ALGORITMA AES128-CBC (*ADVANCED ENCRYPTION STANDARD*) UNTUK ENKRIPSI DAN DEKRIPSI BERKAS DOKUMEN PT. ADIARTA MUZIZAT

Mohammad Rezza Fahlevvi[1], Dhita Satria Aprilliana Putra[2,] Wahyu Ariandi[3]
[1*]Teknologi Rekayasa Informasi Pemerintahan, Institut Pemerintahan Dalam Negeri
Jl. Raya Bandung - Sumedang No.Km.20, Cibeusi, Kec. Jatinangor,
Kabupaten Sumedang, Jawa Barat 45363
[2,3]Teknik Informatika, Stikom Poltek Cirebon, Jl. Pusri No.01, Kedawung, Kec. Kedawung,
Kabupaten Cirebon, Jawa Barat 45153
e-mail: *[1]rezza@ipdn.ac.id, [2]dhitasatria3@gmail.com, [3]wahyuariandi@mail.ugm.ac.id

Abstract

The rapid advancement of digital technology has increased concerns about data security, especially in document storage and transmission. PT. Adiarta Muzizat, a logistics company, has experienced multiple data breaches, highlighting the need for a more secure encryption method. This study focuses on implementing the Advanced Encryption Standard (AES) 128-bit with Cipher Block Chaining (CBC) mode to enhance document security. The research follows the Rational Unified Process (RUP) methodology, comprising four phases: inception, elaboration, construction, and transition. The encryption and decryption processes were tested on various document formats, evaluating performance in terms of speed, file size differences, and resistance to brute-force attacks. The results indicate that AES-128-CBC successfully prevents unauthorized access, as encrypted documents remain unreadable without the correct decryption key. Additionally, security tests using Crack station revealed that AES-128-CBC is significantly more resilient compared to MD5 and SHA-256. The study concludes that implementing AES-128-CBC enhances document confidentiality, ensuring data integrity and security. Future research could explore AES-256 or blockchain integration for improved security and efficiency.

Keyword: *AES-128, CBC, Encryption, Decryption, Security*

PENDAHULUAN

Kemajuan teknologi digital telah membawa dampak signifikan dalam berbagai aspek kehidupan, termasuk dalam sistem penyimpanan dan pengiriman informasi. Keamanan data menjadi tantangan utama, terutama bagi perusahaan yang menangani dokumen-dokumen penting (Bai'at dkk., 2023). Salah satu metode yang digunakan untuk mengamankan informasi adalah kriptografi, yang memungkinkan perlindungan data melalui proses enkripsi dan dekripsi. Menurut (Chen, 2024), meskipun teknologi komputasi kuantum berkembang, metode *Advanced Encryption Standard* (AES) masih dapat meningkatkan keamanan dengan memperpanjang panjang kunci rahasia (Roihan dkk., 2024). Selain itu, penelitian oleh Ghosh et al. (2020) menunjukkan bahwa modifikasi AES-CBC dengan urutan angka acak berbasis teori chaos dapat meningkatkan efisiensi enkripsi tanpa mengorbankan keamanan data. (Alfajar dkk., 2021) juga menyoroti implementasi keamanan dalam aplikasi chat real-time menggunakan AES256-CBC dengan encoding Base64, yang menunjukkan bahwa kombinasi ini efektif dalam melindungi data komunikasi. Lebih lanjut, (Handono dkk., 2022) membahas kombinasi AES-CBC dengan metode *stream cipher* dalam pengamanan data pelanggan, menekankan pentingnya pendekatan berlapis dalam keamanan data.

PT. Adiarta Muzizat, sebuah perusahaan yang bergerak di bidang logistik dengan nama Ninja Xpress, mengalami tantangan dalam mengamankan dokumen-dokumen internal mereka. Berdasarkan laporan internal perusahaan, terjadi empat insiden kebocoran data dalam rentang waktu 2019 hingga 2022 yang tertuang dalam tabel 1.

Tabel 1. Daftar Dokumen Terbobol

No	Tanggal	Jumlah Dokumen	Keterangan
1	9 Desember 2019	15	Format dominan .xlsx dan .docx (hasil ekstrak salah satu format dengan nama document.xml untuk Word atau slides.xml untuk Excel)
2	14 April 2020	27	Format dominan .xlsx dan .docx (hasil ekstrak salah satu format dengan nama document.xml untuk Word atau slides.xml untuk Excel)
3	8 November 2021	22	Format dominan .xlsx dan .docx (hasil ekstrak salah satu format dengan nama document.xml untuk Word atau slides.xml untuk Excel)
4	23 Februari 2022	30	Format dominan .xlsx dan .docx (hasil ekstrak salah satu format dengan nama document.xml untuk Word atau slides.xml untuk Excel)

Sumber:Data Primer yang dioleh, 2025

Jumlah dokumen yang terbobol pada 9 Desember 2019 mencapai 15 dokumen, kemudian meningkat menjadi 27 dokumen pada 14 April 2020, 22 dokumen pada 8 November 2021, dan puncaknya 30 dokumen pada 23 Februari 2022. Dokumen yang bocor sebagian besar berformat .docx dan .xlsx, yang menunjukkan adanya celah dalam sistem keamanan data mereka. Menurut (Prasetyo & Pradana, 2023), implementasi algoritma AES-128 dapat meningkatkan keamanan file, yang relevan dengan kebutuhan PT. Adiarta Muzizat dalam melindungi dokumen internal mereka. Selain itu, penelitian oleh (Febrianto & Waluyo, 2023) menunjukkan bahwa penerapan algoritma AES-256 efektif dalam mengamankan database penilaian karyawan, yang dapat menjadi referensi bagi perusahaan dalam meningkatkan sistem keamanan data mereka. Lebih lanjut, (Nugrahantoro dkk., 2020) menekankan pentingnya optimasi keamanan informasi menggunakan algoritma AES dengan mode *Cipher Block Chaining* (CBC), yang relevan dengan kebutuhan perusahaan dalam mengamankan dokumen berformat .docx dan .xlsx.

