5th Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)
7 September 2024 – Jakarta, Indonesia
Volume 3, Nomor 2, September 2024 - ISSN 2962-8628 (*online*)

PENGAMANAN FILE PADA SISTEM MASTER *VENDOR* BERBASIS WEB MENGGUNAKAN ALGORITMA AES PADA TRINITILAND

Nur Hena [1*], Dewi Kusumaningsih [2]

[1,2] Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur, Jakarta Selatan, Indonesia

Email: [1*]2011511108@student.budiluhur.ac.id, [2]dewi.kusumaningsih@budiluhur.ac.id
(* : *corresponding author*)

Abstrak-Trinitiland, sebagai perusahaan pengembang properti, menghadapi tantangan dalam memastikan pengelolaan kebutuhan barang dan jasa dari *vendor* dilakukan dengan aman dan efisien. Dengan pesatnya perkembangan teknologi informasi, penyimpanan dokumen di *cloud* telah menjadi solusi utama karena menawarkan kemudahan akses dan fleksibilitas. Namun, di balik kemudahan tersebut, terdapat risiko serius terkait keamanan data, terutama jika penyimpanan di *cloud* tidak dilindungi dengan baik. Data yang tidak aman rentan terhadap akses oleh pihak-pihak yang tidak berwenang, yang dapat mengakibatkan kebocoran informasi sensitif. Penelitian ini bertujuan untuk mengatasi masalah tersebut dengan mengusulkan penerapan metode kriptografi guna melindungi keamanan, integritas, dan kerahasiaan data vendor di Trinitiland. Algoritma kriptografi yang dipilih adalah *Advanced Encryption Standard* (AES) dengan panjang kunci 128-bit, yang dikenal memiliki tingkat keamanan tinggi dan efisiensi dalam proses enkripsi dan dekripsi. Algoritma ini diimplementasikan pada sistem database berbasis web yang dikembangkan khusus untuk Trinitiland. Dengan adanya enkripsi AES-128, data *vendor* yang tersimpan di cloud hanya dapat diakses oleh pihak yang memiliki kunci enkripsi, sehingga memastikan bahwa data tersebut terlindungi dari akses yang tidak sah. Sistem ini juga dirancang untuk memberikan kemudahan dan fleksibilitas akses bagi pengguna yang berwenang, tanpa mengorbankan aspek keamanan. Hasil penelitian menunjukkan bahwa sistem yang dikembangkan mampu meningkatkan tingkat keamanan data *vendor* secara signifikan, sambil tetap menjaga kemudahan akses yang dibutuhkan dalam operasional sehari-hari di Trinitiland.

Kata kunci: AES, Enkripsi, Dekripsi

FILE SECURITY IN THE WEB-BASED MASTER VENDOR SYSTEM USING THE AES ALGORITHM AT TRINITILAND

Abstract- *Trinitiland, as a property development company, faces challenges in ensuring that the management of goods and services from vendors is conducted securely and efficiently. With the rapid advancement of information technology, cloud document storage has become the primary solution due to its ease of access and flexibility. However, alongside these conveniences, there are serious risks related to data security, particularly if cloud storage is not properly protected. Unsecured data is vulnerable to unauthorized access, which can lead to the leakage of sensitive information. This research aims to address these issues by proposing the implementation of cryptographic methods to protect the security, integrity, and confidentiality of vendor data at Trinitiland. The chosen cryptographic algorithm is the Advanced Encryption Standard (AES) with a 128-bit key length, known for its high level of security and efficiency in the encryption and decryption processes. This algorithm is implemented in a web-based database system developed specifically for Trinitiland. With AES-128 encryption, vendor data stored in the cloud can only be accessed by those with the encryption key, ensuring that the data is protected from unauthorized access. The system is also designed to provide ease and flexibility of access for authorized users without compromising security. The research results show that the developed system significantly enhances the security level of vendor data while maintaining the ease of access required for Trinitiland's daily operations*.

