E-ISSN : 2407 – 3911
P-ISSN : 2686 - 0333

KOMBINASI ALGORITMA KRIPTOGRAFI AES-256 DAN SHA3-512 UNTUK MENINGKATKAN KEAMANAN DOKUMEN PDF

Asep Rizal Nurjaman[1], Agus Tinus Turnip[2]
Sistem Informasi[1,2]
Institut Teknologi Nasional[1,2]
Jl. PHH Mustofa No. 34, Bandung, Jawa barat, Indonesia
aseprizal@itenas.ac.id

Abstrak

Dokumen digital saat menjadi salah satu komponen penting dalam berbagai aspek kehidupan saat ini, mulai dari penyimpanan data pribadi hingga pertukaran informasi bisnis yang sensitif. Ancaman Intersepsi, atau upaya untuk melakukan intersepsi informasi secara melawan hukum, termasuk situasi di mana pihak yang tidak berwenang mencoba untuk mengakses atau mengambil data tanpa izin dari pemiliknya. Sehingga dibutuhkan tindakan yang tepat untuk mengamankan dokumen digital dari ancaman kejahatan. Salah satu solusi yang ditawarkan dengan menerapkan teknik kriptografi. Beberapa penelitian sebelumnya telah dikembangkan dengan menggabungkan AES dengan algoritma lain. Salah satunya adalah dengan menggunakan fungsi hash kriptografi seperti SHA-256 dan SHA-512. Hingga saat ini SHA (*Secure Hash Algorithm*) masih mendapatkan perkembangan yang lebih baru, SHA-3 menjadi versi terbaru dari SHA yang menawarkan peningkatan dalam keamanan dan efisiensi. Penelitian ini bertujuan untuk mengevaluasi efektivitas penggabungan AES-256 dengan SHA3-512 dalam melindungi dokumen digital, terutama dalam konteks aplikasi web. Peneliti ingin memberikan solusi tingkat keamanan yang lebih baik dengan menggunakan AES-256 dengan SHA3-512. Dengan kombinasi ini, dokumen tidak hanya dienkripsi menggunakan AES-256 untuk melindungi isinya, tetapi juga menggunakan SHA-3 untuk memastikan integritas dan autentikasi. Sehingga penelitian ini dapat membantu memberikan alternatif dalam memperkuat keamanan dokumen digital dan melindunginya dari berbagai ancaman kriminal di lingkungan digital yang semakin kompleks.

Kata kunci: AES-256, SHA3-512, *Algorithm*

Abstract

Digital documents have become one of the important components in various aspects of life today, ranging from personal data storage to the exchange of sensitive business information. The threat of Interception, or attempts to unlawfully intercept information, includes situations where unauthorized parties attempt to access or retrieve data without the permission of the owner. Therefore, proper measures are required to secure digital documents from the threat of crime. One of the most effective solutions is by applying cryptography. Several previous studies have been developed by combining AES with other algorithms. One of them is by using cryptographic hash functions such as SHA-256 and SHA-512. Until now SHA is still getting more recent developments, SHA-3 being the latest version of the SHA that offers improvements in security and efficiency. This research aims to evaluate the effectiveness of combining AES-256 with SHA3-512 in protecting digital documents, especially in the context of web applications. Researchers want to provide a better security level solution by using AES-256 with SHA3-512. With this combination, documents are not only encrypted using AES-256 to protect their content, but also using SHA-3 to ensure integrity and authentication. So that this research can help provide alternatives in strengthening the security of digital documents and protecting them from various criminal threats in an increasingly complex digital environment.

