Aplikasi Enkripsi Kriptografi dengan Algoritma Blowfish dan Kompresi Huffman dalam Security Dokumen

Siti Muryanah[1*] & Syahriani Syam[2]

[1, 2] Teknik Informatika, Universitas Islam Syekh Yusuf
Jalan Maulana Yusuf No. 10 Babakan Kota Tangerang
Email: siti.muryanah@unis.ac.id, ssyam@unis.ac.id

Abstrak. Dengan adanya internet, begitu mudahnya terjadi penyalahgunaan infomasi oleh pihak yang tidak berwenang karena kemudahan dalam mengakses informasi. Oleh karena itu dibutuhkan usaha untuk menjaga keamanan informasi dari penyalahgunaan yang tidak diharapkan. Penelitian ini sebagai salah satu upaya mengamankan dokumen dalam aspek keamanan *authentication, confidentiality*,*integrity, availability* dan *non-repudiation*. Proses pengamanan dokumen dengan kriptografi yaitu perubahan pesan asli (*plaint text)* menjadi pesan rahasia/acak (cipher text). Algoritma kriptografi dalam penelitian ini menggunakan algoritma Blowfish. Namun sebelum melakukan enkripsi, dokumen dikompresi dahulu dengan metode Huffman agar ukuran dokumen menjadi lebih kecil sehingga proses enkripsi lebih optimal. Dimana karakter yang sering muncul frekuensinya dilakukan pengkodean mengunakan serangkaian kode bit pendek, sedangkan yang jarang muncul dengan panjang set bit.Hasil penelitian ini dapat mengamankan dokumen asli menjadi dokumen rahasia atau dokumen yang berisi kata atau kalimat yang sulit dimengerti atau acak dalam fotmat txt, docx, pdf dan xlsx. Selain itu dokumen akan dikompresi dengan metode Huffman yang memperkecil ukuran dokumen aslinya sehingga proses enkripsi menjadi lebih cepat dan optimal dengan waktu proses 1-3 detik.

Kata kunci*:* *Blowfish, Chiper Text, Enkripsi, Metode Huffman, Plain text.*

1 Pendahuluan

Latar belakang penelitian ini adalah masalah kerentanan penyalahgunaan infomasi oleh pihak yang tidak berwenang. Informasi yang dirasa rahasia menjadi informasi umum karena

Diterima 15/09/2021 Direvisi 17/10/2021 Diterima untuk publikasi 23/10/2021

mudahnya infomasi itu diakses dan tidak ada pengamanannya. Upaya menjaga keamanan informasi sangat dibutuhkan dengan penggunaan kriptografi dalam aspek keamanan *authentication, confidentiality,integrity, availability* dan *non-repudiation*[1]. Dalam penelitian ini informasi berbentuk dokumen dengan format txt, docx, pdf dan xlsx akan diubah menjadi dokumen yang berisi infomasi yang sulit dimengerti atau acak dengan format yang sama. Pihak yang tidak berwenang akan kesulitan mengetahui maksud dan tujuan informasi tersebut[2][3].

Beberapa algoritma dalam kriptografi adalah AES, DES, RSA, Blowfish dan lain sebagainya[4][5]. Algoritma tersebut dibedakan berdasarkan jenis kuncinya dalam proses enkripsi atau dekripsi[6][7]. Algoritma simetris (konvensional) adalah kunci yang digunakan untuk enkripsi dan dekripsi menggunkan kunci yang sama, sedangkan algoritma asimetris atau algoritma public key adalah kunci untuk enkripsi dan dekripsi berbeda. [4]Blowfish termasuk ke dalam algoritma simetris, dimana enkripsi dan dekripsi mengunakan kunci yang sama[8][9][1][10][11]. Blowfish juga *open source* dan tidak dipatenkan[12].

