

Paper

Kombinasi Kriptografi Algoritma Polyalphabetic Dan Kompresi Huffman Untuk Pengamanan Data

Author: Aftaul Musla, Tommy, Andi Marwan Elhanafi



SEMINAR NASIONAL TEKNOLOGI INFORMASI & KOMUNIKASI
SNASTIKOM KE - 8 TAHUN 2021

Tema : Menyukseskan Transformasi Digital Menuju Indonesia Maju



Kombinasi Kriptografi Algoritma Polyalphabetic Dan Kompresi Huffman Untuk Pengamanan Data

Aftaul Musla¹, Tommy², Andi Marwan Elhanafi³

^{1,2,3} Universitas Harapan, Medan, Indonesia

¹aftaulmusla45@gmail.com, ²tomshirakawa@gmail.com, ³andimarwanelhanafi@gmail.com

Abstrak- Semakin majunya dunia teknologi menghasilkan aplikasi baru yang bisa membantu memudahkan manusia dalam proses pengiriman file dan karena kemudahan aplikasi tersebut diakses oleh manusia, maka tidak menutup kemungkinan terjadinya kecurangan atau kebocoran file dari pihak yang tidak berwenang. Maka dibutuhkan sistem keamanan data yang semakin kompleks dan ukuran data juga menjadi salah satu faktor untuk mempercepat proses pengirimannya. Penggabungan teknik kriptografi dan kompresi merupakan salah satu solusi yang dapat dilakukan untuk mengatasi masalah tersebut. Kriptografi merupakan keahlian dan ilmu dari cara-cara untuk komunikasi aman pada kehadiran pihak yang tidak berwenang. Kompresi adalah proses untuk memperkecil ukuran data sehingga dapat menyimpan data lebih banyak dan mempercepat proses transmisi data. Algoritma yang digunakan pada penelitian ini adalah Polyalphabetic dan Huffman. Implementasi sistem ini dibangun dalam bentuk website. Hasil penelitian menunjukkan bahwa pengamanan dan kompresi file menggunakan Polyalphabetic dan Huffman dapat mengembalikan isi file semula secara utuh. Pengujian algoritma Huffman menunjukkan bahwa ukuran file hasil kompresi pada beberapa sampel percobaan dengan ukuran berbeda berbanding lurus dengan ukuran file awal sebelum dikompresi.

Kata Kunci: *Polyalphabetic, Huffman, Website*

Abstract- The more advanced the world of technology produces new applications that can help make it easier for humans in the process of sending files and because of the ease with which these applications are accessed by humans, it does not rule out the possibility of fraud or file leaks from unauthorized parties. is also one of the factors to speed up the delivery process. Combining cryptography and compression techniques is one solution that can be done to overcome these problems. Cryptography is the expertise and science of ways for secure communication in the presence of unauthorized parties. Compression is a process to reduce the size of the data so that it can store more data and speed up the data transmission process. The algorithms used in this research are Polyalphabetic and Huffman. Implementation of this system is built in the form of a website. The results showed that securing and compressing files using Polyalphabetic and Huffman could restore the original file contents intact. Testing the Huffman algorithm shows that the compressed file size in several experimental samples with different sizes is directly proportional to the initial file size before being compressed.

Keywords: *Polyalphabetic, Huffman, Website*

1. PENDAHULUAN

Semakin majunya dunia teknologi maka semakin menghasilkan aplikasi baru yang bisa membantu memudahkan pengguna teknologi dalam proses pengiriman *file* dan karena kemudahan aplikasi tersebut diakses oleh manusia, maka tidak menutup kemungkinan terjadinya kecurangan atau kebocoran *file* dari pihak yang tidak berwenang maka dibutuhkan sistem keamanan data yang semakin kompleks. Keamanan data merupakan salah satu hal penting dari sistem informasi. Oleh karena itu, pengguna teknologi semakin mengembangkan sistem keamanan data/informasi yang biasa disebut dengan kriptografi. Kriptografi adalah disiplin ilmu yang mempelajari tata cara bagaimana menjaga agar data atau informasi (pesan) tetap aman saat dikirim dari pengirim ke penerima tanpa gangguan dari pihak lain[1].