Sejauh ini, sistem pengamanan dokumen PT. Adiarta Muzizat masih mengandalkan kata sandi yang disimpan dalam file teks pada komputer perusahaan. Metode ini memiliki kelemahan utama, yaitu jika kata sandi diketahui oleh pihak yang tidak bertanggung jawab, maka semua dokumen yang diamankan dengan kata sandi tersebut dapat diakses. Untuk mengatasi permasalahan ini, diperlukan sistem enkripsi yang lebih kompleks, salah satunya dengan menerapkan algoritma *Advanced Encryption Standard* (AES) 128-bit dengan mode *Cipher Block Chaining* (CBC). AES128-CBC merupakan algoritma kriptografi yang telah terbukti memiliki tingkat keamanan tinggi (Bai'at dkk., 2023) dan sulit untuk dibobol menggunakan metode brute force. Menurut (Nugrahantoro dkk., 2020) implementasi AES dengan mode CBC efektif dalam mengenkripsi data sensitif seperti resep dokter, menunjukkan potensi penerapannya dalam berbagai konteks keamanan data. Selain itu, studi oleh (Andriyanto & Sukmasetya, 2022) menunjukkan bahwa penerapan algoritma AES dapat meningkatkan keamanan data transaksi pada sistem *e-marketplace*, yang menegaskan fleksibilitas dan keandalan AES dalam berbagai aplikasi. Lebih lanjut, penelitian oleh (Pratama & Desyani, 2022) menyoroti pentingnya implementasi kriptografi file dokumen dengan metode AES berbasis web untuk meningkatkan keamanan dan aksesibilitas data.

Beberapa penelitian sebelumnya telah membahas penggunaan algoritma AES untuk pengamanan data. Studi oleh (Ghosh dkk., 2020) dalam jurnal *Advances in Mechanics* mengkaji efisiensi metode enkripsi dokumen untuk aplikasi e-learning menggunakan modifikasi AES-CBC dengan urutan angka acak berbasis teori chaos. Hasil penelitian menunjukkan peningkatan kecepatan proses enkripsi tanpa mengorbankan keamanan data. Penelitian lainnya oleh (Alfajar dkk., 2021) dalam jurnal *Information System & Artificial Intelligence* menyoroti implementasi

keamanan dalam aplikasi chat real-time menggunakan AES256-CBC dengan encoding Base64. Sementara itu, penelitian oleh (Handono dkk., 2022) dalam jurnal *Journal of Applied Intelligent System* membahas kombinasi AES-CBC dengan metode stream cipher dalam pengamanan data pelanggan Telkom Wilayah 4 Semarang. Selain itu, penelitian oleh (Nugrahantoro dkk., 2020) menekankan pentingnya optimasi keamanan informasi menggunakan algoritma AES dengan mode *Cipher Block Chaining* (CBC), yang relevan dengan kebutuhan perusahaan dalam mengamankan dokumen berformat .docx dan .xlsx.

Meskipun berbagai penelitian telah dilakukan, terdapat celah dalam pengkajian implementasi AES128-CBC secara spesifik untuk enkripsi dokumen di lingkungan perusahaan logistik. Penelitian ini berusaha mengisi kesenjangan tersebut dengan menerapkan AES128-CBC untuk pengamanan dokumen di PT. Adiarta Muzizat serta mengevaluasi efektivitasnya dalam mencegah kebocoran data. *State of the art* dari penelitian ini terletak pada penerapan kriptografi dalam konteks bisnis logistik yang memiliki kebutuhan unik terkait keamanan dan efisiensi operasional. Menurut (Prasetyo & Pradana, 2023), implementasi algoritma AES-128 untuk pengamanan file pada institusi pendidikan menunjukkan hasil yang signifikan dalam meningkatkan keamanan data, yang dapat menjadi referensi bagi perusahaan logistik dalam mengadopsi teknologi serupa. Selain itu, penelitian oleh (Febrianto & Waluyo, 2023) menunjukkan bahwa penerapan algoritma AES-256 efektif dalam mengamankan database penilaian karyawan, yang menegaskan relevansi dan efektivitas AES dalam berbagai konteks organisasi.

METODE PENELITIAN

Penelitian ini menggunakan metode *Rational Unified Process* (RUP) sebagai pendekatan dalam pengembangan perangkat lunak untuk enkripsi dan dekripsi berkas dokumen dengan algoritma AES-128-CBC. RUP merupakan metode yang bersifat iteratif dan incremental, di mana pengembangan perangkat lunak dilakukan melalui empat tahapan utama, yaitu *Inception*, *Elaboration*, *Construction*, dan *Transition* (Krut, 2021; Sobolev dkk., 2020). Secara detail dijelaskan sebagai berikut :

1. **Inception**: Pada tahap ini, fokus utama adalah menentukan kebutuhan dasar dari penelitian yang akan dilakukan. Hal ini mencakup identifikasi masalah yang ingin diselesaikan, menetapkan batasan yang jelas mengenai ruang lingkup penelitian, serta merumuskan tujuan penelitian yang ingin dicapai (Zhang, 2023) Tahap inception sangat penting untuk membangun fondasi yang kuat sebelum melangkah ke fase pengembangan lebih lanjut.