Metode kompresi juga digunakan dalam penelitian ini yaitu kompresi Huffman sebagai upaya dalam memperkecil ukuran file dalam penyimpanannya[13][14][3]. Dalam pemakaiannya, hasil kompresi dapat diklasifikasikan menjadi dua yaitu loseless dan lossy[15][3][9]. Perbedaan dari keduanya adalah hasil proses kompresi, dimana untuk metode loseless, informasi sebelum dan sesudah proses kompresi akan sama atau tidak ada perubahan sama sekali[16]. Metode ini biasanya digunakan untuk melakukan kompresi dokumen atau text. Dalam penelitian ini karena penulis ingin melakukan penelitian dengan kompresi text atau dokumen maka metode yang akan digunakan adalah metode loseless. Adapun yang termasuk ke dalam metode ini salah satunya adalah metode Huffman[14][17]. Kebalikan dari metode loseless, metode lossy akan menghasilkan informasi yang berbeda antara sebelum dan sesudah proses kompresi[16][15].

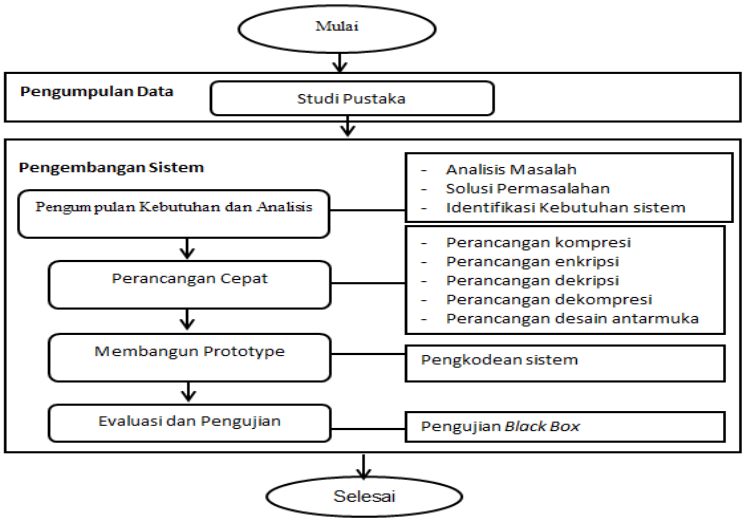
Penelitian yang sejenis mengenai keamanan kriptografi adalah [1] “Menyisipkan Pesan Rahasia Kedalam Gambar Dengan

Metode Blowfish Dan Least Significant Bit (Lsb)" dimana

dalam penelitian tersebut tidak adanya proses kompresi Huffman untuk mengoptimalkan dalam enkripsi dan dekripsi dokumen.

2 Metode Penelitian

Metode yang digunakan dalam penelitian ini adalah metode pengumpulan data dan pengembangan sistem. Informasi tersebut sesuai dengan gambar 1 berikut ini :



Gambar 1. Metode Penelitian

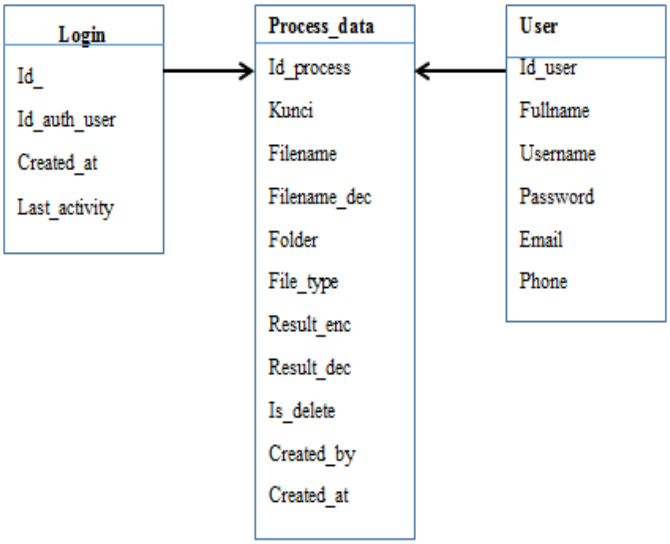
Metode penelitian ini diawali dengan pengumpulan data melalui studi pustaka jurnal-jurnal yang berkaitan dengan keamanan dokumen kriptografi dan kompres huffman. Metode pengembangan sistem diawali dengan pengumpulan kebutuhan dan analisis, perancangan cepat, membangun prototipe dan terakhir evaluasi serta pengujian.