Teknik kriptografi telah dikembangkan sedemikian rupa untuk menangani masalah keamanan ini, dan salah satu teknik untuk melindungi data dari sistem informasi adalah dengan menggunakan algoritma enkripsi. Dengan menggunakan teknik ini, data asli (*plaintext*) dapat diubah menjadi *ciphertext* dengan menggunakan kunci tertentu. Untuk itu, kerahasiaan kunci sangat diperlukan demi keberhasilan proses keamanan data.

Salah satu metode kriptografi yang sudah ada sejak lama adalah algoritma kriptografi klasik. Algoritma kriptografi klasik merupakan metode kriptografi yang berbasis karakter, bukan *bit* seperti pada algoritma *modern*. Salah satu kriptografi klasik yang lazim dikenal adalah *Polyalphabetic substitution ciphers*.

Kriptografi *Polyalphabetic* merupakan improvisasi dari teknik substitusi *Monoalphabetic*. *Polyalphabetic substitution cipher* melakukan teknik kriptografi *monoalphabetic* satu kali proses untuk suatu *plaintext* atau pesan. Algoritma penyandian *Polyalphabetic cipher* membuat *ciphertext* lebih kuat untuk dapat dipecahkan untuk dipahami karena karakter *plaintext* yang sama tidak dienkripsi dengan *ciphertext* yang sama. Kunci (*key*) untuk penyandian *Polyalphabetic Cipher* dapat berbeda sehingga mempunyai kemungkinan kombinasi yang lebih bervariasi dan lebih sulit dipecahkan. Metode *Vigenere Cipher* salah satu yang mendasari dari *Polyalphabetic substitution cipher*. Teknik penyandian *Vigenere Cipher* menggunakan tabel yang dikenal dengan tabel *Vigenere* yang digunakan dan menjadi acuan di beberapa algoritma pengembangan metode *Polyalphabetic Cipher* [2].

Perkembangan aplikasi saat ini menghasilkan ukuran data keluaran yang lebih besar sehingga membutuhkan penyimpanan yang besar pula, oleh karena itu semakin besar ukuran data maka semakin banyak pula ruang dan waktu yang dibutuhkan. Jika ukuran data dapat dikompresi menjadi lebih kecil dari ukuran aslinya, media penyimpanan secara otomatis dapat menyimpan lebih banyak data dan mengirimkannya lebih cepat. Salah satu kompresi data yang dikenal adalah algoritma *Huffman*. Algoritma *Huffman* sendiri merupakan algoritma yang digunakan untuk mengkompresi berbagai tipe data. Proses kompresi algoritma *Huffman* memiliki tiga *fase*, yaitu *fase* pembentukan pohon *Huffman*, kedua *fase encoding* dan *fase decoding* [3].

Tujuan dilakukannya penelitian ini adalah sebagai berikut:

1. Membuat aplikasi enkripsi dan dekripsi file dengan menggunakan kriptografi algoritma *Polyalphabetic* dan metode kompresi *Huffman*.
2. Menguji performa kriptografi enkripsi dan dekripsi dengan algoritma *polyalphabetic* untuk pengamanan data *file*.
3. Menguji performa kompresi algoritma *Huffman* dalam memperkecil ukuran *file*.

Adapun manfaat pada penelitian ini adalah sebagai berikut:

1. Dapat menerapkan kombinasi kriptografi *Polyalphabetic* dan kompresi data *Huffman* untuk keamanan data.
2. Menambah pengetahuan dan wawasan dalam memahami sistem keamanan data.
3. Menambah pengetahuan tentang algoritma kriptografi *Polyalphabetic* dan kompresi data *Huffman*.