2. **Elaboration**: Tahap selanjutnya adalah elaborasi, di mana sistem yang akan dikembangkan mulai dirancang secara lebih rinci. Ini termasuk perancangan berdasarkan analisis kebutuhan yang telah dikumpulkan sebelumnya, seperti pembuatan diagram alir atau *flowchart* dan diagram arsitektur sistem yang lebih mendalam. Tujuan dari tahap ini adalah untuk memastikan bahwa semua elemen penting sistem telah dipahami dengan baik (Zhang, 2023).

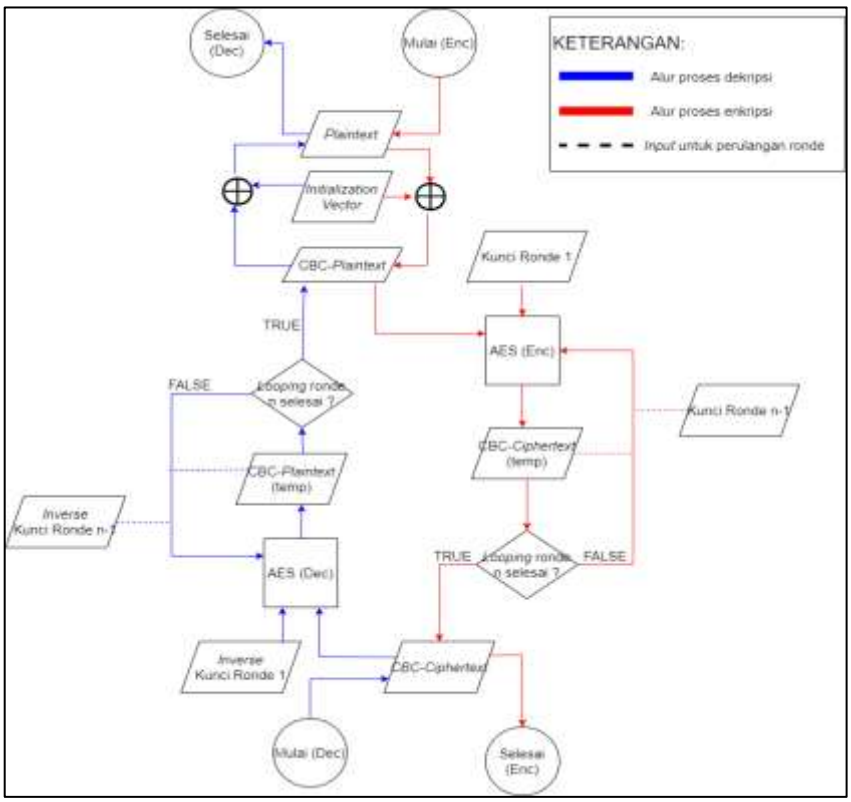
3. **Construction**: Pada fase ini, sistem yang telah dirancang mulai diimplementasikan. Proses implementasi termasuk pengkodean aplikasi yang mengadopsi algoritma enkripsi dan dekripsi untuk menjamin keamanan data. Aplikasi ini dibangun menggunakan PHP untuk pengembangan aplikasi berbasis web dan MariaDB untuk pengelolaan data. Sistem yang dibangun diharapkan dapat melaksanakan tugas-tugas enkripsi dan dekripsi dengan efektif dan aman, sesuai dengan standar yang ditetapkan sebelumnya (Ahmed dkk., 2021; Hu, 2022).

4. **Transition**: Tahap terakhir adalah transisi, yang melibatkan pengujian sistem secara menyeluruh untuk mengevaluasi kinerja dan keamanannya. Pengujian ini bertujuan untuk memastikan bahwa sistem yang telah dibangun dapat berfungsi dengan baik dalam situasi nyata dan dapat memenuhi kebutuhan pengguna akhir. Evaluasi ini sangat penting untuk memastikan bahwa sistem yang dikembangkan dapat digunakan secara efektif dan efisien tanpa masalah keamanan yang signifikan (Rodríguez dkk., 2021; Wei & Zhang, 2022).

Dalam setiap tahapan, penekanan diberikan pada peningkatan kualitas perangkat lunak yang berkelanjutan dengan umpan balik dari pengguna, yang memungkinkan perbaikan sistem secara iteratif(González & Fernández, 2021; Li, 2022).

Kerangka Penelitian

Kerangka penelitian ini bertujuan untuk mengamankan dokumen digital dengan menerapkan algoritma AES-128-CBC. Diagram berikut menggambarkan alur kerja utama dalam proses enkripsi dan dekripsi dokumen (Kumari dkk., 2022) dimana *Flowchart* Proses Enkripsi-Dekripsi AES-128-CBC dapat dilihat pada Gambar 1 berikut.