Di dalam perancangan cepat, proses perancangan yang pertama adalah kompresi huffman dengan maksud ukuran file yang akan dienkripsi menjadi lebih kecil sehingga proses enkripsi menjadi lebih cepat. Setelah itu baru dilakukan perancangan enkripsi. Sebaliknya dari proses kompresi dan enkripsi, perancangan dekripsi dilakukan terlebih dulu daripada dekompresi. Akhir dari perancangan cepat adalah perancangan antar muka yang

memudahkan dalam penggunaan aplikasi ini. Membangun prototype dengan melakukan pengkodean sistem seperti ditunjukkan dalam bab pembahasan. Untuk evaluasi dan pengujian mengunakan pengujian *black box* dan uji coba masing-masing file dengan berbagai extention 3 file yang berbeda agar hasil yang diperoleh lebih akurat. .

3. Hasil dan Pembahasan

Dalam pembuatan aplikasi ini memerlukan basis data untuk menyimpan informasi yang ada. Adapun tabel yang diperlukan antara lain tabel login, tabel proses_data dan tabel user. Berikut class diagramnya :



Gambar 2. Class Diagram

3.1. Proses Kerja Aplikasi

Aplikasi ini terdapat proses enkripsi yang didahului proses kompresi dan dekripsi yang diikuti proses dekompresi. Langkah untuk melakukan enkripsi Blowfish dengan memilih menu proses, enkripsi dan tambah. Masukkan kunci, nama file dan masukkan file lalu simpan.

```
$this->form_validation->set_rules('kunci', '"Key tidak boleh kosong"', 'required');
$this->form_validation->set_rules('filename', '"Nama file tidak boleh kosong"', 'required');

if ($this->form_validation->run() == FALSE){
    $ret['msg'] = validation_errors(' ',' ');
}
else {
    $files = $file['files'];
    if (filesize($files['tmp_name']) > 1048576) {
        $ret['msg'] = 'Ukuran file lebih dari 2 MB!';
```

Gambar 3. Coding Proses Enkripsi (*input* file dan kunci)

Proses kompresi akan dilangsungkan terlebih dahulu baru proses enkripsi.

```
# 1. upload file (rename first)
upload_file_process('files', [
    'file_name' => $new_file_name,
    'path' => UPLOAD_FILE_DIR.$folder
]);

# 2. get data from file
$files['full_dir'] = UPLOAD_FILE_DIR.$folder.'/'.$new_file_name;
$content = extract_file($files);

# 3. compress text using huffman method
$huffman = huffman_method($content['strings']);

# 4. enkrip huffman using blowfish
$blowfish = blowfish_method($huffman, $post['key']);

# 5. package result to files again
$result = package_result($blowfish, $folder, $post['filename'], $files);
```

Gambar 4. Coding Proses Kompresi yang Dilangsungkan Dahulu daripada Proses Enkripsi

File yang digunakan dapat berupa file dengan format txt, docx, xlsx, dan pdf dengan kapasitas maksimal 8 MB. File akan ter-log dalam file enkripsi dan bisa didownload yang nantinya bisa didekripsi kembali.

```
<div class="form-group">
    <label for="files">
        File <i class="required">*</i> (ukuran maks: 8MB)
    </label>
    <input type="file" id="files" class="form-control" name="files" data-validation="required" accept=".txt,.pdf,.docx,.xlsx,.pptx" >
</div>
```

Gambar 5. Coding Ukuran File Masukan

Sedangkan untuk proses dekripsi, memilih menu dekripsi, aksi (di list file hasil enkripsi). Masukan kunci yang sama dalam proses enkripsi, nama file hasil dekripsi dan klik ubah. Untuk melihat hasil dekripsi, klik menu detail.