Keywords: AES-256, SHA3-512, Algorithm

I. PENDAHULUAN

Perkembangan teknologi yang pesat mendorong transformasi digital dari yang awalnya konvensional menjadi otomatis, kemudahan menjadi sebuah alasan untuk digitalisasi dokumen. Banyak dokumen yang sudah ditransformasikan dalam bentuk dokumen digital. Namun keamanan dari dokumen ini menjadi hal yang utama dimana beberapa dokumen penting hanya diberikan akses pada orang-orang tertentu. Bagaiamana kita bisa yakin bahwa tidak ada yang membaca atau megubah dokumen yang kita simpan dokumen tersebut, terlebih saat dokumen tersebut disimpan dalam storage berbasis online. Teori tentang keamanand at aini dipelajari dalam keilmuwan kriptografi. Kriptografi adalah Teknik untuk menyandikan sebuah pesan. Konsep yang ada pad kriptografi ini meliputi CIA (confidentiality, Integrity dan Authentication) (Stallings, W). Pada 2024 PT Artha Bumi Mining mengalami kerugian yang sangat besar disebabkan oleh pemalsuan dokumen tambang di Sulteng (Tim regional). Salah satu solusi yang ditawarkan untuk masalah tersebut adalah algoritma kriptografi dengan skema enkripsi dan dekripsi dokumen digital yang bisa menjaga kerahasiaan, integritas data dan autentikasi dari pengguna. Kriptografi telah terbukti sebagai solusi untuk melindungi data digital dari berbagai ancaman yang semakin kompleks (Nasution, A. B). Algoritma seperti *Advanced Encryption Standard* 256 (AES-256) telah menunjukkan efektivitas yang tinggi dalam melindungi data dari serangan brute force (I. G. Indra). Selain itu, kombinasi AES-256 dengan fungsi *hash* kriptografis seperti SHA-256 dan SHA-512 telah digunakan untuk meningkatkan integritas dan keamanan data (A. Dharmawan)

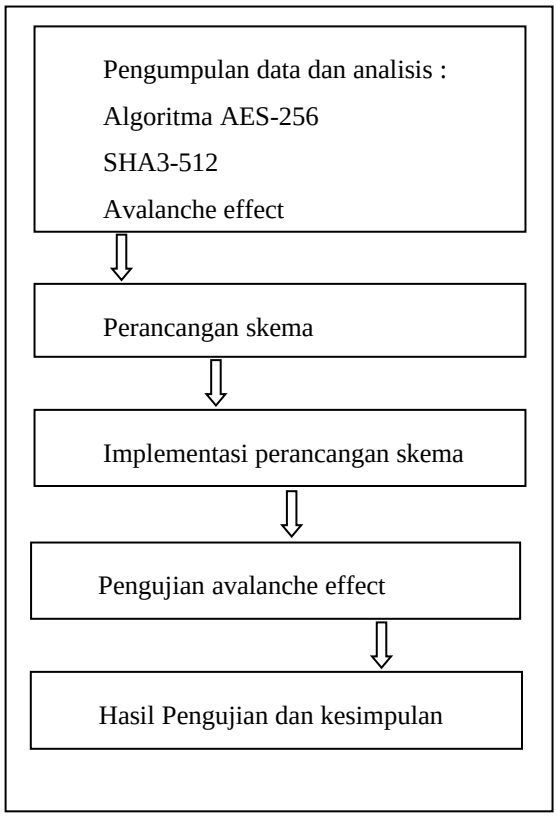
Pada tahun 2022 penelitian dengan mengimpelemtasikan AES untuk pengamanan data pada dokumen menghasilkan sebuah kesimpulan bahwa konsep algoritma *AES* telah berhasil untuk di implementasikan, namun tidak ada pengujian baik dari sisi waktu proses enkripsi dan dekripsi, juga tidak ada pengujian keamanan dari algoritma yang diguanakan (Azhari, M). Pada tahun 2023 penelitian yang menerapkan algoritma kriptografi SHA-256 dan AES-256 untuk mengamankan file pada PT Pelangi Sentral Kreasi menghasilkan sebuah kesimpulan bahwa konsep algoritma AES-256 dan SHA-256 berhasil diimplementasikan pada sistem yang dibangun, menampilkan waktu untuk proses enkripsi dan dekripsi pada file yang digunakan namun tidak disampaikan jenis dokumen yang diamankan, dan kekuatan hasil dari kombinasi algoritmanya (Dharmawan, A). Pada tahun 2023 penelitian dengan menerapkan metode kriptografi AES untuk mengamankan file dokumen menghasilkan sebuah kesimpulan bahwa algoritma AES berhasil dimplementasikan pada sistem di sebuah perusahaan. Namun tidak ada waktu enkripsi dan dekripsi file serta pengujian keacakan atau keamanan file yang di enkripsi (El Anwar).