2. METODE PENELITIAN

2.1 Kriptografi

Kriptografi merupakan salah satu aspek yang berperan penting didalam bidang keamanan informasi. Kriptografi digunakan untuk menjaga keamanan pesan atau informasi, baik itu pesan yang dikirimkan melalui saluran komunikasi maupun pesan yang disimpan dalam media penyimpanan. Kriptografi bisa diartikan sebagai ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan data.[4]. Kriptografi (*Cryptography*) berasal dari bahasa Yunani "*Cryptos*" artinya *secret* (rahasia), sedangkan "*graphien*" artinya "*writing*" (tulisan)[5], jadi bisa diartikan kriptografi itu adalah tulisan rahasia. Dengan kata lain kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan informasi atau pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti maknanya oleh pihak yang tidak terkait.

2.2 Polyalphabetic Cipher

Polyalphabetic cipher melakukan teknik penyandian *Monoalphabetic* satu kali proses untuk suatu *plaintext* pesan. Teknik penyandian *Polyalphabetic Cipher* membuat *ciphertext* lebih kuat untuk dapat dipecahkan karena karakter *plaintext* yang sama tidak dienkripsi dengan *ciphertext* yang sama. *Polyalphabetic Cipher* ditemukan pertama kali oleh Leon Battista pada tahun 1568. Teknik ini digunakan sebagai pengembangan dari Teknik substitusi *Monoalphabetic*. Sesuai dengan namanya, algoritma *Polyalphabetic Cipher* menyandikan sekelompok karakter atau *string* dengan melibatkan penggunaan kunci yang tidak sama atau berbeda[2]. Metode *Vigenere Cipher* bisa disebut sebagai dasar dari *Polyalphabetic Cipher*. Beberapa ketentuan dalam metode ini antara lain: Kata kunci digunakan untuk menentukan enkripsi setiap *alphabet* dalam *plaintext*, setiap kunci dapat disubstitusi dengan bermacam-macam kunci yang lain dan dapat digunakan secara berulang, huruf ke-*i* dalam *plaintext* di spesifikasikan oleh *alphabet* yang digunakan dalam kunci [6].

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X

Gambar 1. Bujursangkar

Vigenere Cipher sangat dikenal karena mudah dipahami dan diimplementasikan. *Vigenere Cipher* menggunakan bujursangkar *Vigenere* untuk melakukan penyandian. Kolom kiri dari bujursangkar menyatakan huruf - huruf kunci, sedangkan baris paling atas menyatakan huruf - huruf *plaintext*. Setiap *Vigenere Cipher*, baris di dalam bujursangkar menyatakan huruf - huruf *ciphertext* yang diperoleh dengan yang mana jumlah pergeseran huruf *plaintext* ditentukan nilai numerik huruf kunci tersebut (yaitu, A = 0, B = 1, C =2, ..., Z = 25).

Tabel 1. Nilai Numerik Dalam *Vigenere Cipher*

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
P	Q	R	S	U	V	W	X	Y	Z					
15	16	17	18	20	21	22	23	24	25					

Secara matematis, rumus enkripsi dan dekripsi *vigenere cipher* sebagai berikut.

Enkripsi

$$C_i = (P_i + K_i) \text{ Mod } 26$$

Dekripsi

$$C_i = (P_i - K_i) \text{ Mod } 26$$

Keterangan:

C_i : Nilai *Ciphertext*

P_i : Nilai *Plaintext*

K_i : Nilai Kunci(*key*)