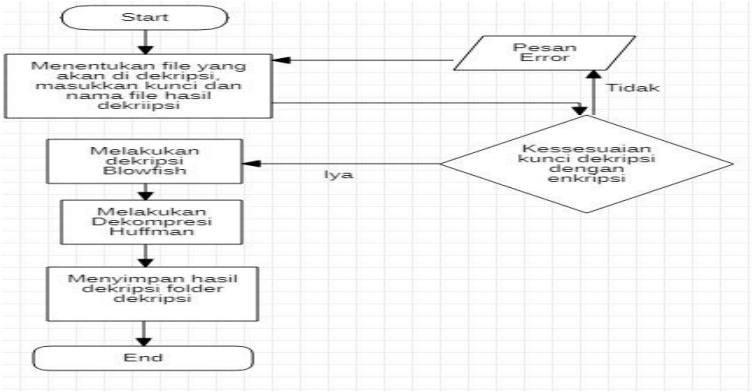
```
*/
function process(){
    $post = html_purify($this->input->post());
    $file = $_FILES;
    $ret['status'] = 0;

    $this->form_validation->set_rules('kunci', '"Key tidak boleh kosong"', 'required');
    $this->form_validation->set_rules('filename_dec', '"Nama file tidak boleh kosong"', 'required');
```

Gambar 6. Coding Proses Dekripsi (*Input* Kunci)

Alur dari proses kompresi dan enkripsi dapat dilihat dalam flowchart berikut ini :

Gambar 7. *Flowchart* Proses Enkripsi Blowfish dan Kompresi Huffman

Gambar 7 menjelaskan proses enkripsi dan kompresi, dimana proses kompresi Huffman dilakukan terlebih dahulu sebelum enkrpsi Blowfish berlangsung. Untuk alur dari proses dekompresi dan dekripsi dapat dilihat dalam flowchart berikut ini :

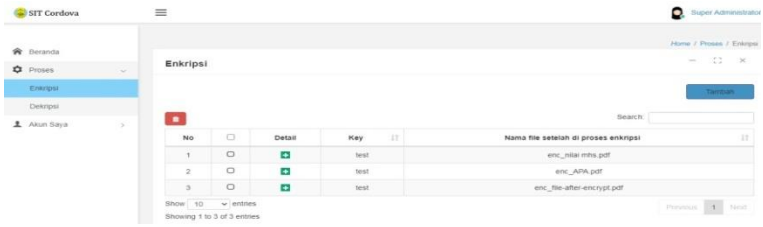


Gambar 8. *Flowchart* Proses Dekripsi Blowfish dan Dekompresi Huffman

Gambar 8 menjelaskan proses dekripsi dan dekompresi, dimana proses dekompresi Huffman dilakukan setelah dekripsi Blowfish berlangsung. Dalam proses ini diperlukan kunci yang sama seperti proses enkripsi, jika kunci tidak sama maka akan muncul notifikasi bahwa password salah (looping).

3.2. Tampilan Antar Muka

Tampilan antar muka dalam proses enkripsi dan dekripsi sebagai berikut :



Gambar 9. Tampilan Antar Muka dalam Proses Enkripsi dan Kompresi

Dalam gambar 9 menampilkan halaman tatap muka proses enkripsi, dimana file hasil enkripsi ter-log secara terurut sesuai waktu melakukan enkripsi.

Gambar 10. Tampilan Antar Muka Proses Dekripsi dan Dekompresi

Sama dengan tampilan antar muka proses enkripsi dan kompresi, gambar 10 juga menampilkan log proses dekripsi dan dekompresi sesuai waktu pelaksanaannya.

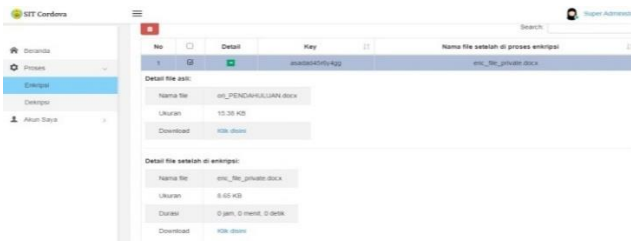
3.3. Evaluasi dan Pengujian

Proses pengujian aplikasi dengan menggunakan metode pengujian *black box*. Metode ini digunakan untuk memastikan bahwa semua perintah masukan dan hasil keluaran sesuai dengan perancangan.

