Contents lists available at MID Publisher International

Technology Sciences Insights Journal

Journal homepage: https://journal.midpublisher.com/index.php/tsij

Keamanan data terenkripsi: studi kasus enkripsi AES dalam pengembangan web formulir aduan PPKS ITH

Rakhmadi Rahman, Andi Riah Zahirah
Prodi Sistem Informasi, Fakultas Sains, Institut Teknologi Bacharuddin Jusuf Habibie, Parepare, Indonesia

INFORMASI ARTIKEL

Sejarah Artikel:
Diterima: Juli 2024
Revisi: September 2024
Diterima: September 2024
Dipublikasi: November 20204

Kata Kunci:
Enkripsi, AES-256-CBC, Keamanan Informasi, Data Sensitif, Platform PPKS

*Penulis Korespondensi:
riahzahiraahh@gmail.com

ABSTRAK

Keamanan informasi menjadi krusial di era digital, terutama untuk data sensitif dari formulir pelaporan kekerasan seksual pada platform PPKS. Penelitian ini bertujuan menerapkan algoritma Advanced Encryption Standard (AES) untuk enkripsi data di platform tersebut. Metode yang digunakan meliputi studi literatur mengenai kriptografi dan AES, desain sistem enkripsi dan dekripsi, serta implementasi kode PHP dengan AES-256-CBC. Hasilnya menunjukkan bahwa enkripsi dan dekripsi data sensitif terlindungi secara efektif, menghasilkan ciphertext yang tidak dapat dibaca tanpa proses dekripsi yang tepat. Namun, ditemukan kendala dalam batasan ukuran data dan pengelolaan kunci. Saran perbaikan meliputi penyesuaian konfigurasi server dan peningkatan penanganan error untuk meningkatkan performa dan keamanan sistem.

ABSTRACT

Information security is crucial in the digital age, particularly for sensitive data from sexual violence reporting forms on the PPKS platform. This study aims to apply the Advanced Encryption Standard (AES) algorithm to encrypt data on this platform. The methodology includes a literature review on cryptography and AES, system design for encryption and decryption, and implementation of PHP code with AES-256-CBC. The results indicate that encryption and decryption of sensitive data are effectively protected, producing ciphertext that cannot be read without proper decryption. However, challenges were found in data size limits and key management. Recommendations for improvement include adjusting server configurations and enhancing error handling to improve system performance and security.

PENDAHULUAN

Di era digital saat ini, keamanan informasi merupakan komponen krusial dalam pengembangan sistem informasi. Data sensitif, terutama yang dikumpulkan melalui formulir, rentan terhadap ancaman seperti pencurian, intersepsi, dan manipulasi oleh pihak yang tidak bertanggung jawab. Oleh karena itu, enkripsi menjadi metode penting untuk menjaga kerahasiaan dan integritas data.

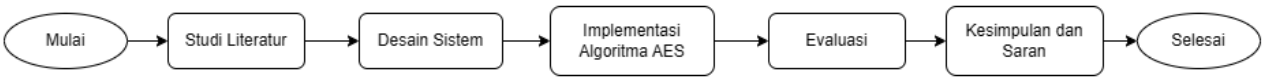
Platform PPKS (Pencegahan dan Penanganan Kekerasan Seksual) yang sedang dikembangkan berfungsi sebagai sarana pelaporan kejadian kekerasan seksual. Mengingat formulir aduan mengandung informasi sensitif mengenai pelapor dan terlapor, platform ini harus melindungi data dalam tabel formulir secara efektif.

Rumusan masalah yang akan dikaji adalah bagaimana algoritma kriptografi AES dapat diterapkan untuk mengenkripsi data dalam tabel formulir pada platform PPKS.Penelitian ini bertujuan untuk menerapkan algoritma kriptografi Advanced Encryption Standard (AES) dalam

enkripsi dan dekripsi data pada tabel formulir di platform PPKS. Dengan penerapan enkripsi yang kuat, diharapkan platform PPKS dapat meningkatkan keamanan sistem dan memperkuat kepercayaan pengguna dalam layanan pelaporan.

METODE

Metode yang akan digunakan pada penelitian ini ditunjukkan pada Gambar 1 yang menggambarkan rangkaian tahapan penelitian.