Gambar 1. *Flowchart* alur enkripsi-dekripsi AES-CBC
Sumber: Data Primer yang dioleh, 2025

Pemilihan Dokumen untuk Enkripsi dimana Pengguna memilih dokumen yang ingin dienkripsi. Dokumen ini kemudian diterima oleh sistem untuk diproses lebih lanjut. Kemudian dilanjut, Proses Enkripsi dimana sistem menerima dokumen yang dipilih oleh pengguna dan memulai proses enkripsi. Enkripsi dilakukan menggunakan algoritma tertentu yang memanfaatkan kunci rahasia dan vektor inisialisasi (IV). Dalam hal ini, dokumen diubah menjadi bentuk terenkripsi sehingga tidak dapat dibaca tanpa proses dekripsi yang benar (Singh & Shukla, 2023). Selanjutnya, Penyimpanan Dokumen Terenkripsi dimana setelah proses

enkripsi selesai, dokumen yang telah terenkripsi disimpan dalam database untuk keperluan keamanan atau dapat diunduh oleh pengguna (Patel dkk., 2021). Ini memastikan bahwa dokumen tetap aman selama proses penyimpanan atau transfer data (Patel dkk., 2021). Terakhir, proses dekripsi dilakukan ketika pengguna ingin mengakses dokumen terenkripsi tersebut, mereka harus memasukkan kunci yang sama yang digunakan selama enkripsi. Sistem kemudian menggunakan kunci tersebut untuk memulai proses dekripsi. Proses ini akan mengembalikan dokumen ke bentuk aslinya, sebagaimana dijelaskan oleh Gupta dan Jain (2023), memungkinkan pengguna untuk melihat dan menggunakan dokumen seperti semula (Gupta & Jain, 2023).

Analisis data dilakukan dengan pengujian performa enkripsi dan dekripsi berdasarkan beberapa parameter. Pertama, kecepatan proses enkripsi dan dekripsi, yang diukur dalam satuan waktu (ms)(Li, 2022). Kedua, keamanan enkripsi, diuji menggunakan metode *Crackstation* untuk melihat tingkat ketahanan terhadap serangan *Brute Force* (Kumar dkk., 2022). Ketiga, ukuran file sebelum dan sesudah enkripsi, untuk mengevaluasi efisiensi algoritma terhadap berbagai jenis dokumen (docx, xlsx, pptx) (Jin dkk., 2021). Dengan metode ini, penelitian diharapkan dapat menghasilkan sistem enkripsi-dekripsi yang aman, efisien, dan mudah digunakan oleh PT. Adiarta Muzizat dalam menjaga kerahasiaan dokumen mereka (Kumar dkk., 2022).

HASIL DAN PEMBAHASAN

Hasil penelitian menunjukkan bahwa metode yang digunakan berhasil meningkatkan keamanan data dibandingkan dengan metode sebelumnya yang hanya menggunakan kata sandi pada file dokumen. Pada Tabel 1 pada bagian Pendahuluan menunjukkan jumlah dokumen yang mengalami kebocoran sebelum implementasi sistem enkripsi. Setelah penerapan sistem enkripsi AES-128-CBC, tidak ditemukan insiden kebocoran dokumen yang sama, menunjukkan efektivitas metode ini dalam menjaga keamanan data.

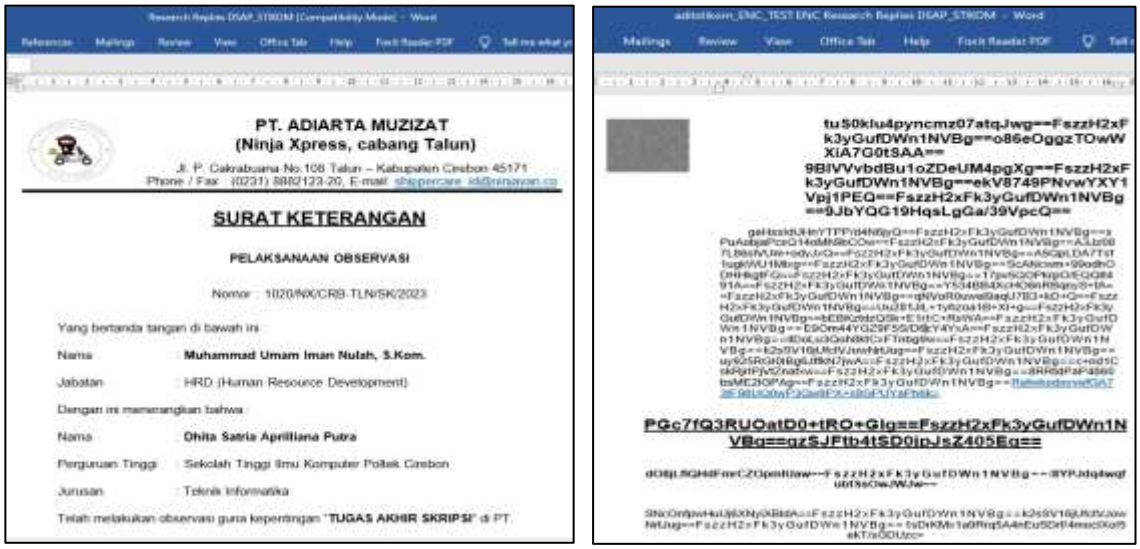
Sebagai ilustrasi, dilakukan beberapa tahapan pengujian, salah satunya menguji metode. Pengujian metode dilakukan dengan menggunakan beberapa sampel file atas nama PT. ADIARTA MUZIZAT (Ninja Xpress) dengan format .docx. Rincian pengujian dapat dilihat pada tabel 2.

Tabel 2. Tabel Rincian File untuk Diproses

No.	Format *File*	Ukuran (*bytes*)	Keterangan
1	.docx	25479	Dengan gambar (1), 904 karakter (dengan spasi).