Tabel 1. Evaluasi dan Pengujian

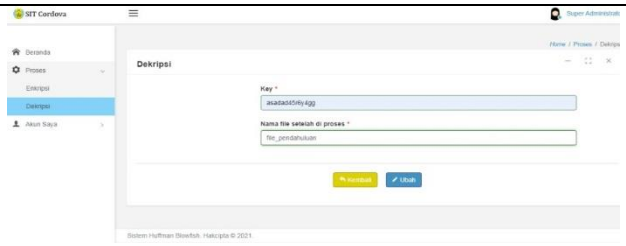
No	Item Uji	Kegiatan	Hasil yang diharapkan	Ket
1	Enkripsi dan kompresi	• Memasukkan kunci • Memberi nama file • Memilih file yang akan dienkripsi dan kompresi	Berhasil melakukan proses enkripsi dan kompresi.	✓

Gambar 11. Tampilan Antar Muka Proses Enkripsi

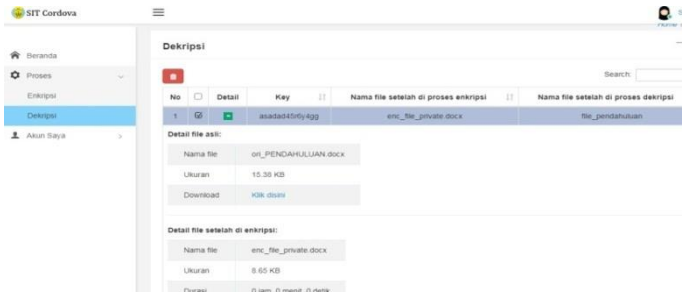


Gambar 12. Tampilan List Hasil Enkripsi

2	Dekripsi dan dekompresi	• Memasukkan kunci yang sama seperti enkripsi • Memberi nama file.	Berhasil melakukan proses dekripsi dan dekompresi.	✓



Gambar 13. Tampilan Antar Muka Dekripsi



Gambar 14. Tampilan List Dekripsi

Dalam evaluasi dan pengujian ini dilakukan ujicoba dengan beberapa extension file yang berbeda dengan masing-masing 3 file untuk memastikan keakuratannya.

Tabel 2. Pengujian Enkripsi dan Dekripsi

No	Tipe file	Kunci	Ukuran file asli	Ukuran file hasil enkripsi	Ukuran file hasil dekripsi	Waktu enkripsi	Waktu dekripsi
1.	Pdf	cbfnhmjm7	229.31 KB	20.53 KB	16.39 KB	00:00:02	00:00:01
2.		mkjhj8u76y7h6y	357.74 KB	27.63 KB	25.75 KB	00:00:02	00:00:02
3.		h8u6yjg8ug75r	489.75 KB	35.93 KB	1.18 KB	00:00:03	00:00:01
4.	Txt	ggdgd	1.75 KB	1.09 KB	1.75 KB	00:00:00	00:00:00
5.		gh6y6y7hhjjjhhjjj	2.21 KB	1.38 KB	2.21 KB	00:00:00	00:00:00
6.		selalilkheeee45r5r	978 byte	744 byte	978 byte	00:00:00	00:00:00
7.	Docx	jikaaaaa3w2	15.13 KB	8.65 KB	7.84 KB	00:00:00	00:00:00

8.		luthfiiiiii5r 45r5r5r	12.84 KB	8.14 KB	7.56 KB	00:00: 01	00:00:00
9.		Yusuf3333 333	12.84 KB	8.13 KB	7.56 KB	00:00: 00	00:00:01
1 0		f5r6y6y6y gt6y	11.72 KB	6.85 KB	11.72 KB	00:00: 00	00:00:00
1 1	Xls x	aaaTGTTT 76Y	13.93 KB	7.62 KB	13.93 KB	00:00: 00	00:00:01
1 2		aaagt6y6y6 y6y5r	9.84 KB	7.30 KB	9.84 KB	00:00: 00	00:00:00

Dari tabel 2 menunjukkan bahwa ukuran file hasil proses enkripsi menjadi sangat kecil dari ukuran file aslinya dan waktu proses sangat cepat berkisar 1 atau 3 detik.

4 Kesimpulan

Kesimpulan dari penelitian "Aplikasi Enkripsi Kriptografi dengan Algoritma Blowfish dan Kompresi Huffman dalam Security Dokumen" bahwa aplikasi ini berhasil mengamankan file dalam extention txt, docx, xlsx dan pdf dengan mengenkripsi file dan mendekripsikan file kembali.