Gambar 1. Rangkaian Tahapan Penelitian

Studi Literatur

1. Kriptografi

Kriptografi adalah bidang yang menyelidiki bagaimana menyembunyikan tulisan atau huruf sehingga orang yang tidak berkepentingan tidak dapat membacanya. Dua komponen utama kriptografi adalah enkripsi dan dekripsi. Enkripsi mengubah pesan asli menjadi tidak dapat dipahami dalam bentuk aslinya. Dekripsi mengembalikan pesan yang telah disandikan ke bentuk aslinya. Pesan yang sudah disembunyikan disebut ciphertext, sedangkan pesan asli disebut plaintext.

Gambar 2 menunjukkan proses yang dilakukan untuk mengubah plaintext yang dimasukkan ke dalam blok enkripsi menjadi ciphertext. Kemudian, ciphertext ini diproses dalam blok dekripsi untuk menghasilkan kembali plaintext.

Gambar 2. Proses Enkripsi dan Dekripsi

Ada dua model algoritma enkripsi yaitu simetrik dan asimetrik. Model simetrik, atau enkripsi konvensional, menggunakan kunci yang sama untuk enkripsi dan dekripsi. Model asimetrik, atau enkripsi kunci publik, menggunakan dua kunci, satu untuk enkripsi dan satu lagi untuk dekripsi. Untuk enkripsi, kunci publik dapat dibagikan, tetapi kunci pribadi untuk dekripsi hanya dapat disimpan oleh pemiliknya[2][3].

2. AES (Advance Encryption Standard)

AES (Advanced Enctyption Standard) merupakan enkripsi simetris yang menggunakan kunci yang sama untuk enkripsi dan deskripsi. Kunci dari algoritma ini bersifat rahasia sehingga dapat disebut dengan algoritma kunci rahasia.

Proses enkripsi algoritma AES terdiri dari empat jenis transformasi byte: AddRoundKey, SubBytes, ShiftRows, dan MixColumns. Pada awal proses enkripsi, input disalin ke dalam state dan mengalami transformasi AddRoundKey. State kemudian melalui iterasi transformasi SubBytes, ShiftRows, MixColumns, dan AddRoundKey secara berulang sesuai dengan jumlah putaran yang ditentukan oleh panjang kunci. Jumlah putaran dapat dilihat pada tabel 1[4].

Tabel 1. Jumlah Putaran

	Jumlah Key (Nk)	Ukuran Block (Nb)	Jumlah Putaran(Nr)
AES – 128	4	4	10
AES – 192	6	4	12
AES – 256	8	4	14

Proses enkripsi dan dekripsi AES dijelaskan secara rinci di sini:

a. Inisialisasi dan Penyiapan Data

Blok Data: AES mengenkripsi data dalam blok 128-bit yang dibagi menjadi matriks 4x4 byte. Kunci: Panjang kunci 128-, 192-, atau 256-bit menentukan berapa banyak ronde yang digunakan dalam proses enkripsi.

b. AddRoundKey

Transformasi Awal Penambahan Kunci Ronde: Pada langkah pertama, setiap byte dalam blok data (state) dikenakan operasi XOR dengan kunci ronde saat ini. Ini adalah langkah penambahan kunci awal sebelum transformasi lainnya dimulai.

c. Iterasi Ronde

1) SubBytes (Subtitusi)

Tabel substitusi tetap yang dikenal sebagai S-Box menggantikan setiap byte dalam blok data dengan byte baru, meningkatkan keamanan data. Tabel subtitusi dapat dilihat pada gambar 3 di bawah ini :

x \ y	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Gambar 3. tabel S-Box

2) ShiftRows (Pergeseran Baris)

Dalam matriks data, baris-baris digeser secara siklik. Baris pertama tidak digeser; baris kedua digeser satu posisi ke kiri; baris ketiga dan keempat digeser dua posisi ke kiri, masing-masing. Difusi dalam blok data meningkat sebagai hasil dari langkah ini.

3) MixColumns (Pencampuran Kolom)

Dalam matriks data, setiap kolom dianggap sebagai polinomial dan dikalikan dengan polinomial tetap dalam ruang finite. Hal ini menghasilkan pencampuran data tambahan dalam kolom, yang meningkatkan keamanan.

Catatan: Langkah ini dihilangkan pada ronde terakhir.