Konsep algoritma kriptografi AES di terapkan pada dokumen digital, konsep pengamanan dokumen digital ini membuat pemilik dokumen lebih yakin jika dokumen yang disimpan dalam penyimpanan online akan lebih terjaga kerahasiaannya dan autentikasi pemilik dokumen dibutuhkan untuk bisa memebaca dokumennya.

Pada penelitian dengan judul "Penerapan Algoritma Kriptografi SHA-256 dan AES-256 untuk Pengamanan File Pada PT Pelangi Sentral Kreasi" konsep algoritma AES-256 dan SHA-256 berhasil diimplementasikan pada sistem yang dibangun, menampilkan waktu untuk proses enkripsi dan dekripsi pada file yang digunakan namun tidak disampaikan jenis dokumen yang diamankan, dan kekuatan hasil dari kombinasi algoritmanya, sehingga pada penelitian yang akan diajukan, penulis akan mengkombinasikan algoritma AES-256 dan SHA3-512 yang akan dibandingkan dengan algoritma AES-256 dan SHA-512 untuk menguji keacakan dari kombinasi kedua algoritma dengan menggunakan konsep *avalanche effect* dan akan membandingkan proses waktu enkripsi dekripsi pada dokumen berekstensi .pdf.

II. METODE PENELITIAN

Metodologi pada penelitian ini dapat diihat pada gambar 1.



Gambar 1 Metodologi Penelitian

Berdasarkan gambar 1, metodologi penelitian yang diterapkan pada peneliatian ini adalah sebagai berikut:

1. Pengumpulan data dan analisis

   Pada tahap ini, penulis memilih algoritma yang akan digunakan pada tanda tangan digital dan metode pengujian yang akan dilakukan pada penelitina ini. Algoritma yang digunakan yaitu :

   a. Algoritma AES

      AES adalah algoritma block cipher simetris yang dirancang untuk menawarkan tingkat keamanan yang tinggi dengan berbagai pilihan panjang kunci (Al-gohany, N.A.). AES memiliki panjang kunci 128-bit, 192-bit, dan 256-bit, yang masing-masing mempengaruhi jumlah putaran (rounds) dan struktur kunci. Pada AES-128, panjang kunci (Nk) adalah 4 kata (words), dengan setiap kata berukuran 32-bit, menghasilkan total panjang kunci sebesar 128-bit. Algoritma ini menggunakan ukuran blok 128-bit dan melibatkan 10 putaran (Nr) selama proses enkripsi.

      Selain parameter panjang kunci, AES juga melibatkan parameter lain yang penting, yaitu Nb (Number of Block Words). Nb merujuk pada jumlah kata dalam blok data yang diproses oleh algoritma. Dalam AES, ukuran blok selalu tetap 128-bit, yang setara dengan 4 kata (4 x 32-bit = 128-bit). Oleh karena itu, Nb untuk AES adalah 4, dan parameter ini menentukan struktur internal data yang diproses dalam algoritma. Dalam setiap putaran enkripsi, data yang dibagi menjadi 4 kolom (words) akan diproses melalui langkah-langkah transformasi yang melibatkan operasi SubBytes, ShiftRows, MixColumns, dan AddRoundKey. (Ariyus, D.).