Ukuran file hasil kompresi Huffman juga semakin kecil daripada ukuran file aslinya, sehingga waktu proses yang dibutuhkan untuk enkripsi dan dekripsi lebih cepat berkisar 1-3 detik (berbanding lurus dengan ukuran filenya).

5 Referensi

[1] S.- Muryanah, "Menyisipkan Pesan Rahasia Kedalam Gambar Dengan Metode Blowfish Dan Least Significant Bit (Lsb)," *JIKA (Jurnal Inform.*, vol. 4, no. 3, p. 87, 2020, doi: 10.31000/jika.v4i3.2869.

[2] D. Ariyus, *Pengantar Ilmu Kriptografi : Teori, Analisis, dan Implementasi*. Yogyakarta: Andi, 2008.

[3] A. Pahdi, "Algoritma Huffman Dalam Pemampatan Dan Enksripsi 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] Y. Kumar, R. Munjal, and H. Sharma, "Comparison of Symmetric and Asymmetric Cryptography with Existing Vulnerabilities and Countermeasures," vol. 11, no. 03, pp. 60–63, 2011.

[5] P. Patil, P. Narayankar, D. G. Narayan, and S. M. Meena, "A

Comprehensive Evaluation of Cryptographic Algorithms : DES ," *Procedia - Procedia Comput. Sci.*, vol. 78, no. December 2015, pp. 617–624, 2016, doi: 10.1016/j.procs.2016.02.108.

[6] "A Comparative Study and Performance Evaluation of Cryptographic Algorithms : AES and A Comparative Study and Performance Evaluation of Cryptographic Algorithms : AES and Blowfish," no. December, 2016.

[7] A. Verma, P. Guha, and S. Mishra, "Comparative Study of Different Cryptographic Algorithms Comparative Study of Different Cryptographic," no. March, 2016.

[8] A. Rana, "A Symmetrical key Cryptography Analysis using Blowfish Algorithm A Symmetrical key Cryptography Analysis using Blowfish Algorithm," no. July, 2016, doi: 10.17577/IJERTV5IS070276.

[9] Y. S. Triana and A. Retnowardhani, "Blowfish algorithm and Huffman compression for data security application," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 453, no. 1, 2018, doi: 10.1088/1757-899X/453/1/012074.

[10] M. Suresh and M. Neema, "Hardware implementation of blowfish algorithm for the secure data transmission in Internet of Things," *Procedia Technol.*, vol. 25, no. Raerest, pp. 248–255, 2016, doi: 10.1016/j.protcy.2016.08.104.

[11] M. N. Valmik and P. V. K. Kshirsagar, "Blowfish Algorithm," vol. 16, no. 2, pp. 80–83, 2014.

[12] T. S. Atia, "DEVELOPMENT OF A NEW ALGORITHM FOR KEY AND S-BOX GENERATION IN BLOWFISH ALGORITHM 2 . Description of Used Material," vol. 9, no. 4, pp. 432–442, 2014.

[13] S. S. A. P. U. Suherman, "Huffman Text Compression Technique," *Int. J. Comput. Sci. Eng.*, vol. 3, no. 8, pp. 103–108, 2016, doi: 10.14445/23488387/ijcse-v3i8p124.

[14] L. Laurentinus, H. A. Pradana, D. Y. Sylfania, and F. P. Juniawan, "Performance comparison of RSA and AES to SMS messages compression using Huffman algorithm," *J. Teknol. dan Sist. Komput.*, vol. 8, no. 3, pp. 171–177, 2020, doi: 10.14710/jtsiskom.2020.13468.

[15] M. Roslin and A. Neta, "Perbandingan Algoritma Kompresi Terhadap Objek Citra Menggunakan JAVA," vol. 2013, no. November, pp. 224–230, 2013.

[16] 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.

[17] S. Rahmawati, I. Taufik, and G. Sandi, "Implementasi Algoritma AES (Advanced Encryption Standard) 256 Bit Dan Kompresi Menggunakan Algoritma Huffman Pada Aplikasi Voice Recorder," *Prosiding-Seminar Nas. Tek. Elektro UIN Sunan Gunung Djati Bandung*, pp. 91–99, 2018.