Keywords*: : AES, Encryption, Decryption.*

1. PENDAHULUAN

Dalam kemajuan teknologi informasi, banyak pengguna memilih untuk menyimpan dokumen penting di *cloud*. Pilihan ini dikarenakan kemudahan dan kenyamanan yang didapatkan di penyimpanan *cloud*. Namun, dokumen yang disimpan dengan tidak benar menjadi rentan terhadap penyalahgunaan oleh pihak yang tidak berwenang. Oleh karena itu, menjaga keamanan data merupakan hal yang penting untuk memastikan integritas dan kerahasiaan data tetap terjaga. Salah satu cara untuk mengatasi hal ini adalah dengan menerapkan metode kriptografi. Pada kriptografi melibatkan proses enkripsi dan dekripsi untuk menjaga agar data tetap aman dan tidak dapat dibaca oleh pihak yang tidak berwenang. Penerapan kriptografi ini dapat meningkatkan tingkat perlindungan data dan membantu mengurangi risiko terkait pencurian data serta serangan terhadap sistem keamanan data [1].

Kriptografi adalah ilmu yang mempelajari teknik untuk menyembunyikan pesan baik dalam komunikas maupun penyimpanan data, sehingga hanya dapat dibaca oleh pihak yang memiliki otorisasi yang tepat. Proses utama dalam kriptografi adalah enkripsi, yang mengubah teks biasa (*plaintext*) menjadi teks yang terenkripsi (*ciphertext*) serta proses dekripsi, yang mengembalikan teks terenkripsi ke bentuk aslinya [2].

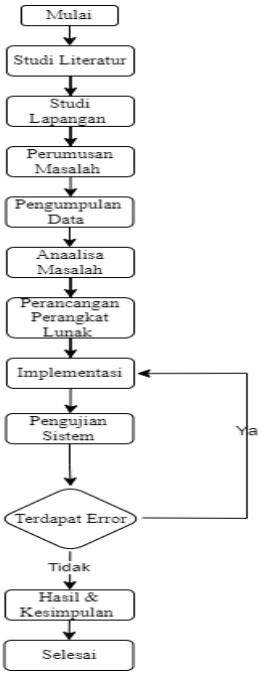
Pada penelitian sebelumnya yang dilakukan oleh Cristy & Riandri pada tahun 2021, algoritma AES telah diterapkan pada sistem yang digunakan oleh SMK Harapan Bangsa untuk mengamankan isi data uang SPP. Hasil dari penelitian tersebut menunjukan bahwa algoritma AES dengan panjang kunci 128 bit berhasil melakukan enkripsi dan dekripsi, sehingga dapat mengamankan data yang bersifat *sensitive* [3]. Selain itu Bibiola, dll pada tahun 2023 menerapkan algoritma AES untuk mengamankan *file* dalam aplikasi berbasis web. Berdasarkan pengujian yang telah dilakukan, algoritma AES dapat diimplementasikan pada bahasa pemrograman PHP & menggunakan *database* MySQL. Dengan adanya aplikasi ini, *file* yang disimpan di *server* telah dienkripsi, sehingga hanya dapat diakses oleh pihak berwenang yang memiliki *password* untuk melakukan dekripsi[4]. Perbedaan antara penelitian sebelumnya dan penelitian ini terletak pada kemampuan untuk mengenkripsi dan mendekripsi semua format *file* tanpa ada batasan. Sementara itu, penelitian sebelumnya hanya mampu mengenkripsi file dalam format PDF, *Word*, dan *Excel*.

