik ketika di dekripsi file kembali ke file awal sebelum di enkripsi [8].

Penelitian yang dilakukan oleh luca De Feo dkk. dengan judul "On the (In) Security of Elgamal in OpenPGP". Penelitian ini memvalidasi asumsi tentang serangan ujung-ke-ujung dalam model ancaman SMT-colocated dengan membuat prototipe yang melibatkan proses penyerang dan korban. Penyerang membagi waktu menjadi N slot dan mengukur waktu akses cache, kemudian mengumpulkan 10.000 pengukuran dengan memicu korban secara berulang [9].

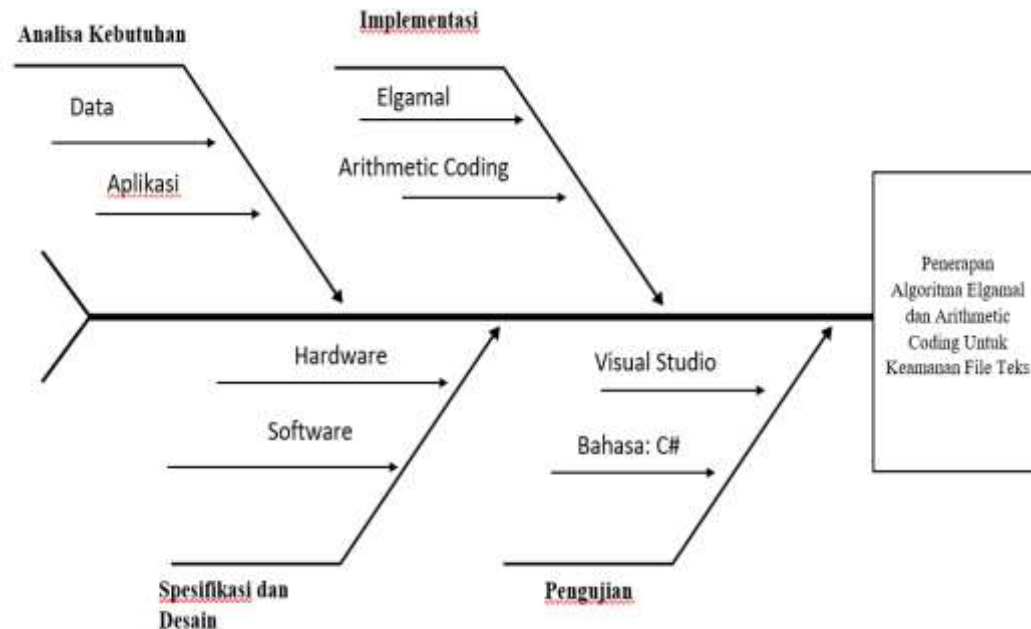
Berdasarkan data yang dipublikasikan oleh Ratiya Tari, Surya Darma Nasution, dkk. Penelitian "Penerapan Algoritma Pengkodean Aritmatika untuk Mengompresi Catatan Basis Data pada Aplikasi Ensiklopedia Botani Berbasis Android" Berdasarkan contoh kompresi dalam penelitian ini, penerapan algoritma pengkodean aritmatika untuk mengompresi entri database untuk aplikasi ensiklopedia botani menghasilkan 72 bit setelah kompresi, rasio kompresi 2,2, rasio kompresi 45%, dan redundansi 55% [10].

Penelitian yang dilakukan oleh M.A. Budiman dkk dengan judul "A Crypto Compression System Using ElGamal Public Key Encryption Algorithm and Even-Rodeh Codes". Teknik yang dikembangkan dalam penelitian ini mampu mengompresi data dengan rasio kompresi 2,11% dan menghemat ruang penyimpanan sebesar 52,25% [11].

2. METODE

A. Metode Penelitian

Metode penelitian adalah prosedur sistematis yang digunakan peneliti untuk mengumpulkan, menganalisis, dan menafsirkan data untuk menjawab pertanyaan atau menguji hipotesis tertentu[12]. Dalam penelitian ini menggunakan diagram fishbone untuk menggambarkan langkah-langkah penelitian:



Gambar 1. Diagram Fishbone

Keterangan:

1. Analisis kebutuhan

Siklus pengembangan perangkat lunak dimulai dengan tahap analisis kebutuhan perangkat lunak, yang mengumpulkan kebutuhan untuk setiap komponen sistem yang akan dibangun. Berikut beberapa contoh analisis kebutuhan untuk perancangan sistem yang akan dibuat:

- Informasi atau data langkah-langkah pembuatan aplikasi inilah yang akan dibuat.
- Aplikasi dibuat dengan *Arithmetic Coding* dan Algoritma Elgamal.