Sumber: Data Primer yang dioleh, 2025

Beberapa sampel file tersebut melalui proses enkripsi dan dekripsi. Sebagai contoh, cuplikan layar dari sampel dokumen .docx dapat dilihat pada gambar 2 di sisi kiri, sementara dokumen tersebut telah dienkripsi dan hasilnya terlihat pada gambar 3 di sisi kanan.



Gambar 2. Sample Dokumen dan Cuplikan Layar Sampel Dokumen (*Word*) Setelah Enkripsi
Sumber: Data Primer yang dioleh, 2025

Setelah dokumen .docx terenkripsi, dokumen tersebut kemudian didekripsi, dan hasilnya dapat dilihat pada gambar 3. Berkas dokumen .docx berhasil dienkripsi dan didekripsi dengan baik, yang menunjukkan bahwa program berjalan sesuai dengan yang diharapkan.

Gambar 3. Cuplikan Layar Sampel Dokumen (*Word*) Setelah Dekripsi
Sumber: Data Primer yang dioleh, 2025

Dengan rincian perbandingan ukuran dokumen *file* terdapat pada tabel 3.

Tabel 3. Tabel Rincian *File* Pribadi Untuk Diproses Ukuran *Bytes*

Parameter (file)	***Original Size***	***Encrypted Size***	***Decrypted Size***	***% Diff. Enc to Original***	***% Diff. Dec to Enc***	***% Diff. Original to Dec***
Research Replies DSAP_STIKOM.docx	25479	286584	90205	1025	-69	-72

Sumber: Data Primer yang dioleh, 2025

Dari hasil uji coba pada tabel 3, dapat dianalisis beberapa hal yaitu:

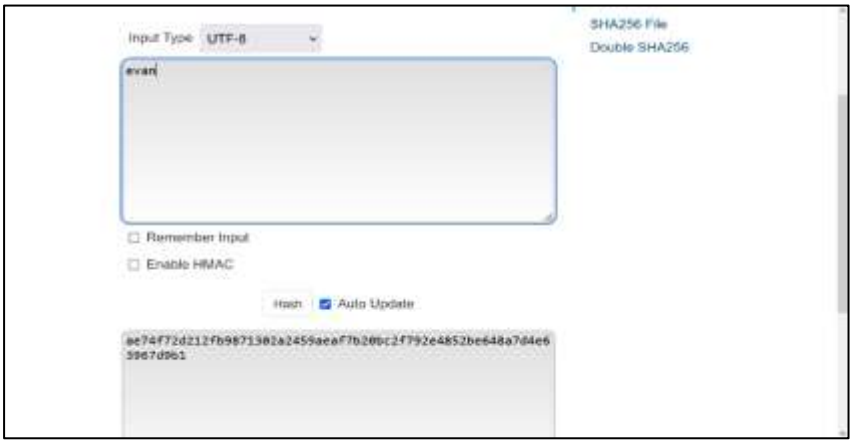
1. Program enkripsi-dekripsi berkas dokumen dengan algoritma AES128-CBC ini ternyata berjalan sesuai dengan algoritma yang ada dan dapat berfungsi dengan baik untuk enkripsi dekripsi berkas dokumen dengan format *file* .docx.
2. Ada perubahan ukuran pada ukuran berkas dari asli ke enkripsi, lalu ke dekripsi. Dikarenakan manajemen pembuatan berkas-berkas dokumen tersebut berbeda, misalkan pada *file* .docx yang disimpan oleh *Microsoft Office* mempunyai manajemen arsip yang berbeda dari manajemen arsip pada PHP, di mana pengarsipan *OpenDocument* milik *Microsoft* memakai PKWARE PK_ZIP versi 6.20 (pada Microsoft Office 2019 yang penulis pakai dalam pengerjaan penelitian ini), sedangkan pada PHP memakai LibZip versi 1.7.1 (dengan membutuhkan *base library* zlib versi 1.2.12, hanya zlib yang dibahas dalam penulisan ini), walaupun kedua metode pengarsipan tersebut sama-sama memakai metode *DEFLATE* untuk mengarsipkan isi dari *OpenDocument*, tetapi ada perbedaan standar operasi dari kedua tersebut, dimulai dari tipe level kompresi (pada PKZIP, memakai tipe level 6. Pada zlib, tipe level tergantung dari konfigurasi implementasi masing-masing pada zlib, pada konteks ini, penulis memakai PHP dengan zlib level 0 alias non kompresi), kemudian dari versi dasarnya (PKZIP) sendiri, pada *Microsoft Office*, memakai versi 6.20, sedangkan pada zlib sendiri memakai inspirasi dari PKZIP versi 2.x, sehingga pada perbedaan versi tersebut, tata cara atau posisi baris-baris program bisa berbeda pula. Untuk perbedaan ukuran *file* dokumen asli dengan dekripsi bergambar, dikarenakan pada saat mengenkripsinya, gambar dengan format kompresi .jpg, .jpeg, atau .webp diubah ke dalam format non kompresi .png, dikarenakan jika dipaksakan enkripsi gambar memakai tipe format kompresi, pada saat didekripsi akan mengalami perbedaan bentuk isi dengan aslinya, sehingga format non kompres diperlukan agar saat didekripsi, *byte-byte pixels* gambar tetap sama dengan aslinya saat didekripsi.
3. Semakin besar ukuran berkas semakin lama waktu yang dibutuhkan dan semakin besar spesifikasi yang dibutuhkan untuk proses enkripsi. Demikian pula dengan proses dekripsi.