Trinitiland merupakan grup perusahaan yang bergerak dalam industri pengembangan properti. Perusahaan ini memiliki beberapa proyek yang tersebar di berbagai lokasi seperti Tangerang, NTT, dan Batam, yang masing – masing memiliki Penanggung Jawab Proyek yang berbeda. Dengan banyaknya proyek yang sedang berlangsung, Trinitiland telah menjalin kerja sama dengan sejumlah vendor untuk memenuhi kebutuhan barang dan jasa dalam proyek-proyeknya. Setiap *vendor* yang bermitra dengan Trintiland diwajibkan untuk mengirimkan dokumen legalitas dan profil perusahaan mereka untuk dimasukkan ke dalam *database* Trinitiland. Dokumen-dokumen legalitas yang dikirimkan oleh vendor merupakan informasi rahasia perusahaan, sehingga hanya pihak berwenang yang diperbolehkan untuk mengaksesnya. Dokumen-dokumen ini akan disimpan oleh administrator master *vendor* pada komputer lokal yang berada di kantor. Namun, keterbatasan akses hanya dapat dilakukan dari kantor saja menyulitkan fleksibilitas, terutama jika suatu saat dokumen dari vendor tersebut diperlukan di luar kantor. Selain itu, penyimpanan dokumen pada komputer lokal juga menimbulkan kekhawatiran akan keamanan, karena komputer di kantor dapat diakses oleh siapa saja, yang berpotensi mengancam kerahasiaan data dan bahkan menyebabkan kehilangan data.

Berdasarkan latar belakang yang telah dijelaskan, penulis membangun sistem database berbasis web menggunakan algoritma AES 128 untuk mengamankan data vendor yang terdapat pada Trinitiland. AES 128 telah diuji dari beberapa penelitian sebelumnya bahwa dapat memenuhi mengamankan *file* dengan sangat baik. AES 128 memberikan keseimbangan yang optimal antara keamanan dan efisiensi yang sangat penting dalam operasional perusahaan. Dokumen yang telah dienkripsi hanya dapat diakses oleh team internal perusahaan yang mengetahui kuncinya, sehingga risiko akses oleh pihak eksternal sangat rendah. Dengan adanya *database* berbasis web ini, akses terhadap data vendor menjadi lebih 3 mudah dapat dilakukan di mana pun. Data disimpan dengan aman menerapkan algoritma AES 128, sehingga hanya pihak yang memiliki kunci enkripsi saja yang dapat mengakses dokumen legalitas *vendor*.

2. METODE PENELITIAN

Metode penelitian berfungsi sebagai panduan dan referensi dalam pelaksanaan penelitian agar hasil yang didapat sejalan sesuai dengan tujuan yang telah ditetapkan. Gambar 1 menunjukkan tahapan yang dilakukan dalam penerapan metode penelitian yang digunakan dalam penelitian ini.



Gambar 1. Metode Penelitian

a. Studi literatur, pengumpulan dan analisis informasi dari berbagai sumber seperti buku, artikel jurnal, dan dokumen lainnya untuk memahami kriptografi dan metode AES, guna membuat keputusan yang terinformasi dalam penelitian.
b. Studi lapangan, interaksi langsung dengan pihak Trinitiland untuk mengumpulkan data *vendor* dan memahami masalah yang dihadapi, agar solusi yang diusulkan dapat diterapkan dengan tepat.
c. Perumusan masalah, menetapkan masalah utama yaitu mengamankan data *vendor* di Trinitiland dengan penerapan metode AES-128.
d. Pengumpulan data, dilakukan melalui wawancara dan studi dokumentasi, sesuai dengan prosedur yang telah dijelaskan sebelumnya
e. Analisa Masalah, dilakukan melalui beberapa tahap analisis yang diperlukan dalam penelitian ini. Tahapan tersebut meliputi:
    1) Analisis Data
    Mengumpulkan dan mendeskripsikan data untuk merancang program yang efektif.
    2) Analisa Penerapan Algoritma
    Menganalisis penerapan AES untuk enkripsi dan dekripsi data, termasuk menentukan kunci dan proses enkripsi-dekripsi.
    3) Analisis Sistem
    Mengimplementasikan enkripsi *file* untuk melindungi data dalam *folder*, dengan modul enkripsi dan dekripsi terintegrasi dalam aplikasi.
f. Perancangan perangkat lunak, merancang sistem berdasarkan analisis yang telah dilakukan, termasuk desain sistem login, modul enkripsi dan dekripsi, serta antarmuka pengguna, mengikuti metode *waterfall*.
g. Implementasi, mengubah desain modul ke dalam bahasa pemrograman PHP dan menggunakan PHPMyAdmin untuk DBMS, dengan perangkat keras yang meliputi prosesor *Intel Core* i5, RAM 16 GB, dan SSD 512 GB.
h. Pengujian Sistem, memverifikasi bahwa sistem berfungsi sesuai dengan analisis dan desain, serta melakukan evaluasi kinerja, keamanan, dan keandalan untuk mengidentifikasi serta memperbaiki potensi masalah.
i. Hasil dan Kesimpulan, menyajikan kesimpulan mengenai efektivitas penerapan metode AES dalam mengamankan *file*, serta memberikan saran untuk perbaikan dan pengembangan sistem lebih lanjut.