2. Spesifikasi dan desain

Untuk membangun aplikasi yang akan dirancang, spesifikasi kebutuhan perangkat lunak adalah dokumen yang menjelaskan secara menyeluruh kemampuan perangkat lunak tanpa menjelaskan bagaimana hal tersebut dilakukan. Berikut spesifikasi yang dibutuhkan untuk mengembangkan aplikasi yang akan dibuat:

a. Detail Perangkat Keras

Persyaratan perangkat keras yang diperlukan adalah sebagai berikut:

- *Processor :Intel® core™ i5-1035Gi*
- *Memory DDR4 8 GByte*
- *Harddisk 500 Gbyte*

b. Detail Perangkat Lunak

Persyaratan perangkat lunak yang diperlukan adalah:

- Sistem operasi Windows 11
- Microsoft Visual Studio 2022

3. Implentasi

Tahap implementasi adalah tahap di mana sistem diimplementasikan untuk dioperasikan, fase ini meliputi implementasi perangkat lunak, implementasi perangkat keras, implementasi database, implementasi instalasi program, penggunaan program, dan implementasi antarmuka[13]. Verifikasi program digunakan untuk memastikan bahwa program itu benar. Metode ini memberikan jaminan kebenaran berdasarkan komputasi matematis untuk mencegah kesalahan. Tentu saja pendekatan ini berbeda dengan pengujian, yang berarti perangkat lunak menggunakan data sebagai masukan untuk mengidentifikasi kesalahan dan kebenaran.

4. Pengujian

Tahap ini berfungsi untuk menemukan kesalahan. Setelah proses pengkodean selesai, sistem ini akan diuji untuk memastikan berfungsi sebagaimana mestinya, menghasilkan data yang tepat dan akurat, dan beroperasi dengan benar.

B. Pengumpulan Data

Dalam metode tentunya memerlukan suatu teknik pengumpulan informasi, dalam proses pengumpulan informasi terdapat beberapa strategi, antara lain:

1. Studi literatur
kegiatan penelitian yang dilakukan dengan menggunakan teknik pengumpulan informasi dari hasil penelitian yang dilakukan sebelumnya, yang berkaitan dengan kriptografi, algoritma elgamal, kompresi data, dan arithmetic coding[14].
2. Buku
Mereview beberapa buku yang berkaitan dengan algoritma elgamal dan arithmetic coding.
3. Dokumentasi
Dokumentasi adalah proses pencatatan informasi, data, atau fakta secara sistematis dengan tujuan melestarikan, mendistribusikan, dan memfasilitasi akses terhadap informasi di masa mendatang[15].

3. HASIL DAN PEMBAHASAN

A. Pembahasan

Studi kasus:

Untuk studi kasus mengambil sampel yang digunakan untuk enkripsi dan dekripsi yaitu "SAMUEL" perlu diingat sampel berjenis huruf kapital karena dalam pengubahan ke tabel ascii ini akan sangat memengaruhi hasil enkripsi. Berikut pemaparan dari sampel:

1. Enkripsi Sampel

Pesan "SAMUEL" kode ascii = "83,65,77,85,69,76". $K=3$,

- $S=83$

$$a = g^k \bmod p$$

$$a = 2^3 \bmod 131 = 8$$

$$b = y^k \bmod p$$

$$b = 32^3 \bmod 131 = 53$$

- $A=65$

$$a = g^k \bmod p$$

$$a = 2^3 \bmod 131 = 8$$

$$b = y^k \bmod p$$

$$b = 32^3 \bmod 131 = 122$$

proses terus dilakukan sehingga *Cipherteks* yang dihasilkan adalah (8,53),(8,122),(8,76),(8,89),(8,63),(8,58).

2. Dekripsi Sampel

Proses enkripsi dari algoritma elgamal terdiri dari 2 yaitu:

- Pakai privat key x untuk mengkalkulasikan $(ax)-1 = ap-1-x \bmod p$
- Tentukan plainteks m dengan rumus:

$$m = b/ax \bmod p = b(ax)-1 \bmod p$$

- $C = (8,53)$

$$\begin{aligned} (ax)-1 &= ap-1-x \bmod p \\ &= 8131-1-5 \bmod 131 \\ &= 8125 \bmod 131 \\ &= (85 \bmod 131)25 \bmod 131 \\ &= 1825 \bmod 131 \\ &= (185 \bmod 131)5 \bmod 131 \\ &= 245 \bmod 131 = 51 \end{aligned}$$

$$\begin{aligned} m &= b(ax)-1 \bmod p \\ &= 53.51 \bmod 131 = 83 = S \end{aligned}$$

- $C = (8,122)$

$$\begin{aligned} (ax)-1 &= ap-1-x \bmod p \\ &= 8131-1-5 \bmod 131 \\ &= 8125 \bmod 131 \\ &= (85 \bmod 131)25 \bmod 131 \\ &= 1825 \bmod 131 \\ &= (185 \bmod 131)5 \bmod 131 \\ &= 245 \bmod 131 = 51 \end{aligned}$$

$$\begin{aligned} m &= b(ax)-1 \bmod p \\ &= 122.51 \bmod 131 = 65 = A \end{aligned}$$

Proses terus dilakukan sehingga hasil dekripsi yang didapat kan yaitu "SAMUEL". Hasil berubah ke bentuk teks awal maka dari itu dapat disimpulkan proses berhasil.