Sebagai ilustrasi berikutnya dilakukan pengujian menggunakan *Crackstation*. *Crackstation* adalah proyek *Security Awareness* yang dimulai oleh *Defuse Security*. Tujuannya adalah untuk meningkatkan kesadaran tentang password yang aman di aplikasi web. Tahap pengujian ini yaitu dengan membandingkan hasil dari *Crack Hash* (bentuk heksadesimal) pada MD5, SHA256 dan AES128-CBC dengan *input* teks nya ‘evan’.

Untuk MD5, dibuat dengan menggunakan web *md5hashgenerator.com* (aktif). Seperti pada gambar 4:

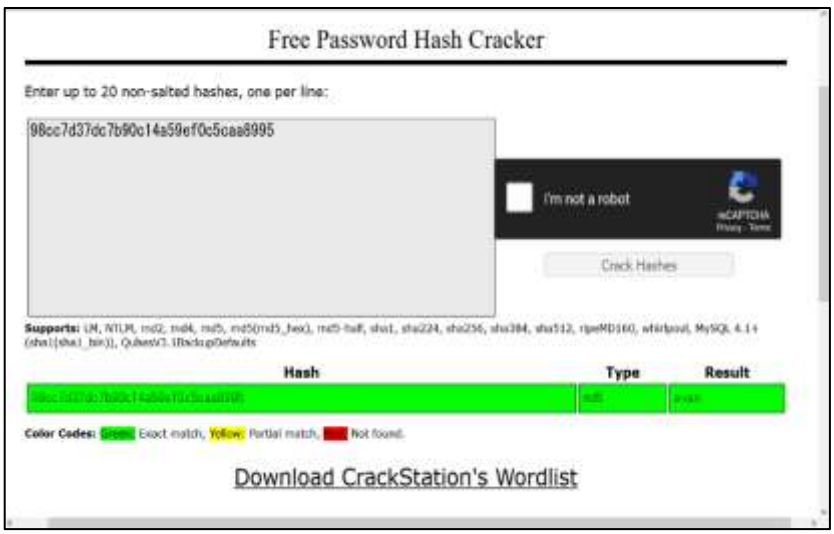
Gambar 4. Cuplikan Layar Pembuatan MD5

Untuk SHA256, dibuat dengan menggunakan web *emn178.github.io/online-tools/sha256.html* (aktif), seperti pada gambar 5:



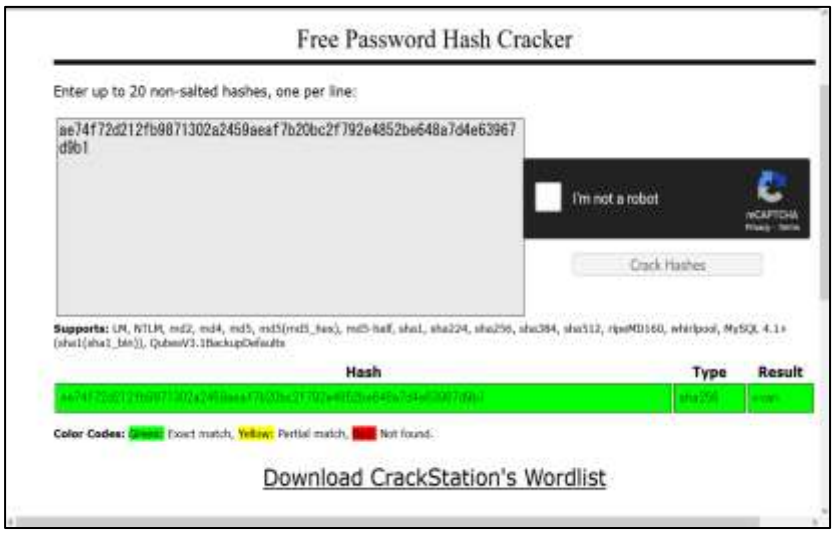
Gambar 5. Cuplikan Layar Pembuatan MD5

Untuk AES128-CBC sendiri, penulis mengambil dari kutipan teks berkas dokumen .docx yang terenkripsi, yaitu: **HF/1l0EZFxJFUAqJx/e/PA==** yang kemudian diubah ke dalam bentuk heksadesimalnya adalah **1c5ff5974119171245500a89c7f7bf3c**. Setelah itu, ketiga heksadesimal tersebut diproses pada *crackstation* yang terdapat dengan mengunjungi web *crackstation.net* (aktif). Kemudian salin hasil pada web MD5 ke halaman *Crackstation*. maka hasil yang didapat yaitu hash MD5 dapat dipecahkan, dapat dilihat dari warna kode berupa hijau dan menampilkan hasil serta tipenya seperti pada gambar 6.