2.1 Data Penelitian

Data yang digunakan dalam penelitian ini adalah data vendor yang digunakan oleh Trinitiland. Sumber data ini diambil langsung oleh peneliti dari sumbernya tanpa melibatkan pihak ketiga, dengan cara mengumpulkan

informasi langsung dari responden. Data yang diperoleh ini digunakan untuk menguji keamanan file menggunakan metode AES-128.

Tabel 1. Data Penelitian

Nama *File*	Jenis *File*	Ukuran *File*
Akta Perusahaan HCP	.pdf	415 kb
Data Perusahaan HCP	.docx	815 kb
Foto Perusahaan HCP	.png	457 kb
MOM HCP	.txt	13 kb
Proposal Kerjasama HCP	.pptx	96 kb
Scoring Perusahaan HCP	.xlsx	115 kb
Video Company Profile HCP	.mp4	5.625 kb
Foto Perusahaan HCP	.jpg	28 kb

2.2 Algortima AES

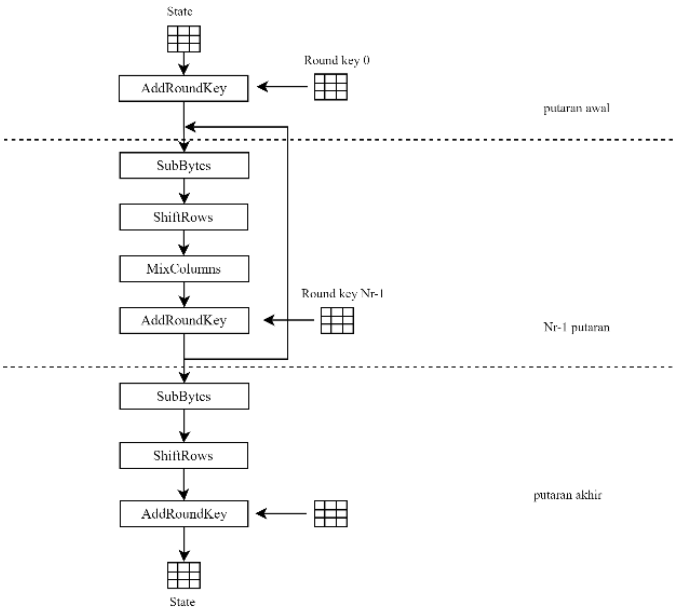
Algoritma AES merupakan sebuah teknik kriptografi blok simetris yang digunakan untuk melakukan enkripsi dan dekripsi terhadap berbagai jenis informasi, pesan, atau data. Dalam prosesnya, AES menggunakan kunci kriptografi dengan panjang yang bervariasi, yaitu 128, 192, dan 256-bit, untuk mengamankan proses enkripsi dan dekripsi terhadap blok data berukuran 128 bit[5]. Tiap blok data mengalami serangkaian putaran enkripsi yang khusus sesuai dengan jenis kunci yang digunakan. Variasi AES terdiri dari tiga tipe utama, yakni AES 128, AES 192, dan AES 256, yang dipilih tergantung pada kebutuhan keamanan dan kompleksitas data yang dihadapi. Pemilihan ukuran blok data dan kunci kriptografi ini mempengaruhi jumlah langkah yang harus dilalui dalam proses enkripsi [6]. Untuk rincian lebih lanjut mengenai jumlah putaran yang diperlukan untuk setiap blok data, dapat ditemukan pada Tabel 2 di bawah ini:

Tabel 2. Varian AES

Varian AES	Jumlah *Key* (Nk)	Jumlah Blok (Nb)	Jumlah Putaran (Nr)
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

2.3 Proes Enkripsi AES-128Bit

Enkripsi adalah langkah penting yang melibatkan mengubah data asli (plaintext) menjadi bentuk yang tidak dapat dimengerti (*ciphertext*). Tujuan utama dari proses enkripsi adalah untuk melindungi integritas dan privasi saat data atau pesan dikirim atau disimpan [7]. Dalam proses enkripsi, algoritma kriptografi digunakan bersama dengan kunci enkripsi. Selanjutnya, algoritma menghasilkan ciphertext dengan menjalankan serangkain operasi matematis dan logika pada data plaintext sesuai dengan kunci yang digunakan. Kunci enkripsi dapat berupa rangkaian angka atau teks, dan keamanan enkripsi sangan bergantung pada kompleksitas dan kerahasiaan kunci.



Gambar 2. Skema Enkripsi

Tahapan enkripsi AES melibatkan serangkaian operasi yang diterapkan pada setiap blok data *plaintext*. Berikut penjelasan 4 tahapan enkripsi utama yang digambarkan pada gambar 2 :

a. *AddRoundKey*

Pada tahap *AddRoundKey*, blok *plaintext* berukuran 128 bit dinyatakan dalam variable *state* terlebih dulu di-XOR dengan round key 0 atau kunci eksternal yang diberikan oleh pengguna yang juga memiliki panjang 128 bit [8]. Dengan menambahkan *roundkey* ke dalam *state*, keamanan enkripsi bergantung pada kunci yang digunakan.

Tabel 3. Ilustrasi Proses *AddRoundKey*

Plaintext					AES Key					*AddRoundKey Result*			
6D	6D	61	61		64	61	66	63		09	0C	07	02
75	61	69	62	XOR	62	30	65	30	=	17	51	0C	52
68	64	73	74	⊕	37	34	64	30		5F	50	17	44
61	72	72	73		35	61	62	62		54	13	10	11

Plaintext XOR AESKey

$6D$ = 0110 1101

64 = 0110 0100

0000 1001

= 09 ($1x1$)

b. *SubBytes*

SubBytes menjalankan operasi substitusi dengan memetakan setiap byte dari state dengan menggunakan S-*box*. S-*box* adalah matriks berukuran 16 x16, di mana setiap elemen matriks berisi nilai-nilai heksadesimal yang dihasilkan melalui perhitungan tertentu. Substitusi ini diterapkan pada setiap elemen dalam matriks *state* [9]. Proses substitusi secara keseluruhan menghasilkan nilai state yang baru, yang membantu meningkatkan kompleksitas data yang dienkripsi pada tahap ini.

Tabel 4. Ilustrasi Proses *SubBytes*

09	0C	07	02		01	FE	C5	77
17	51	0C	52	→	F0	D1	FE	00
5F	50	17	44		CF	53	F0	1B
54	13	10	11		20	7D	CA	82

c. *ShifRows*

ShiftRows melakukan operasi permutasi dengan menggeser tiga baris terakhir dari *array state*. Elemen-elemen pada baris 0 tetap pada tempatnya, elemen pada baris ke satu digeser satu *byte* ke kiri, elemen pada baris kedua

digeser sejauh dua *byte* ke kiri, elemen pada baris ketiga digeser sejauh tiga byte ke kiri. *ShiftRows* bertujuan untuk menciptakan variasi struktural dalam blok data yang dienkripsi, sehingga menghasilkan *ciphertext* yang lebih kompleks dan sulit untuk dianalisis [10].