4) AddRoundKey (Penambahan Kunci Ronde)
Untuk mengenkripsi blok data dalam setiap ronde, kunci ronde yang dihasilkan dari ekspansi kunci ditambahkan ke blok data melalui operasi XOR.

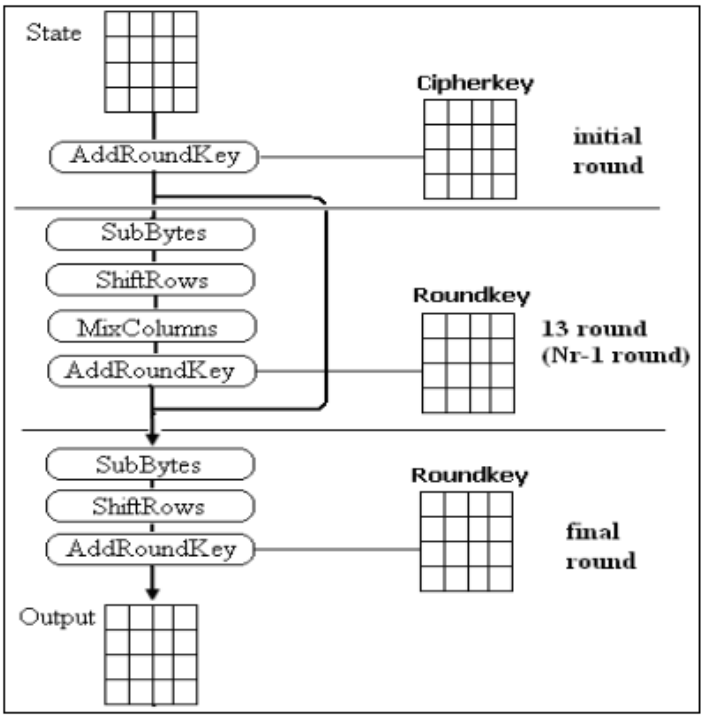
d. Ronde Terakhir

Langkah MixColumns tidak diterapkan pada ronde terakhir; transformasi SubBytes, ShiftRows, dan AddRoundKey hanya dilakukan pada ronde terakhir.

e. Hasil Enkripsi

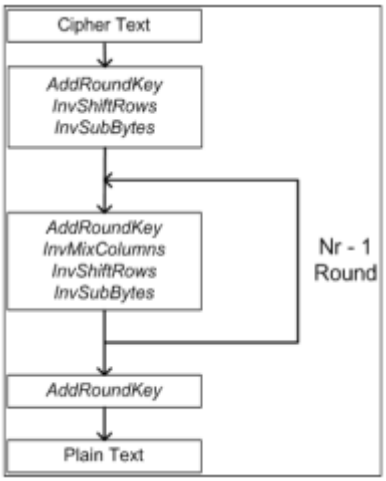
Ciphertext, atau data yang telah dienkripsi, adalah hasil akhirnya dari transformasi terakhir blok data.

Ilustrasi proses enkripsi AES dapat digambarkan seperti pada Gambar 4 di bawah ini :



Gambar 4. Ilustrasi Proses Enkripsi AES

Proses dekripsi adalah kebalikan dari enkripsi dan melibatkan langkah-langkah terbalik: Inverse SubBytes: Menggunakan tabel invers S-Box. Inverse ShiftRows: Baris-baris dipindahkan kembali ke posisi awal. Inverse MixColumns: Menggunakan matriks invers untuk pencampuran kolom. AddRoundKey: Sama dengan proses enkripsi, tetapi dilakukan dalam urutan terbalik dengan kunci ronde yang sesuai[4]. Ilustrasi proses dekripsi AES dapat digambarkan seperti pada Gambar 5 di bawah ini :