3. Kompresi

Sampel yang digunakan pada kasus ini yaitu "MANURUNG". Dengan jumlah sampel yang diambil berjumlah 8 karakter maka pada probabilitas setiap karakter yang muncul dibandingkan dengan 8 karakter pada sampel. Hasil probabilitas dapat dilihat pada tabel 1 dibawah ini.

Tabel 1 Probabilitas simbol

Karakter	Probabilitas	
A	1/8	0.125
G	1/8	0.125
M	1/8	0.125
N	2/8	0.25
R	1/8	0.125
U	2/8	0.25

Setelah mengetahui peluang setiap huruf. Setiap simbol/karakter diurutkan

berdasarkan urutan alphabet dan diberi rentang tertentu yaitu nilai low range dan high range, dan nilainya antara 0 dan 1 bergantung pada probabilitasnya. Low range awal diatur 0 dan high range didapatkan dari pembagian nilai probabilitas pada karakter. Lalu high range yang didapat di letakan pada low range untuk karakter selanjutnya dengan penjumlahan nilai setiap karakter pada tabel probabilitas diatas, sampai karakter terakhir, hasil dapat dilihat pada tabel 2 berikut.

Tabel 2 Range simbol

Karakter	Low Range	High Range
A	0	0.125
G	0.125	0.25
M	0.25	0.375
N	0.375	0.625
R	0.625	0.75
U	0.75	1

Selanjutnya, prosedur encoding dijalankan menggunakan algoritma berikut:

- Tetapkan high = 1,0; low = 0,0
- Ambil simbol masukan selagi simbol masukan masih ada.
- $CR = High - Low$
- $High = low + CR * high\ range$
- $Low = low + CR * low\ range$
- Cetak Low

Hasil dari prosedur Arithmetic Coding adalah 'Low'. Pada langkah pertama, low range dan high range disetel ke 0 dan 1, dan tujuannya adalah mencari nilai karakter berikutnya. Untuk kata "MANURUNG" diatas kita menggunakan huruf "M" terlebih dahulu. Maka kita mendapatkan nilai .

$$high = 0.00 + CR * 0.375 = 0.375$$

$$low = 0.00 + CR * 0.25 = 0.25$$

selanjutnya pilih karakter 'E'. Nilai CR adalah $0.375 - 0.25 = 0.125$.

$$High_range(A) = 0.125,$$

$$Low_range(A) = 0.$$

Maka diperoleh nilai $high = 0.25 + CR * 0.125 = 0.265625$ $low = 0.25 + CR * 0 = 0.25$ Begitu seterusnya yang dirinci pada tabel 3 berikut ini.

Tabel 3 Hasil Kompresi

Karakter	Low Range	High Range	Code Range
	0	1	1
M	0.25	0.375	0.125
A	0.25	0.265625	0.015625
N	0.255859375	0.259765625	0.003906
U	0.258789063	0.259765625	0.000977
R	0.259399414	0.259521484	0.000122
U	0.259490967	0.259521484	3.05E-05
N	0.259502411	0.25951004	7.63E-06
G	0.259503365	0.259504318	9.54E-07

Data yang telah didapatkan nantinya akan diolah pada sistem untuk dilakukannya proses kompresi.

4. Dekompresi

Untuk melakukan decoding dipakai algoritma berikut:

- Ambil encoded-symbol (ES)
- Do
- Cari range dari simbol yang melingkupi ES
- Cetak simbol
- $CR = high_range - low_range$
- $ES = ES - low_range$
- $ES = ES / CR$
- Until simbol habis

Prosedur decoding untuk pesan yang dikodekan sebelumnya ($ES = 0.259503365$) adalah sebagai berikut. Simbol/karakter 'M' ditemukan sebagai rangkaian simbol yang mencakup ES.

$Low_range = 0.25$ $High_range = 0.375$

$CR = 0.375 - 0.25 = 0.125$

$ES = 0.259503365 - low_range$

$ES = 0.009503365$

$ES = 0.009503365 / CR$

$ES = 0.076026917$

Hasil dari semua proses encode dapat dilihat pada tabel 4 berikut:

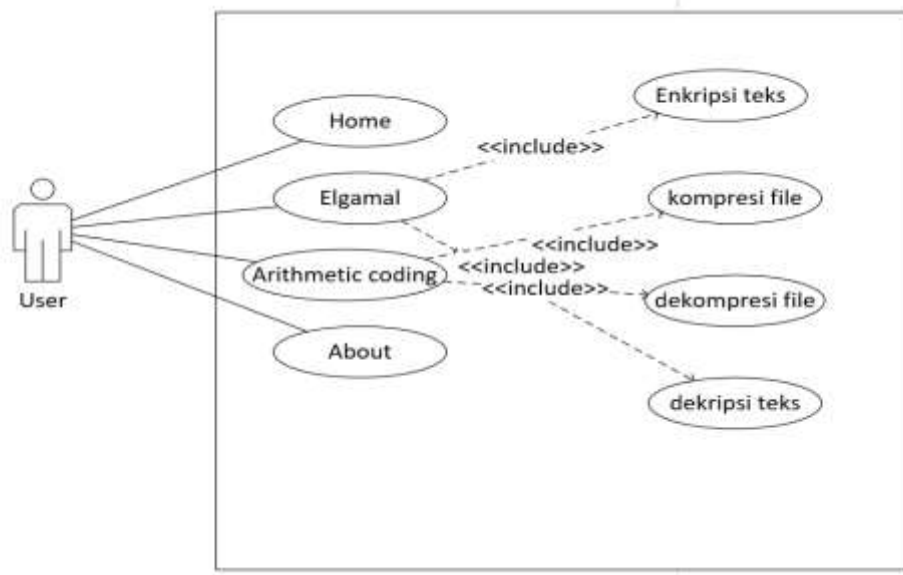
Tabel 4 Dekompresi

Es	Karakter	Low Range	High Range	Code Range
0.076026917	M	0.25	0.375	0.125
0.608215332	A	0	0.125	0.125
0.932861328	N	0.375	0.625	0.25
0.731445313	U	0.75	1	0.25
0.8515625	R	0.625	0.75	0.125
0.40625	U	0.75	1	0.25
0.125	N	0.375	0.625	0.25
0	G	0.125	0.25	0.125

Sama halnya dengan proses kompresi data dari dekompresi akan diolah pada sistem untuk mendapatkan nilai dekompresi.

Usecase Diagram

Untuk membantu proses analisa bagian-bagian penyusun sistem agar dapat dipahami oleh pembaca, penulis membuat diagram use-case. Gambar 2 menunjukkan usecase diagram sebagai berikut:



Gambar 2. Usecase Diagram Sistem

B. Hasil

1. Halaman Home

Home pada sistem adalah halaman ketika dijalankan akan menampilkan judul penelitian, nama penulis, nim, logo, program studi, fakultas, nama universitas, dan tahun penelitian. Gambar 3 berikut menunjukkan halaman beranda.

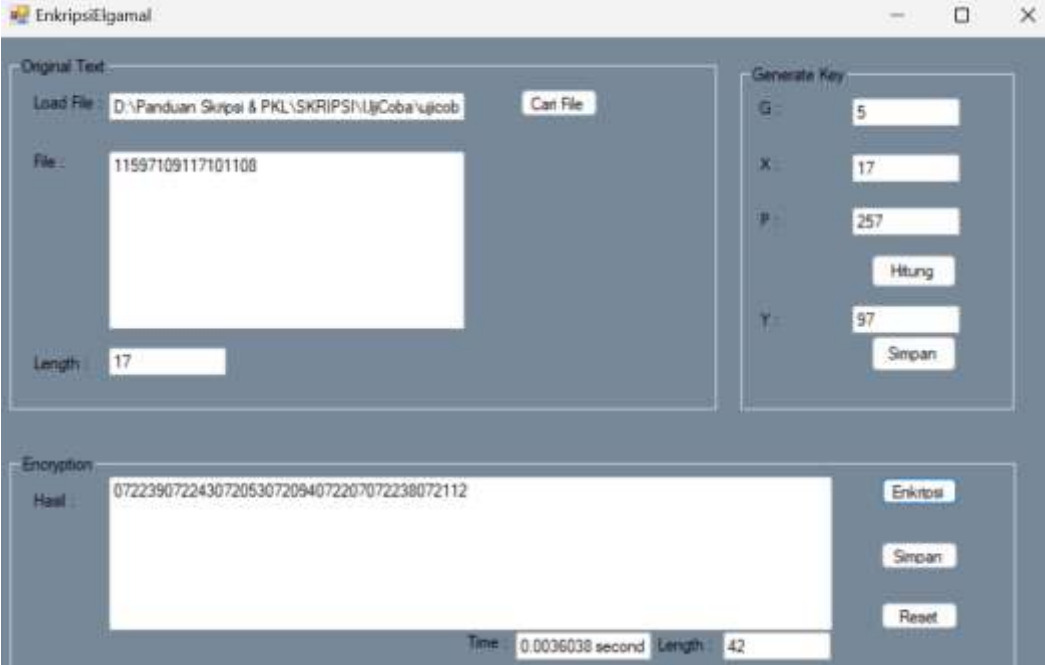


Gambar 3. Tampilan Halaman Home

2. Halaman Enkripsi

Halaman enkripsi terdapat 3 bagian utama yaitu original text yang dimana fungsi dari bagian ini adalah menginput file asli kedalam sistem untuk dilakukan proses enkripsi dan dapat menampilkan jumlah karakter yang ada pada file, selanjutnya ada generate key fungsinya yaitu membuat kunci privat dan public dengan cara menginput nilai G,X,dan P. Terakhir ada bagian encryption fungsi ini bertujuan untuk

menghasilkan chiperteks ketika semua komponen yang dibutuhkan telah diinputkan, pada bagian ini ada time dan length untuk melihat waktu yang diperlukan dan jumlah karakter yang didapatkan tentunya ini akan menjadi indikator untuk menganalisis sistem. Gambar 4 menunjukkan halaman enkripsi:

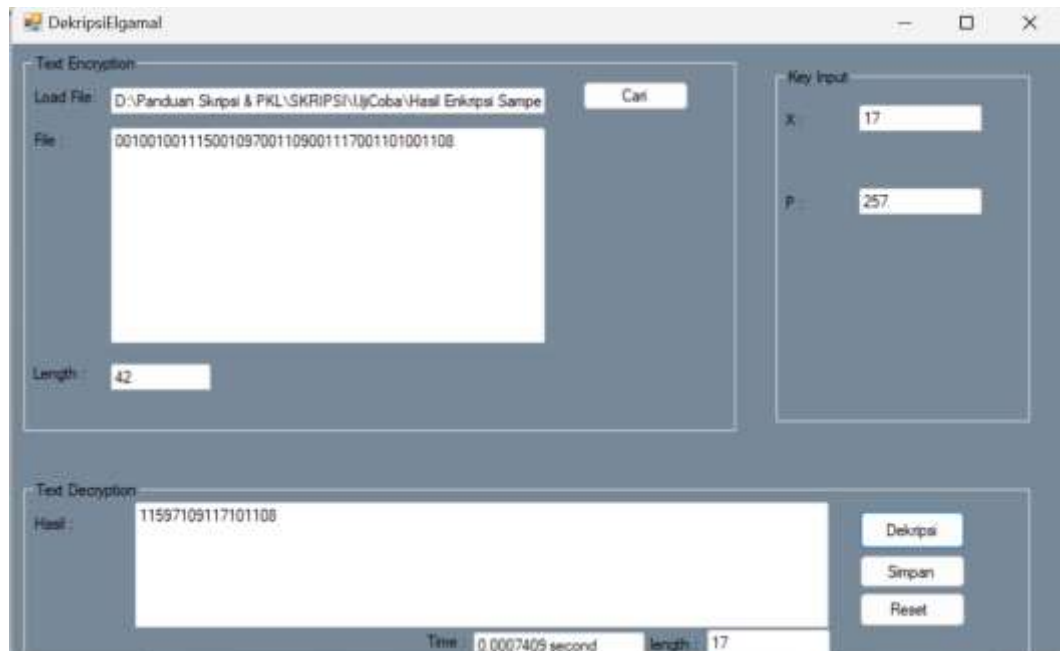


The screenshot displays the 'EnkripsiElGamal' application window. It is divided into three main sections: 'Original Text', 'Generate Key', and 'Encryption'.
1. 'Original Text' section: Includes a 'Load File' button with a file path 'D:\Panduan Skripsi & PKL\SKRIPSI\UjiCoba\ujicob', a 'Car File' button, a 'File' input field containing '11597109117101108', and a 'Length' input field set to '17'.
2. 'Generate Key' section: Features input fields for 'G' (5), 'X' (17), 'P' (257), and 'Y' (97). It includes 'Hitung' and 'Simpan' buttons.
3. 'Encryption' section: Shows a 'Hasil' output field with the encrypted text '072239072243072053072094072207072238072112'. It has 'Enkripsi', 'Simpan', and 'Reset' buttons. At the bottom, it displays 'Time: 0.0036038 second' and 'Length: 42'.

Gambar 4. Tampilan Halaman Enkripsi

3. Halaman Dekripsi

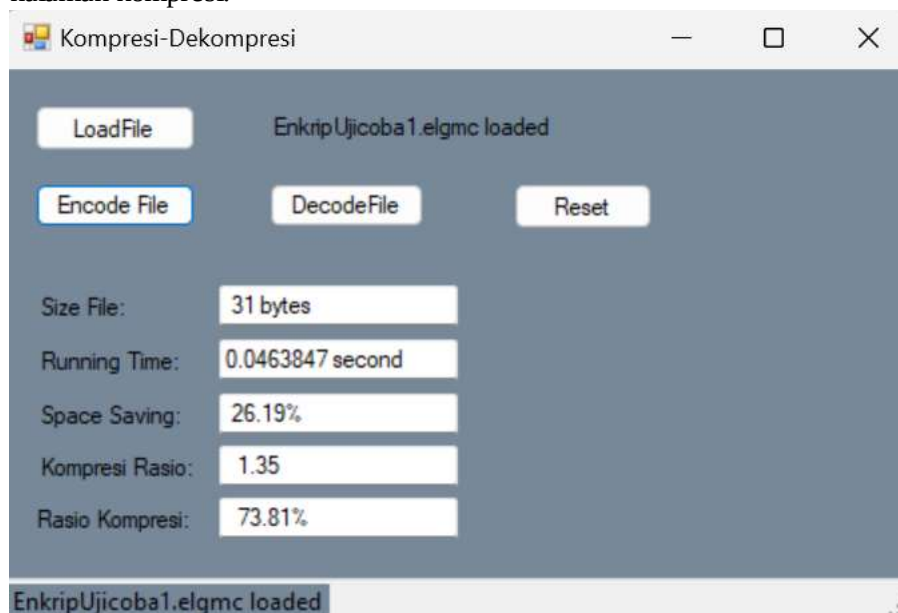
Pada halaman dekripsi tidak berbeda jauh dengan halaman enkripsi masih memiliki 3 bagian utama yang membedakan pada bagian key input pada halaman dekripsi yang diinputkan adalah kunci privat, teks yang didekripsi adalah teks enripsi, gambar 5 menunjukkan halaman dekripsi:



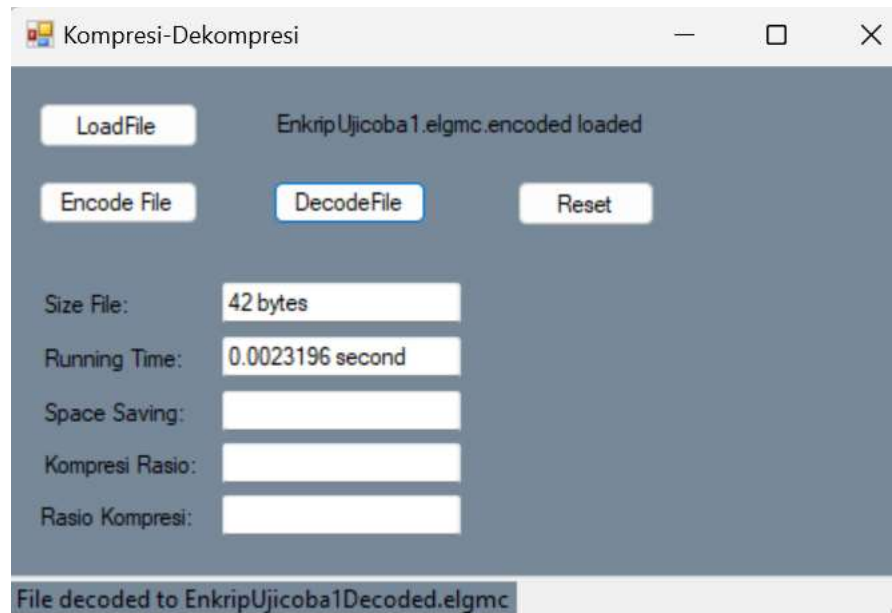
Gambar 5. Tampilan Halaman Dekripsi

4. Halaman Kompresi dan Dekompresi

Halaman kompresi bertujuan mengompres dan mengdekompres file, kedua teknik ini dibuat dalam satu halaman dengan 3 indikator yaitu size file, running time, dan space saving. Tujuan indikator dibuat untuk membandingkan sampel dari beberapa file sehingga mempermudah analisis kedepannya. Gambar 6 dibawah menunjukkan halaman kompresi:



Gambar 6. Tampilan Halaman Kompresi



Gambar 7. Tampilan Halaman Dekompresi

5. Halaman About

Halaman About adalah halaman yang menampilkan identitas penulis yang terdapat pada sistem, identitas penulis pada halaman about ini terdiri dari: Judul penelitian, foto penulis, nama, nim, kelas, tempat/tanggal lahir, Alamat, email.



Gambar 8. Tampilan Halaman About

6. Hasil Pengujian

Pada penelitian ini untuk mendapatkan hasil dari pengujian dibutuhkan 10 sampel yang setiap sampel diuji satu persatu, Berikut hasil pengujian yang dipisah kedalam 4 tabel.

Tabel 5 Hasil Uji Coba Enkripsi Elgamal

	Original Teks (byte)	Perubahan ke ascii (byte)	Chiperteks (byte)	Time(second)
Ujicoba1.txt	6	17	42	0.002
Ujicoba2.txt	39	95	240	0.003
Ujicoba3.txt	92	226	558	0.006
Ujicoba4.txt	124	305	750	0.007
Ujicoba5.txt	167	421	1008	0.013
Ujicoba6.txt	226	537	1362	0.038
Ujicoba7.txt	286	733	1722	0.013
Ujicoba8.txt	374	965	2250	0.027
Ujicoba9.txt	412	1062	2478	0.024
Ujicoba10.txt	458	1183	2754	0.053

Tabel 5 menunjukkan bahwa ukuran file terenkripsi jauh lebih besar dibandingkan dengan plainteks sebelum enkripsi. Hal ini berarti bahwa file chiperteks memerlukan ruang penyimpanan yang lebih besar. Selain itu, dapat dilihat bahwa waktu yang dibutuhkan untuk enkripsi lebih lama semakin besar ukuran file, meskipun pada ujicoba ke-7 dan ke-9 terjadi penurunan waktu.