Tabel 5. Ilustrasi Proses *ShifRows*

01	FE	C5	77		01	FE	C5	77
F0	D1	FE	00	→	D1	FE	00	F0
CF	53	F0	1B		F0	1B	CF	53
20	7D	CA	82		82	20	7D	CA

d. *MixColumns*

MixColumns adalah proses dimana matriks state dikalikan dengan matriks tetap dalam bidang galois. Proses ini melibatkan operasi perkalian dan penjumlahan modulo. Setelah operasi matriks selesai, nilai *state* yang baru dihasilkan, membuat pola dalam data menjadi lebih kompleks dan sulit untuk diprediksi [11].

Tabel 6. Ilustrasi Proses *MixColumns*

Hasil *MixColumns*			
18	C5	23	7C
31	14	F2	B3
B6	56	C7	64
3D	BC	61	B5

2.4 Rancangan Pengujian

Beberapa langkah pengujian dilakukan untuk memastikan setiap fitur aplikasi berfungsi dengan baik, serta untuk mengevaluasi waktu enkripsi dan dekripsi pada berbagai jenis dan ukuran file. Detail tahapan pengujian dapat dilihat dalam Tabel 7 berikut.

Tabel 7. Tahapan Pengujian

Komponen yang Diuji	Hasil yang Diharapkan
Tombol *Login*	Masuk ke *dashboard*
Tombol Tambah Data	Masuk ke *form* tambah data
Tombol Simpan Data	Menyimpan *form* yang telah diisi data baru
Tombol Lihat Data	Melihat data yang telah diinput
Tombol *Edit*	Mengubah data *vendor*
Tombol Hapus Data	Menghapus data *vendor*
Tombol *Upload*	Masuk ke *form upload file vendor*
Tombol Simpan	Menyimpan *file* yang telah di enkripsi
Tombol Lihat *File*	Melihat *file* yang telah di dekripsi
Tombol Hapus *File*	Menghapus *file*

Detail jenis dan ukuran file dapat dilihat dalam Tabel 8 berikut.

Tabel 8. Jenis dan Ukuran *File*

Nama *File*	Jenis *File*	Ukuran *File*
Akta Perusahaan HCP	.pdf	415 kb
Data Perusahaan HCP	.docx	815 kb
Foto Perusahaan HCP	.png	457 kb
MOM HCP	.txt	13 kb
Proposal Kerjasama HCP	.pptx	96 kb
Scoring Perusahaan HCP	.xlsx	115 kb
Video Company Profile HCP	.mp4	5.625 kb
Foto Perusahaan HCP	.jpg	28 kb

3. HASIL DAN PEMBAHASAN

Pada bagian hasil dan pembahasan ini, dibahas tentang pengujian program yang dilakukan untuk memastikan fungsi dan kinerja program secara akurat. Pengujian tersebut disajikan dalam bentuk tabel yang menunjukkan bahwa setiap fitur aplikasi berfungsi sesuai dengan rencana pengujian, serta tabel yang memuat waktu proses enkripsi dan dekripsi untuk berbagai jenis dan ukuran file.