Mod : Modulus 26 karakter

2.3 Kompresi Data

Kompresi data (pemampatan data) merupakan suatu teknik untuk memperkecil jumlah ukuran data (hasil pemampatan) dari data aslinya. pemampatan data umumnya diterapkan pada mesin komputer, hal ini dilakukan karena setiap simbol yang dimunculkan pada komputer memiliki nilai *bit-bit* yang berbeda. Kompresi data digunakan untuk mengurangi jumlah *bit* yang dihasilkan dari setiap simbol yang muncul. Dengan kompresi ini diharapkan dapat mengurangi dalam ruang penyimpanan[7]. Berdasarkan hasil kompresi atau output yang dihasilkan, kompresi data dibedakan menjadi dua bagian yaitu kompresi *Loseless* yaitu teknik kompresi dimana data hasil kompresi dapat didekompres lagi dan hasilnya tepat sama seperti data sebelum proses kompresi. dan kompresi *Lossy* adalah teknik kompresi dimana data hasil dekompresi tidak sama dengan data sebelum kompresi namun sudah “cukup” untuk digunakan [8].

2.4 Metode Huffman

Metode *Huffman* adalah algoritma yang dikembangkan David A. Huffman pada 1952. Metode *Huffman* menggunakan prinsip pengkodean yang mirip dengan kode *morse*, yaitu setiap karakter (*symbol*) dikodekan hanya dengan rangkaian beberapa *bit*, dimana karakter yang sering muncul dikodekan dengan rangkaian *bit* yang pendek dan karakter yang jarang muncul dikodekan dengan rangkaian *bit* yang lebih panjang, karena prosesnya yang menggunakan kode ini, membuat algoritma *Huffman* sebagai algoritma keluarga dengan *variable codeword length* [9].

Kode *Huffman* adalah *string biner* yang digunakan untuk mengkodekan setiap karakter di dalam data. Ide dari kode *Huffman* adalah menggunakan representasi dengan *bit* yang sesedikit mungkin untuk karakter yang

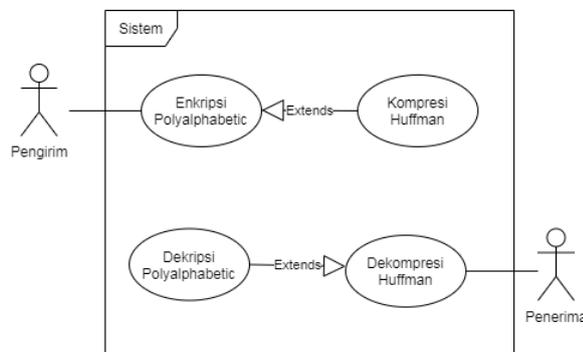
memiliki jumlah kemunculan yang tinggi, dan sebaliknya, karakter yang memiliki jumlah kemunculan lebih sedikit memiliki representasi dengan jumlah *bit* yang lebih sedikit [10]. Kode *Huffman* pada dasarnya merupakan kode prefiks (*prefix code*). Kode prefiks adalah himpunan yang berisi sekumpulan kode *biner* yang dalam hal ini tidak ada kode *biner* yang menjadi awal bagi kode *biner* yang lain. Kode prefiks biasanya direpresentasikan sebagai pohon *biner* yang berlabel, dimana setiap sisi diberi label 0 atau 1. Rangkaian *bit* yang terbentuk pada setiap lintasan dari akar ke daun merupakan kode prefiks untuk karakter yang berpadanan. Pohon *biner* ini biasa disebut pohon *Huffman*.

2.5 Perancangan Sistem

Analisis sistem adalah bagian dari proses pembangunan aplikasi dengan mengidentifikasi bagian-bagian yang dibutuhkan dalam pembangunan aplikasi. Analisis sistem dapat dilakukan dengan menentukan input, proses (pengolahan data) dan output dari sistem.

2.6 Pemodelan Sistem

Pemodelan sistem ini menggunakan diagram UML (*Unified Modelling Language*). Diagram UML yang digunakan *use case diagram*, *activity diagram* dan *flowchart*. Berikut ini adalah diagram *use case* yang dirancang sebagai pemodelan persyaratan sistem berdasarkan informasi kebutuhan sistem dan aktor yang berperan didalamnya.