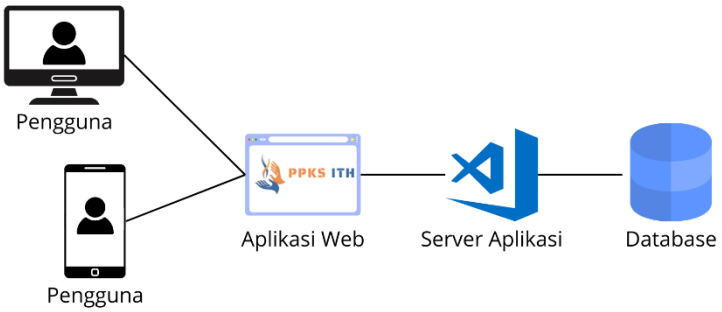
Gambar 5. Ilustrasi Proses Dekripsi AES

3. Desain Sistem

Ini akan menjelaskan cara platform PPKS menggunakan sistem enkripsi dan dekripsi Advanced Encryption Standard (AES). Desain sistem mencakup beberapa komponen utama, seperti arsitektur sistem serta alur kerja enkripsi dan dekripsi.

a. Arsitektur Sistem

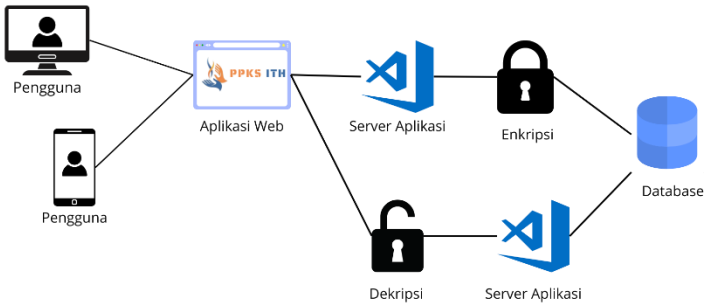
Arsitektur sistem adalah kerangka kerja yang menentukan struktur, komponen, dan hubungan antar komponen sistem teknologi informasi. Arsitektur sistem platform PPKS (Pencegahan dan Penanganan Kekerasan Seksual) dimaksudkan untuk menjaga data yang dikumpulkan melalui formulir aduan aman. Arsitektur Sistem dapat digambarkan seperti pada gambar 6.



Gambar 6. Arsitektur Sistem PPKS

b. Alur Kerja Enkripsi dan Dekripsi

Alur kerja enkripsi dan dekripsi adalah proses yang melindungi data dengan mengubahnya menjadi bentuk yang tidak dapat dibaca tanpa menggunakan kunci enkripsi yang tepat dan mengembalikannya ke bentuk aslinya ketika diperlukan. Untuk melindungi data pribadi dari orang yang tidak berhak mengaksesnya, proses ini sangat penting. Alur kerja dapat digambarkan seperti pada gambar 7.



Gambar 7. Alur Kerja Enkripsi dan Dekripsi

c. Implementasi Algoritma

Pada bagian ini, algoritma kriptografi AES-256-CBC diterapkan dalam kode PHP. Implementasi ini memastikan data pribadi yang dikumpulkan melalui formulir di platform PPKS (Pencegahan dan Penanganan Kekerasan Seksual) dilindungi.

Berikut adalah kode PHP yang digunakan untuk enkripsi dan dekripsi data menggunakan algoritma AES-256-CBC. Kode PHP yang digunakan digambarkan dalam gambar 8.

```php
<?php
require 'config.php';

function encryptData($data) {
    $key = hash('sha256', ENCRYPTION_KEY);
    $iv = openssl_random_pseudo_bytes(openssl_cipher_iv_length('aes-256-cbc'));
    $encrypted = openssl_encrypt($data, 'aes-256-cbc', $key, 0, $iv);
    return base64_encode($encrypted . '::' . $iv);
}

function decryptData($data) {
    $key = hash('sha256', ENCRYPTION_KEY);
    list($encrypted_data, $iv) = explode('::', base64_decode($data), 2);
    return openssl_decrypt($encrypted_data, 'aes-256-cbc', $key, 0, $iv);
}
?>
```