Tabel 1 AES berdasarkan panjang kunci

Algoritma	Panjang kunci (bit)	Ukuran blok (bit)	Jumlah Putaran
AES-128	128	128	10
AES-192	192	128	12
AES-256	256	128	14

Algoritma Rijndael, yang digunakan dalam Advanced Encryption Standard (AES), memperlihatkan perbedaan signifikan dengan DES dalam pendekatan dan operasinya. Seperti DES, Rijndael juga menggabungkan langkah-langkah substitusi dan permutasi dengan mengoperasikan sejumlah putaran (cipher berulang). Kunci internal (Round key) yang berbeda digunakan dalam setiap putaran. Namun, perbedaan mendasar terletak pada orientasi operasi, di mana DES mengoperasikan dalam orientasi bit, sedangkan operasi pada Rijndael menggunakan byte, memudahkan dalam implementasi algoritma ke dalam *software dan hardware*. Algoritma Rijndael bekerja pada blok 128-bit dengan kunci 128-bit. Proses utama dalam algoritma ini, di luar

pembangkitan Round key, adalah sebagai berikut:

1. *AddRoundKe*y atau *Initial Round* Tahap pertama melibatkan operasi XOR antara *plaintext* sebagai *state* awal dengan *cipher key*. Tahap ini juga dikenal *sebagai Initial Round.*

2. Putaran (*Round*): Langkah ini terdiri dari sejumlah putaran, dengan jumlan putaran ditentukan oleh parameter Nr. Proses yang dilakukan pada setiap putaran adalah sebagai berikut:

a SubBytes adalah senuah proses dimana setiap byte pada *array state* diganti dengan byte yang sesuai dari sebuah tabel substitusi (S-box).

b *ShiftRows* merupakan proses dimana baris-baris *array state* digeser dengan cara tertentu, mempertahankan integritas data.

c *MixColumns* merupakan proses dimana data di masing-masing kolom *array state* diacak menggunakan operasi yang kompleks.

d AddRoundKey: Dilakukan XOR antara state saat ini dan Round key yang sesuai.

3. Putaran Terakhir (*Final Round*): Proses khusus dilakukan pada putaran terakhir, termasuk tahap *SubBytes*, *ShiftRows, dan AddRoundKey*.

Garis besar algoritma enkripsi Rinjdael versi AES-128 dapat dilihat pada gambar 2 yang memperlihatkan pendekatan lebih kompleks dan adaptif dibandingkan dengan DES, serta memanfaatkan orientasi byte untuk memfasilitasi implementasi yang efisien dalam berbagai platform perangkat lunak dan perangkat keras. Penelitian ini akan secara spesifik menggunakan AES-256 dalam proses enkripsi dan dekripsi sehingga akan memuat jumlah putaran hingga empat belas kali, tiga belas putaran pada proses putaran (round) dan satu kali pada putaran terkahir sehingga totalnya memiliki 14 putaran (round).

Gambar 2 Proses Enkripsi Pada Algortima Rijndael

Tahap *Subbytes transformations* bergantung pada *S-box* untuk menggantikan sebuah *byte* dalam *state* dengan *byte* lain. Berdasarkan prinsip *confusion* dan *diffusion* Shannon dalam perancangan algoritma kriptografi, tahap ini memiliki peran penting dalam meningkatkan keamanan. Pada Gambar 3 Proses *Subbyte Transformation* dilakukan dengan mengubah nilai heksadesimal menjadi heksadesimal dari S-box. Tabel S-box telah ditentukan atau distandarisasi.

Gambar 3 Proses Subbytes Transformations

Tahap transformasi *ShiftRows* adalah langkah penting dalam proses enkripsi AES

yang berfungsi untuk mencampurkan *byte-byte* dalam *state*. Proses ini dimulai dengan baris pertama dari *state* yang tidak mengalami perubahan. Pada baris kedua, dilakukan pergeseran melingkar ke kiri sejauh 1 *byte*. Baris ketiga mengalami pergeseran melingkar ke kiri sejauh 2 *byte*. Sementara itu, baris keempat mengalami pergeseran melingkar ke kiri sejauh 3 *byte*. Proses Transformasi *ShiftRows* dapat dilihat pada Gambar 4 *Proses ShiftRows Transformation*.