Tabel 6 Hasil Ujicoba Dekripsi Elgamal

	Chiperteks (byte)	Plainteks (Byte)	Time (Second)
Ujicoba1.txt	42	17	0.001
Ujicoba2.txt	240	95	0.001
Ujicoba3.txt	558	226	0.001
Ujicoba4.txt	750	305	0.002
Ujicoba5.txt	1008	421	0.002
Ujicoba6.txt	1362	537	0.004
Ujicoba7.txt	1722	733	0.004
Ujicoba8.txt	2250	965	0.008
Ujicoba9.txt	2478	1062	0.007

Ujicoba10.txt	2754	1183	0.034
---------------	------	------	-------

Pada Tabel 6 terlihat bahwa ukuran file hasil dekripsi sesuai dengan ukuran file asli. Hal ini menunjukkan bahwa proses dekripsi dapat mengembalikan isi chiperteks ke bentuk semula. Selain itu, meskipun ukuran file lebih lama dalam proses dekripsi, waktu yang diperlukan untuk dekripsi tidak lebih lama daripada enkripsi.

Tabel 7 Hasil Ujicoba Kompresi Arithmetic Coding

	Chiperteks (byte)	Size File (byte)	Runing Time (second)	Space Saving (%)	Ratio of Compression (%)	Compression Ratio
Ujicoba1.txt	42	31	0.001	26,19	73,81	1,35
Ujicoba2.txt	240	145	0.003	39,58	60,42	1,66
Ujicoba3.txt	558	269	0.004	52,79	48,21	2.07
Ujicoba4.txt	750	335	0.007	55,33	44,67	2.24
Ujicoba5.txt	1008	460	0.007	54,37	45,63	2.19
Ujicoba6.txt	1362	616	0.026	54,77	45,23	2,21
Ujicoba7.txt	1722	736	0.012	57,26	42,74	2,34
Ujicoba8.txt	2250	970	0.031	56,89	43,11	2,32
Ujicoba9.txt	2478	973	0.018	60,73	39,27	2,55
Ujicoba10.txt	2754	1170	0.034	56,21	43,79	2,28
Rata-Rata			0.014	51,41	48,68	2,12

Pada data dari Tabel 7 terlihat bahwa waktu yang diperlukan rata-rata 0,014 second dan ukuran kompresi hanya separuh dari ukuran chiperteks, hal ini juga diperkuat oleh penghematan ruang penyimpanan yang mencapai rata-rata 51,41%. Ini menunjukkan bahwa sistem kompresi yang menggunakan algoritma arithmetic coding dapat mengurangi ukuran file hampir setengah dari ukuran aslinya, dan efisiensi dari sistem ini terbilang baik karena rasio kompresinya memiliki rata-rata 48,68% dan kompresi rasio rata-ratanya adalah 2,12.

Tabel 8 Hasil Ujicoba Dekompresi Arithmetic Coding

	Ukuran <i>file</i> kompresi (byte)	Ukuran <i>file</i> dekompresi (byte)	Runing Time (second)
Ujicoba1.txt	31	42	0.004

Ujicoba2.txt	145	240	0.003
Ujicoba3.txt	269	558	0.006
Ujicoba4.txt	335	750	0.007
Ujicoba5.txt	460	1kb	0.026
Ujicoba6.txt	616	1.33kb	0.013
Ujicoba7.txt	736	1.68kb	0.033
Ujicoba8.txt	970	2.20kb	0.039
Ujicoba9.txt	973	2.42kb	0.038
Ujicoba10.txt	1kb	2.69kb	0.051

Pada Tabel 8 dapat diamati bahwa sistem mampu mengembalikan file yang telah dikompresi ke keadaan semula. Terkait dengan waktu proses dekompresi, terlihat bahwa waktu yang diperlukan cenderung meningkat, meskipun terjadi penurunan pada ujicoba 2 dan 6. Hal ini menunjukkan bahwa waktu yang dibutuhkan untuk dekompresi berkorelasi positif dengan ukuran hasil.

4. KESIMPULAN

Kesimpulan yang dapat diambil dari pengimplementasian algoritma elgamal dan arithmetic coding pada file teks antara lain:

1. Sistem mampu mengamankan isi file dengan menggunakan algoritma Elgamal.
2. Sistem mampu memperkecil ukuran file tanpa menghilangkan isi file karena sistem dapat mengembalikannya ke bentuk semula.
3. Hasil kompresi menggunakan algoritma arithmetic coding berdasarkan parameter yang digunakan menunjukkan bahwa nilai rata-rata Compression Ratio (CR) adalah 2.12, Ratio of Compression (RC) adalah 48.68%, dan Space Saving (SS) adalah 51.41%.
4. File hasil enkripsi berektensi .elgmc dan hasil file kompresi berektensi .encode

Sistem ini hanya diuji dengan kunci 256 bit dengan interval 32 bit, kunci yang lebih panjang dapat digunakan untuk pengembangan berikutnya. Sebaliknya, meningkatkan keamanan hanya dapat dicapai dengan menghasilkan p yang sangat besar, atau kunci prima. Diharapkan untuk penelitian selanjutnya dapat dilakukan dengan berbagai jenis file lainnya misal image, audio, video, dan dapat diimplementasikan kedalam android maupun web.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Universitas Potensi Utama dan dosen pembimbing atas berbagai masukan dan saran yang berharga dalam penyelesaian penelitian ini.