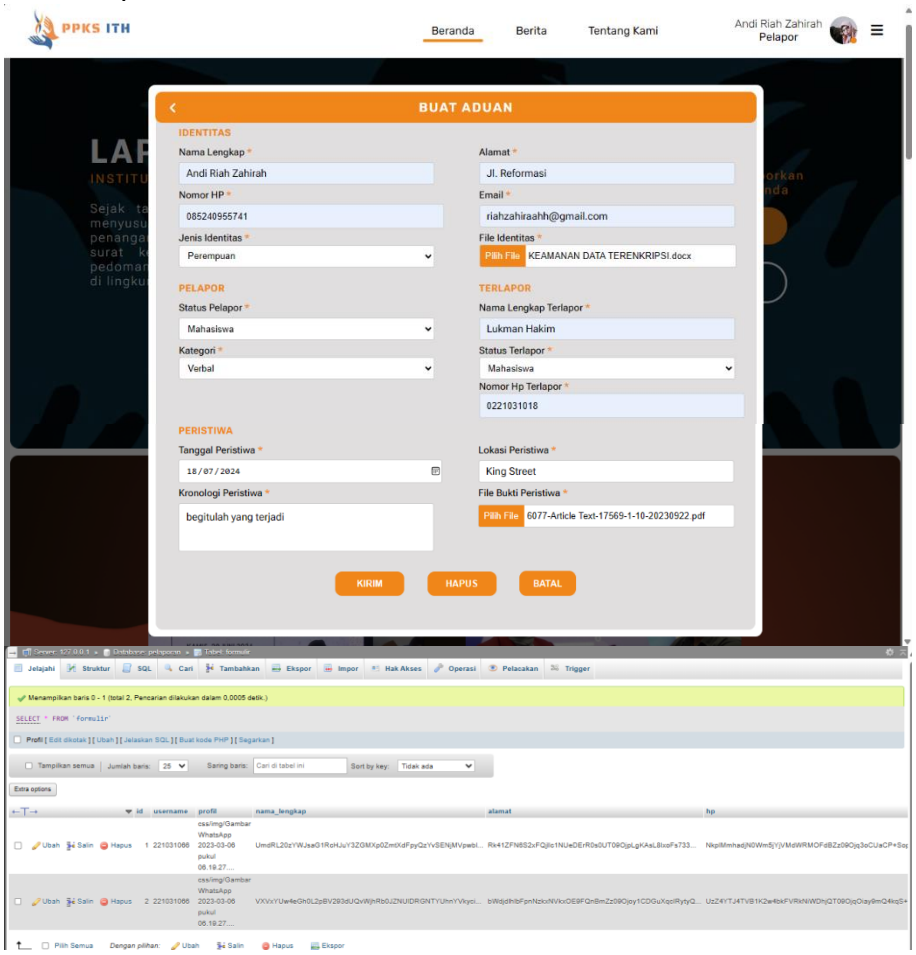
Gambar 8. Kode PHP Algoritma AES-256-CBC

HASIL

Hasil implementasi kode enkripsi dan dekripsi menggunakan algoritma AES-256-CBC akan dibahas pada bagian ini. Data yang dikumpulkan melalui formulir di platform PPKS dilindungi oleh kode yang menggunakan enkripsi simetris yang kuat.

1. Enkripsi

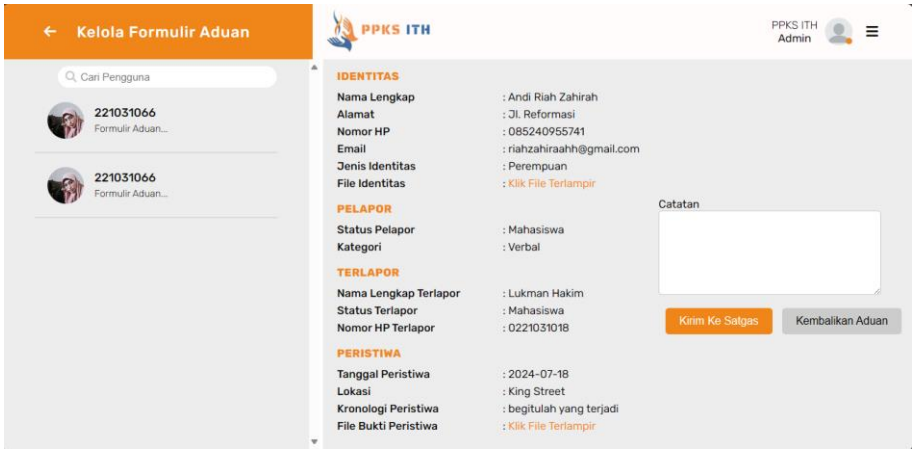
Fungsi enkripsiData berhasil mengenkripsi data sensitif dalam sistem. Fungsi ini menggabungkan data terenkripsi dengan vektor inicialisasi acak (IV), kemudian mengenkode hasilnya dengan Base64. Sebagai hasil dari pengujian, ciphertext yang dibuat tidak dapat dibaca tanpa menggunakan proses dekripsi yang tepat. Proses enkripsi ini dapat digambarkan pada Gambar 9, yang menunjukkan bagaimana data terenkripsi dan IV digabungkan sebelum di-encode dengan Base64.