Tabel 9. Hasil Rancangan Pengujian

Komponen yang Diuji	*Input*	Ekspektasi	Hasil
Tombol *Login*	*Username*, *Password*, Jabatan yang benar, klik tombol *login*	Berhasil masuk ke halaman *dashboard*	Sesuai ekspektasi, berhasil masuk ke halaman *dashboard*
Tombol Tambah Data	Klik tombol tambah data	Berhasil masuk ke halaman form tambah data	Sesuai ekspektasi, berhasil masuk ke halaman *form* tambah data
Tombol Simpan Data	*Input* data baru dan klik tombol simpan	Berhasil menyimpan data ke *database*	Sesuai ekspektasi, data berhasil disimpan di *database*
Tombol Lihat Data	Klik tombol lihat data	Berhasil melihat data yang telah di-*input*	Sesuai ekspektasi, berhasil melihat data yang telah di-*input*
Tombol *Edit*	Klik tombol *edit*	Berhasil masuk ke form *edit* data	Sesuai ekspektasi, berhasil, mengubah data pada *form edit* data
Tombol Hapus Data	Pilih *vendor*, lalu klik hapus data	Berhasil menghapus data *vendor*	Sesuai ekspektasi, berhasil mengapus data *vendor*
Tombol *Upload File*	Klik tombol *upload file*	Berhasil masuk ke *form upload file vendor*	Sesuai ekspektasi, berhasil masuk ke *form upload file vendor*
Tombol Simpan	Input file, kunci enkrips klik tombol simpan	Berhasil menyimpan *file vendor* yang telah di enkripsi	Sesuai ekspektasi, berhasil menyimpan *file vendor* yang telah di enkripsi
Tombol Lihat *File*	Input kunci dekripsi, klik tombol lihat *file*	Berhasil melihat *file* yang telah di dekripsi	Sesuai ekspektasi, berhasil melihat *file vendor* yang telah di dekripsi
Tombol Hapus *File*	Pilih *file*, klik tombol hapus	Berhasil menghapus *file* dari *database*	Sesuai ekspektasi, berhasil menghapus *file* dari *database*

Pada Tabel 10 dibawah ini menunjukan bahwa waktu proses enkripsi *file* bervariasi tergantung pada ukuran *file*, semakin besar ukuran *file* maka semakin lama waktu yang diperlukan untuk proses enkripsi.

Tabel 10. Tabel Pengujian Enkripsi

Nama *File*	Jenis *File*	Ukuran *File*	Waktu Enkripsi (*Seconds*)
Akta Perusahaan HCP	.pdf	415 kb	13,23
Data Perusahaan HCP	.docx	815 kb	25,08
Foto Perusahaan HCP	.png	457 kb	14,4
MOM HCP	.txt	13 kb	0,35
Proposal Kerjasama HCP	.pptx	96 kb	2,97
Scoring Perusahaan HCP	.xlsx	115 kb	3,44
Video Company Profile HCP	.mp4	5.625 kb	180,72
Foto Perusahaan HCP	.jpg	28 kb	0,82

Pada Tabel 11 dibawah ini menunjukan bahwa waktu proses dekripsi *file* bervariasi tergantung pada ukuran *file*, semakin besar ukuran *file* maka semakin lama waktu yang diperlukan untuk proses dekripsi.

Tabel 11. Tabel Pengujian Dekripsi

Nama *File* Setelah Enkripsi	Jenis *File*	Ukuran *File*	Waktu Dekripsi (*Seconds*)
88289-akta-perusahaan-hcp.pdf	.pdf	415kb	8,39
89915-data-perusahaan-hcp.docx	.docx	815kb	18,34
6982-foto-perusahaan-hcp.png	.png	457kb	9,42
37496-momhcp.txt	.txt	13kb	0,2
94107-proposal-kerjasama-hcp.pptx	.pptx	96kb	1,44
92987-*scoring*-perusahaan-hcp.xlsx	.xlsx	115kb	1,75
54718-*video-company-profile*--hcp.mp4	.mp4	5.625kb	131,56
73478-foto-perushaan-hcp.jpg	.jpg	28kb	9,01

4. KESIMPULAN

Berdasarkan hasil implementasi pengamanan database pada sistem Master Vendor berbasis web menggunakan algoritma *Advanced Encryption Standard* (AES) pada Trinitiland, dapat disimpulkan yaitu dengan adanya database berbasis web ini, mempermudah akses terhadap data *vendor* dan memungkinkan akses dari mana saja. Meskipun data dapat diakses dengan mudah, penerapan enkripsi AES-128 data menjamin bahwa data tersebut

tetap aman, karena hanya pihak yang mengetahui kunci enkripsi atau dekripsi saja yang dapat mengakses file tersebut. Berdasarkan kesimpulan yang sudah dijelaskan, penulis menyampaikan saran sebagai berikut :

a. Pengembangan lebih lanjut untuk pengamanan database : Implementasi pengamanan *database* pada sistem Master *Vendor* Trinitiland perlu dikembangkan lebih lanjut untuk meningkatkan keamanan data *vendor*. Hal ini penting untuk mencegah penyalahgunaan data dan kebocoran informasi yang mungkin terjadi.
b. Pengembangan metode enkripsi terbaru: pada pengembangan sistem selanjutnya, disarankan untuk menggunakan metode enkripsi yang lebih baru lagi atau mengkombinasikan dengan metode lain. Hal ini bertujuan agar hasil proses enkripsi menjadi lebih aman.