REFERENSI

- [1] M. D. Irawan, "Implementasi Kriptografi Vigenere Cipher Dengan Php," *J. Teknol. Inf.*, vol. 1, no. 1, p. 11, 2017, doi: 10.36294/jurti.v1i1.21.
- [2] G. Geta Putri, W. Styorini, and R. Dian Rahayani, "Ethos (Jurnal Penelitian dan Pengabdian Masyarakat): 197-207 Analisis Kriptografi Simetris AES dan Kriptografi Asimetris RSA Pada Enkripsi Citra Digital," *Ethos (Jurnal Penelit. dan Pengabd. Masyarakat)*, vol. 3, no. 8, pp. 197–207, 2015.
- [3] A. Karima and A. Saputro, "Pembangkitan Kunci pada Algoritma Asimetris ElGamal untuk Meningkatkan Keamanan Data bertipe . docx Key Generation on ElGamal Asymmetric Algorithm To Enhance . docx Format Data Security," *Sisfotenika*, vol. 6, no. 2, pp. 170–181, 2016, [Online]. Available: <https://www.mendeley.com/catalogue/pembangkitan-kunci-pada-algoritma-asimetris-elgamal-untuk-meningkatkan-keamanan-data-bertipe-docx/>.
- [4] S. R. Saragih and D. P. Utomo, "Penarapan Algoritma Prefix Code Dalam Kompresi Data Teks," *KOMIK (Konferensi Nas. ...)*, vol. 4, no. 1, pp. 249–252, 2020, doi: 10.30865/komik.v4i1.2691.
- [5] D. Rachmawati, M. A. Budiman, and M. A. Subada, "Comparison study of Fibonacci code algorithm and Even-Rodeh algorithm for data compression," *J. Phys. Conf. Ser.*, vol. 1321, no. 3, 2019, doi: 10.1088/1742-6596/1321/3/032015.
- [6] R. Ariska, "Penerapan Algoritma Arithmetic Coding Pada Aplikasi Kamus Teknologi Informasi Berbasis Android," *TIN Terap. Inform. Nusant.*, vol. 2, no. 7, pp. 407–413, 2021, [Online]. Available: <http://ejurnal.seminar-id.com/index.php/tin/article/view/948%0Ahttps://ejurnal.seminar-id.com/index.php/tin/article/download/948/682>.
- [7] C. Desri and N. Situmorang, "Penerapan Algoritma Arithmetic Coding Pada Kompresi File Video Streaming Berbasis Android," vol. 7, pp. 124–132, 2024, doi: 10.30865/komik.v6i1.8046.
- [8] F. Husaini, A. M. H. Pardede, and I. Gultom, "Penerapan Enkripsi Menggunakan Metode Elgamal guna Meningkatkan Keamanan Data Text dan Gambar," *JUKI J. Komput. dan Inform.*, vol. 4, no. 1, pp. 67–73, 2022.
- [9] L. De Feo, B. Poettering, and A. Sorniotti, "On the (In)Security of ElGamal in OpenPGP," *Commun. ACM*, vol. 66, no. 6, pp. 107–115, 2023, doi: 10.1145/3592835.
- [10] R. Tari, S. D. Nasution, and T. Zebua, "Penerapan Algoritma Arithmetic Coding Untuk Mengkompresi Record Database Pada Aplikasi Ensiklopedia Flora Berbasis Android," *TIN Terap. Inform. Nusant.*, vol. 2, no. 7, pp. 433–444, 2021, [Online]. Available: <http://ejurnal.seminar-id.com/index.php/tin/article/view/950>.
- [11] M. A. Budiman, E. M. Zamzami, and C. L. Ginting, "A Crypto Compression System Using ElGamal Public Key Encryption Algorithm and Even-Rodeh Codes," *J. Phys. Conf. Ser.*, vol. 1566, no. 1, pp. 0–5, 2020, doi: 10.1088/1742-6596/1566/1/012071.
- [12] Hardani *et al.*, *Metode Penelitian Kualitatif dan Kuantitatif*, no. Maret. 2020.
- [13] R. Sanjaya and S. Hesinto, "Rancang Bangun Website Profil Hotel Agung Prabumulih Menggunakan Framework Bootstrap," *J. Teknol. dan Inf.*, vol. 7, no. 2, pp. 57–64, 2018, doi: 10.34010/jati.v7i2.758.
- [14] M. Ovartadara, Firman, and Desyandri, "Penerapan Model Project Based Learning Dalam Meningkatkan Komunikasi Matematis Siswa Sekolah Dasar," *Didakt. J. Ilm. PGSD STKIP Subang*, vol. 8, no. 2, pp. 2667–2678, 2023, doi: 10.36989/didaktik.v8i2.579.

- [15] W. S. Fatahillah, "Manajemen Arsip Digital Berbasis SIMAK dalam Peningkatan Layanan Administrasi di Madrasah Pembangunan UIN Jakarta," 2019.