Gambar 9 Proses Enkripsi Berhasil

2. Dekripsi

Fungsi decryptData berhasil mengembalikan data terenkripsi ke bentuk aslinya. Proses ini menggunakan kunci yang sama untuk dekripsi dan memisahkan data terenkripsi serta IV dari hasil dekripsi Base64. Hasil dekripsi konsisten dengan data asli sebelum enkripsi. Proses dekripsi ini dapat digambarkan pada Gambar 10, yang menunjukkan langkah-langkah pemisahan dan dekode untuk mengembalikan data asli.



Gambar 10 Proses Dekripsi Berhasil

3. Masalah yang Ditemukan

Sistem mengalami batasan dalam ukuran data yang dapat diproses. Pesan kesalahan yang muncul seperti "POST Content-Length of 59804836 bytes exceeds the limit of 41943040 bytes" menunjukkan bahwa ukuran data yang dikirim melebihi batas maksimum yang diizinkan oleh konfigurasi server. Hal ini dapat menghambat proses enkripsi dan dekripsi untuk file atau data besar.

KESIMPULAN DAN SARAN

Dengan menggunakan algoritma kriptografi AES-256-CBC pada platform PPKS, telah terbukti bahwa itu melindungi data sensitif. Fungsi enkripsi dan dekripsi yang berhasil diterapkan memastikan bahwa data pribadi yang dikumpulkan melalui formulir dilindungi dengan baik.

AES-256-CBC enkripsi berhasil menjaga kerahasiaan data sensitif dengan membuat ciphertext yang tidak dapat dibaca tanpa proses dekripsi yang tepat. Proses enkripsi, yang melibatkan penggabungan data terenkripsi dengan IV dan encoding Base64, berhasil mengembalikan data terenkripsi ke bentuk aslinya dengan menggunakan kunci yang sama untuk dekripsi. Hasil dekripsi konsisten dengan data asli sebelum enkripsi.

Beberapa kendala teknis ditemukan, termasuk pengelolaan kunci dan IV, serta batasan ukuran data yang diproses. Masalah-masalah ini mempengaruhi kemampuan sistem untuk menangani data besar dan dapat mengurangi kinerja sistem.

Untuk meningkatkan performa dan keamanan sistem, beberapa saran berikut dapat dipertimbangkan:

1. Mengatasi masalah batasan ukuran data dengan menyesuaikan konfigurasi server, seperti meningkatkan nilai post_max_size dan upload_max_filesize dalam file konfigurasi PHP (php.ini). Ini akan memungkinkan sistem untuk menangani file atau data besar dengan lebih baik.
2. Tingkatkan penanganan error dalam sistem dengan memastikan adanya validasi input yang memadai dan memberikan umpan balik yang jelas jika terjadi kesalahan dalam proses enkripsi atau dekripsi. Ini akan meningkatkan keandalan dan keamanan sistem secara keseluruhan.

DAFTAR PUSTAKA

Azhari, M., Mulyana, D. I., Perwitosari, F. J., & Ali, F. (2022). Implementasi Pengamanan Data pada Dokumen Menggunakan Algoritma Kriptografi Advanced Encryption Standard (AES). *Jurnal Pendidikan Sains Dan Komputer*, *2*(01), 163–171.

Sofwan, A., & Susanto, T. (2006). Aplikasi Kriptografi Dengan Algoritma Message Digest 5 (Md5). *Transmisi: Jurnal Ilmiah Teknik Elektro*, *8*(1), 22–27.

Sholeh, M., & Hamokwarong, J. v. (2011). Aplikasi Kriptografi Dengan Metode Vernam Cipher dan Metode Permutasi Biner. *Jurnal Ilmiah Momentum*, *7*(2).

Yuniati, V., & Indriyanta, G. (2011). Enkripsi dan dekripsi dengan algoritma aes 256 untuk semua jenis file. *Jurnal Informatika*, *5*(1).