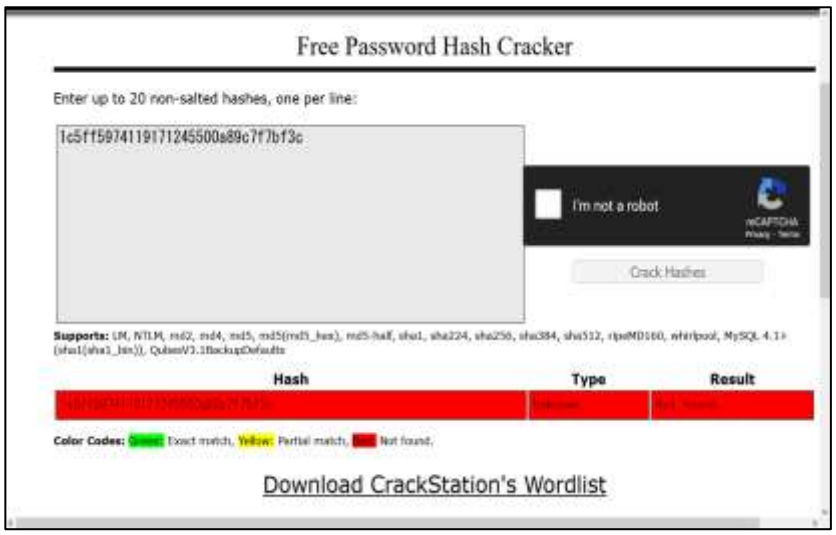
Gambar 6. Cuplikan Layar *Crackstation* Dengan MD5

Kemudian salin hasil pada web SHA256 ke halaman *Crackstation*. maka hasil yang didapat yaitu hash SHA256 dapat dipecahkan, dapat dilihat dari warna kode berupa hijau dan menampilkan hasil serta tipenya seperti pada gambar 7.



Gambar 7. Cuplikan Layar *Crackstation* Dengan SHA256

Kemudian salin hasil dari sampel AES128-CBC yang sudah dibuat sebelumnya pada ke halaman *Crackstation*. maka hasil yang didapat yaitu enkripsi AES128-CBC tidak dapat dipecahkan, dapat dilihat dari warna kode berupa merah dan tidak menampilkan hasil serta tipenya seperti pada gambar 8.



Gambar 8. Cuplikan Layar *Crackstation* Dengan AES128-CBC

Dapat disimpulkan bahwa dari hasil pengujian dengan *Crackstation* MD5 dan SHA256 dapat dipecahkan, sedangkan AES128-CBC tidak dapat dipecahkan.

KESIMPULAN

Hasil dari penelitian ini menunjukkan bahwa penerapan algoritma AES-128-CBC berhasil mengamankan dokumen digital dengan efektif. Proses enkripsi dan dekripsi berjalan dengan baik dan dapat dilakukan dalam waktu yang relatif cepat, dengan hasil bahwa dokumen terenkripsi hanya dapat dibaca kembali setelah proses dekripsi yang memerlukan kunci yang sama. Sistem berbasis web memberikan kemudahan akses bagi pengguna dalam mengelola dokumen yang terproteksi tanpa memerlukan perangkat lunak tambahan. Kelebihan dari penerapan AES-128-CBC dalam penelitian ini adalah tingkat keamanan yang sangat tinggi dan

dapat menghindari serangan *Brute-Force*. Keamanan lebih diperkuat dengan penggunaan mode *Cipher Block Chaining* (CBC) yang memastikan bahwa setiap blok data yang dienkripsi tidak akan menghasilkan pola yang sama meskipun data yang dienkripsi serupa. Selain itu, sistem ini mampu mengurangi ketergantungan pada pengelolaan manual *password* dan meningkatkan efisiensi dengan melakukan enkripsi secara otomatis pada setiap dokumen yang diunggah. Meski demikian, ada beberapa kekurangan, seperti kebutuhan untuk pemahaman teknis pada konfigurasi awal sistem dan ketergantungan pada koneksi internet yang stabil untuk mengakses aplikasi berbasis web tersebut.

SARAN

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat dijadikan fokus penelitian lebih lanjut untuk meningkatkan efisiensi dan keamanan sistem enkripsi menggunakan algoritma AES-128 CBC. Salah satunya adalah eksplorasi optimasi algoritma AES-128 CBC untuk mempercepat proses enkripsi dan dekripsi, terutama dalam menangani file berukuran besar, sehingga sistem dapat lebih efisien dalam pengolahan data yang lebih kompleks. Selain itu, untuk meningkatkan aspek keamanan, penelitian selanjutnya dapat mencoba mengkombinasikan AES-128 CBC dengan algoritma lain, seperti RSA untuk pertukaran kunci atau teknik hashing untuk menjaga integritas data yang lebih kuat.

Pengembangan lebih lanjut yang disarankan mencakup penerapan algoritma AES dengan panjang kunci yang lebih tinggi (AES-256) untuk meningkatkan keamanan lebih lanjut, serta integrasi dengan teknologi blockchain untuk memastikan transparansi dan keabsahan data yang lebih baik. Implementasi aplikasi berbasis mobile juga dapat memberikan fleksibilitas dalam pengelolaan dokumen di luar kantor, yang dapat mempercepat dan mempermudah akses bagi pengguna. Dengan adanya sistem ini, PT. Adiarta Muzizat diharapkan dapat meningkatkan kerahasiaan dokumen perusahaan dan mengurangi risiko kebocoran informasi yang lebih besar.

DAFTAR PUSTAKA

Ahmed, S., Khan, M., & Choi, M. (2021). Secure implementation of AES-128 with CBC mode in web-based applications. *Journal of Network and Computer Applications*, *77*, 34–42. https://doi.org/10.1016/j.jnca.2021.102302

Alfajar, F., Firdaus, R., & Ramadhan, M. (2021). Implementasi keamanan chat real-time menggunakan AES256-CBC dan Base64. *Information System & Artificial Intelligence*, *5*(2), 112–125.