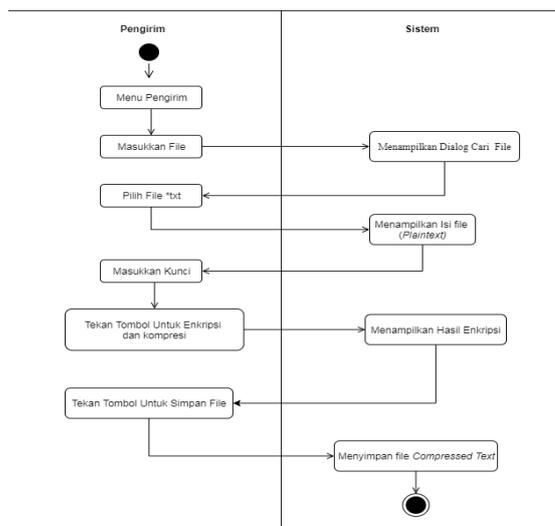


Gambar 2. Use Case Sistem

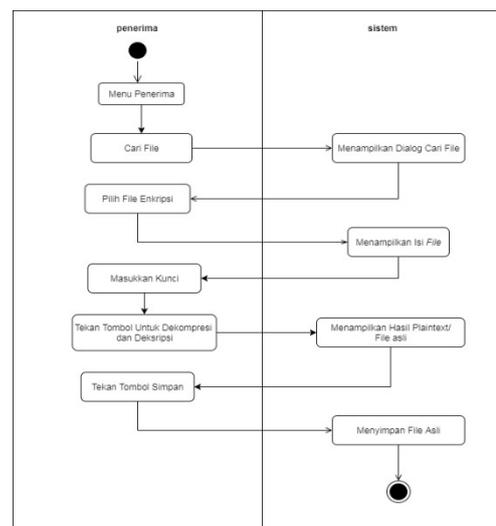
Use case sistem menjelaskan bahwa sistem dapat diakses oleh dua aktor yaitu pengirim dan penerima. Aktor pengirim yaitu enkripsi *Polyalphabetic* dan Kompresi *Huffman* dan aktor penerima yaitu dekomposisi *Huffman* dan deksripsi *Polyalphabetic*. Aktor pengirim melakukan enkripsi terlebih dahulu dengan *Polyalphabetic* dengan menggunakan sebuah kunci karakter dan setelah dienkripsi menghasilkan *ciphertext* kemudian *ciphertext* di kompresi dengan algoritma *Huffman* dan menghasilkan *compressed text*. Aktor penerima menerima *compressed text* dan kunci dari pengirim dan melakukan dekomposisi atas *file compressed text* dan menghasilkan *ciphertext* kembali. Kemudian *ciphertext* di deksripsi menggunakan kunci hingga menghasilkan *Plaintext/file* asli.

1. Activity Diagram Sistem

Berdasarkan *use case* yang telah didefinisikan sebelumnya, dihasilkanlah *Activity Diagram* aplikasi kriptografi kombinasi anatar dua metode yang dapat dilihat pada gambar 3 dan gambar 4.