DAFTAR PUSTAKA

[1] Dola Ramalinda, Jayadi, and Agung Rachmat Raharja, "Strategi Perlindungan Data Menggunakan Sistem Kriptografi Dalam Keamanan Informasi," *J. Int. Multidiscip. Res.*, vol. 2, no. 6, pp. 665–671, 2024, doi: 10.62504/jimr679.
[2] M. Sari, H. D. Purnomo, and I. Sembiring, "Review : Algoritma Kriptografi Sistem Keamanan SMS di Android," *J. Inf. Technol.*, vol. 2, no. 1, pp. 11–15, 2022, doi: 10.46229/jifotech.v2i1.292.
[3] N. Cristy and F. Riandari, "Implementasi Metode Advanced Encryption Standard (AES 128 Bit) Untuk Mengamankan Data Keuangan," *J. Ilmu Komput. dan Sist. Inf.*, vol. 4, no. 2, pp. 75–85, 2021.
[4] F. Bibiola, T. U. Kalsum, and H. Alamsyah, "Penerapan Algoritma Advance Encryption Standard (AES) Untuk Pengamanan File Pada Aplikasi Berbasis WEB," *J. Surya Energy*, vol. 8, no. 1, p. 35, 2023, doi: 10.32502/jse.v8i1.6461.
[5] R. Mulud Muchamad, A. Asriyanik, and A. Pambudi, "Implementasi Algoritma Advanced Encryption Standard (Aes) Untuk Mengenkripsi Datastore Pada Aplikasi Berbasis Android," *J. Mnemon.*, vol. 6, no. 1, pp. 55–64, 2023, doi: 10.36040/mnemonic.v6i1.5889.
[6] N. Wachid Hidayatulloh, M. Tahir, H. Amalia, N. Afdlolul Basyar, A. Faizal Prianggara, and M. Yasin, "Mengenal Advance Encrytion Standard (AES) Sebagai Algoritma Kriptografi Dalam Mengamankan Data," *Digit. Transform. Technol.*, vol. Vol.03, no. No.1, pp. 1–10, 2023, [Online]. Available: https://jurnal.itscience.org/index.php/digitech/article/view/2293
[7] K. Yang and D. Kreativitas, "3 1,2,3," vol. 3, no. 4, pp. 413–446, 2024.
[8] Y. L. Ulu and Y. R. Kaesmetan, "Perbandingan Metode data Encryption Standard (DES) Dan Advanced Encryption Standard (AES) Pada Keamanan Jaringan Komputer Di SMK Willibrodus Betun," vol. 01, no. 05, pp. 26–32, 2024.
[9] F. Shofyan and R. T. Shita, "Implementasi Web Service Restful API dengan Autentikasi Personal Access Tokens dan Algoritma," vol. 12, pp. 108–114, 2024.
[10] D. Fahrizal, "Implementasi Kriptografi Aes 256 Bit Pada Aplikasi Pesan Di Android Dengan Raspberry Pi Server Berbasis Open Source," *TECHSI - J. Tek. Inform.*, vol. 14, no. 2, p. 107, 2023, doi: 10.29103/techsi.v14i2.12456.
[11] F. Duta Nugraha, K. Ahmad Baihaqi, H. Yulia Novita, and A. Mutoi Siregar, "Analysis and Implementation of Aes-128 Algorithm in Sukaharja Karawang Village Service System," *J. Tek. Inform.*, vol. 5, no. 3, pp. 855–864, 2024, [Online]. Available: https://doi.org/10.52436/1.jutif.2024.5.3.2038