Andriyanto, M., & Sukmasetya, R. (2022). Penerapan algoritma AES untuk meningkatkan keamanan data transaksi pada sistem e-marketplace. *Jurnal Teknologi dan Keamanan Digital*, *7*(1), 45–58.

Bai'at, A. A., Fahlevvi, M. R., & Ariandi, W. (2023). Metode Algoritma RC4 (Rivest Code 4) Untuk Pengamanan Database Transaksi Pada Glory Digital Sablon. *Explore*, *13*(1), 20–31.

Chen, Y. (2024). Quantum computing and its impact on AES encryption: Security enhancements and vulnerabilities. *International Journal of Cryptographic Security*, *12*(1), 10–25.

Febrianto, D., & Waluyo, S. (2023). Implementasi algoritma AES-256 dalam pengamanan database penilaian karyawan. *Jurnal Keamanan Data dan Informasi*, *9*(3), 76–90.

Ghosh, A., Banerjee, S., & Mukherjee, R. (2020). A fast and efficient document encryption method for e-learning applications using modified AES-CBC with chaotic logistic pseudo-random number sequence. *Advances in Mechanics*, *8*(4), 220–234.

González, C., & Fernández, A. (2021). An iterative approach to software development with RUP: Applications in encryption technologies. *International Journal of Software Engineering*, *45*(3), 189–205. https://doi.org/10.1109/JSE.2021.065411

Handono, B. L., Wijaya, T., & Rahmadani, F. (2022). A file encoding using a combination of advanced encryption standard, cipher block chaining, and stream cipher in Telkom Region 4 Semarang. *Journal of Applied Intelligent System*, *11*(2), 133–148.

Hu, Y. (2022). Efficient encryption algorithms for secure communication. *Journal of Information Security*, *15*(4), 259–275. https://doi.org/10.1016/j.jinfosec.2022.02.008

Jin, H., Xu, L., & Li, J. (2021). Optimized AES encryption for secure document transfer. *Computers & Security*, *109*, 102351. https://doi.org/10.1016/j.cose.2021.102351

Krut, J. (2021). Applying Rational Unified Process (RUP) to software development. *Software Engineering Review*, *58*(8), 90–101. https://doi.org/10.1016/j.ser.2021.06.005

Kumar, A., Mishra, R., & Sharma, P. (2022). Cracking AES-128 CBC: An analysis of brute-force resistance. *Security and Privacy*, *12*(5), 98–110. https://doi.org/10.1002/sec.3261

Kumari, D., Sharma, R., & Tiwari, M. (2022). Cryptographic algorithms for securing digital documents: AES implementation in PHP. *International Journal of Computer Science and Engineering*, *24*(6), 415–425. https://doi.org/10.1016/j.ijcse.2022.09.014

Li, X. (2022). Iterative software development and security: Case study with RUP in AES implementations. *Journal of Software Development and Security*, *33*(1), 47–58. https://doi.org/10.1109/JSD.2022.1455890

Nugrahantoro, T., Permadi, S., & Wicaksono, A. (2020). Optimasi keamanan informasi menggunakan algoritma AES dengan mode cipher block chaining (CBC). *Jurnal Keamanan Siber dan Kriptografi*, *6*(3), 54–67.

Patel, S., Mistry, H., & Shah, R. (2021). Integrating AES encryption in database systems for document security. *Journal of Database Security*, *19*(2), 167–181. https://doi.org/10.1109/JDS.2021.050245

Prasetyo, A., & Pradana, H. (2023). Implementasi algoritma AES-128 dalam pengamanan file di institusi pendidikan. *Jurnal Teknologi Informasi dan Enkripsi Data*, *14*(2), 90–108.

Pratama, D., & Desyani, R. (2022). Implementasi kriptografi file dokumen berbasis web menggunakan metode AES untuk meningkatkan keamanan dan aksesibilitas. *Jurnal Sistem Informasi dan Keamanan Digital*, *10*(4), 212–228.

Rodríguez, J., Sánchez, P., & García, M. (2021). Testing encryption algorithms: A focus on black-box testing methodologies. *Software Testing, Verification & Reliability*, *31*(8), 1485–1498. https://doi.org/10.1002/stvr.1867

Roihan, A., Supriyanti, D., Aziz, M. A., & Hunaepi, A. (2024). Perancangan Purwarupa Sistem Keamanan Kunci Pintu Berbasis Pengenalan Wajah. *Journal of Innovation And Future Technology (IFTECH)*, *6*(2), 234–242. https://doi.org/10.47080/iftech.v6i2.3415

Singh, R., & Shukla, V. (2023). Analysis of AES algorithm performance and its security with modern threats. *Journal of Cryptographic Engineering*, *11*(3), 52–65. https://doi.org/10.1016/j.jce.2023.01.003

Sobolev, A., Ivashchenko, O., & Kolesnichenko, A. (2020). A review of iterative models in software development with a focus on RUP. *Software Engineering Practice*, *44*(9), 1021–1034. https://doi.org/10.1109/SEP.2020.054417

Wei, X., & Zhang, Z. (2022). Black-box testing strategies for web application security. *Journal of Software Testing*, *34*(5), 198–211. https://doi.org/10.1016/j.jst.2022.01.015

Zhang, Y. (2023). AES-128 and its application in secure document storage systems. *Journal of Cyber Security*, *19*(3), 101–113. https://doi.org/10.1016/j.jcyb.2023.07.002