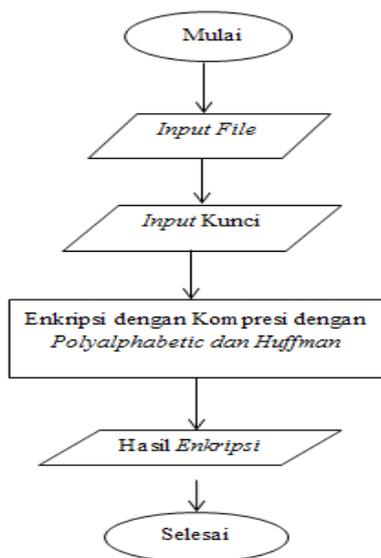
Gambar 3. Activity Diagram pengirim



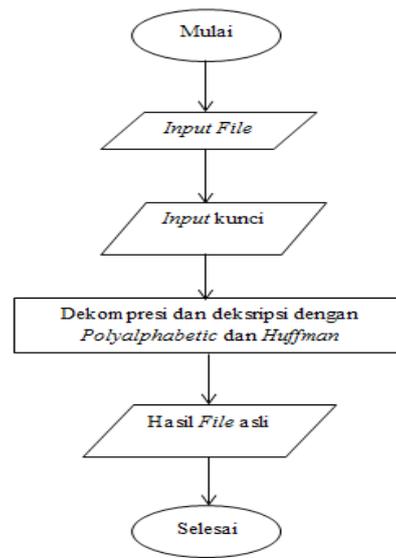
Gambar 4. Activity Diagram penerima

2. Flowchart sistem

Sistem ini dibangun dengan dua aktor utama yaitu aktor pengirim yang memiliki fungsi enkripsi dan kompresedangkan penerima memiliki dekompresi dan deksripsi.



Gambar 5. Flowchart Pengirim



Gambar 6. Flowchart Penerima

3. HASIL DAN PEMBAHASAN

3.1 Hasil Tampilan Sistem

Sistem yang dibangun terdiri dari tiga form yaitu form about sebagai halaman pembuka, form pengirim untuk menampilkan halaman untuk melakukan enkripsi dan kompresi, dan form penerima untuk menampilkan halaman untuk melakukan dekompresi dan deksripsi.

1. Form About.

Form about berfungsi sebagai halaman yang menampilkan data singkat pembuat aplikasi.



Gambar 7. Form About

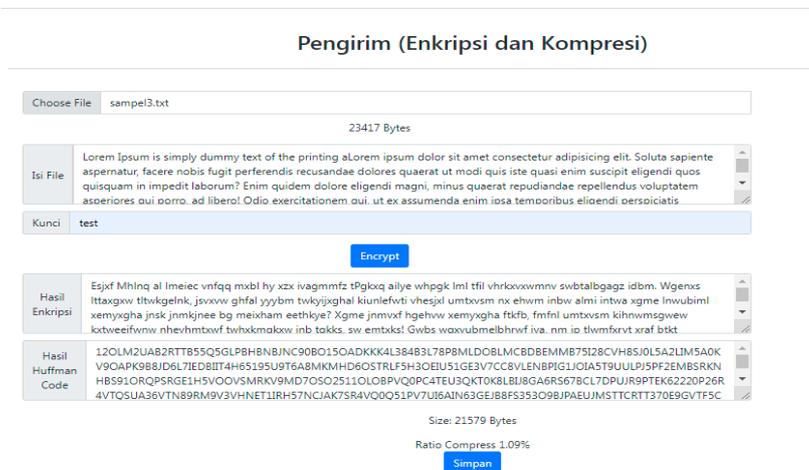
2. Form Pengirim

Halaman Pengirim merupakan halaman yang digunakan oleh pihak pengirim untuk melakukan proses enkripsi dan kompresi.



Gambar 8. Form Pengirim

Halaman pengirim ini akan muncul setelah mengklik *button* pengirim. Kemudian masukkan file dengan format .txt serta masukkan kunci dalam bentuk karakter. Setelah itu klik tombol *encrypt* sehingga tampil gambar 9.



Gambar 9. Hasil enkripsi dan Kompresi

Setelah tampil hasil Enkripsi dan kompresi dari file seperti diatas, maka simpan hasilnya dengan klik tombol simpan.

3. Form Penerima

Form penerima merupakan halaman yang digunakan oleh pihak penerima untuk melakukan proses Dekompresi dan Dekripsi.

Gambar 10. Form Penerima

Form penerima ini akan muncul setelah mengklik *button* penerima. Kemudian masukan file *ciphertext* berformat *.txt* yang telah disimpan sebelumnya serta masukkan kunci sebelumnya dalam bentuk karakter. Setelah itu klik tombol *decrypt* sehingga tampil gambar 11 berikut ini.

Gambar 11. Hasil Dekompresi dan Dekripsi

4. KESIMPULAN

Berdasarkan pembahasan dan hasil dari penelitian ini, maka dapat diperoleh beberapa kesimpulan sebagai berikut:

1. Dengan menggunakan algoritma *Polyalphabetic file* akan menjadi lebih aman karena file yang sudah dienkripsi menjadi *ciphertext* memiliki karakter yang tidak dapat dipahami.
2. Penggunaan metode *Huffman* dapat mengompres *file* menjadi lebih kecil dan dari hasil pengujian kompresi pada beberapa sampel dengan ukuran berbeda-beda menghasilkan ukuran ratio kompres yang berbeda, dan dari pengujian beberapa sampel menghasilkan kesimpulan semakin besar ukuran *file* yang dikompresi maka rasio kompresinya juga semakin besar dan jika ukuran file 3 kb maka kompresinya berhasil sedangkan jika filenya dibawah 3 kb maka ukuran filenya membesar.

3. Kombinasi antara pengamanan dan kompresi menggunakan algoritma *Polyalphabetic* dan *Huffman* merupakan perpaduan yang tepat. Dibuktikan dari file hasil dekripsi sama persis, seperti file awal sebelum dilakukan proses enkripsi, misalnya jenis huruf, ukuran huruf dan sebagainya.

DAFTAR PUSTAKA

- [1] M. Mawardina, U. I. N. Sunan, G. Djati, and M. Mawardina, "Aplikasi Kriptografi dengan Metode Vigenere Cipher Berbasis Web," no. May, 2016, doi: 10.13140/RG.2.1.2096.9849.
- [2] Y. Permanasari and E. Harahap, "Kriptografi Polyalphabetic," *Matematika*, vol. 17, no. 1, pp. 31–34, 2018, doi: 10.29313/jmtm.v17i1.4065.
- [3] A. Pahdi, "Algoritma Huffman Dalam Pemampatan Dan Enkripsi Data," *IJNS - Indones. J. Netw. Secur.*, vol. 6, no. 3, pp. 1–7, 2017, [Online]. Available: <http://ijns.org/journal/index.php/ijns/article/view/1461>.
- [4] E. Tarigan and D. H. S. Maha, "Kombinasi Vigenere Cipher Dan Polyalphabetic Cipher Pada Pengamanan File Text," *Publ. Ilm. ...*, pp. 71–77, 2018, [Online]. Available: <http://jurnalnya.stmikneumann.ac.id/index.php/pitin/article/view/41>.
- [5] I. M. Yusup and I. Purnamasari, "Implementasi Algoritma Caesar Cipher Dan Steganografi Least Significant Bit Untuk File Dokumen," vol. 6, pp. 434–441, 2020.
- [6] Y. Permanasari and E. Harahap, "Kriptografi Polyalphabetic One-Time Pad," *Matematika*, vol. 17, no. 2, pp. 45–48, 2018, doi: 10.29313/jmtm.v17i2.4451.
- [7] A. Wibowo, "Kompresi Data Menggunakan Metode Huffman," *Semantik*, vol. 2, no. 1, pp. 47–51, 2012, [Online]. Available: <http://publikasi.dinus.ac.id/index.php/semantik/article/view/70>.
- [8] R. Kasmala, A. B. Purba, U. T. Lenggana, and T. Informatika, "Kompresi Citra Dengan Menggabungkan Metode Discrete Cosine Transform (DCT) dan Algoritma Huffman," vol. 2, no. 1, pp. 1–9, 2017.
- [9] A. I. Auliyah, "Implementasi Kombinasi Algoritma Enkripsi Rivest Shamir Adleman (Rsa) dan Algoritma Kompresi Huffman Pada File Document," *Indones. J. Data Sci.*, vol. 1, no. 1, pp. 23–28, 2020, doi: 10.33096/ijodas.v1i1.6.
- [10] M. Akhmad, "Untuk Kompresi Data String Karakter," no. 13508059, 2019.