Gambar 4 Proses Shiftrows Transformations

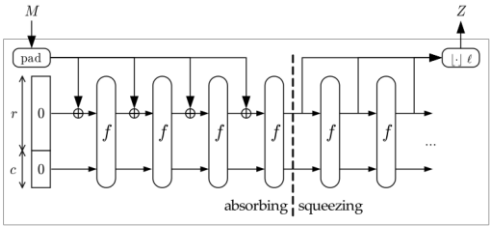
Sebaliknya, transformasi *InvShiftRows* adalah kebalikan dari *ShiftRows* dan diterapkan pada tahap dekripsi. Dalam *InvShiftRows*, pergeseran melingkar dilakukan ke arah yang berlawanan untuk ketiga baris terakhir. Baris kedua mengalami pergeseran melingkar ke kanan sejauh 1 *byte*, dan seterusnya untuk baris ketiga dan keempat dengan pergeseran yang sesuai. Transformasi ini memastikan bahwa data yang dienkripsi dengan *ShiftRows* dapat dikembalikan ke bentuk aslinya selama proses dekripsi.

Transformasi *MixColumns* adalah langkah penting dalam algoritma AES yang berfungsi untuk mencampurkan *byte-byte* dalam setiap kolom dari *state* secara individu. Proses ini bekerja dengan cara mengubah setiap *byte* dalam sebuah kolom menjadi nilai baru yang merupakan fungsi dari keempat *byte* dalam kolom tersebut. Gambar 5. menunjukkan transformasi dapat didefinisikan melalui perkalian matriks, di mana setiap elemen dalam matriks hasil produk merupakan jumlah dari hasil perkalian elemen-elemen pada baris dan kolom yang bersangkutan.

Gambar 5 Proses *Mixcolumns Transformation*

b. SHA3-512

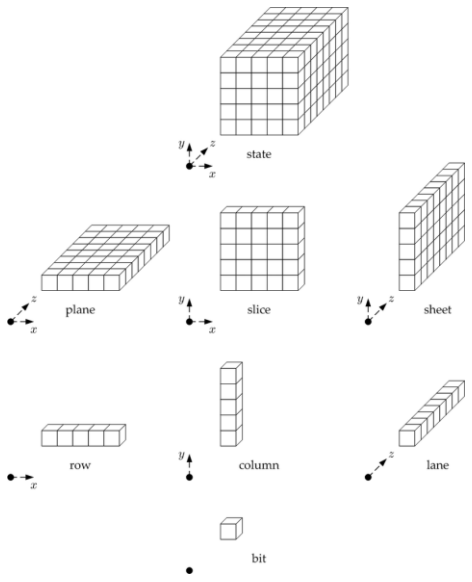
SHA3-512 adalah jenis algoritma dari fungsi *hash keccak*. Algoritma ini digunakanguna menjaga integritas pesan fungsi *hash keccak* ditunjukan pada Gambar 6.



Gambar 6 Fungsi Hash Keccak (J. Kelsey) (M.J. Dworkin)

Secara garis besar algoritma pada fungsi *hash keccak* ini terdiri dari *absorbing & sequeezing* (Kelsey) (Dworkin). Pada *absorbing*, masukkan akan di-*xor* kan dengan *bitrate* (*r)* lalu diteruskan ke fungsi *f* yang memiliki 5 tahapan operasi yaitu: *diffusion (θ), inter-slice dispersion (ρ), disturbing the horizontal / vertical alignment (π), non-linierity (X),* dan *break symetric (i).* Pada *Sequeezing* akan didapatkan luaran dimana penggabungan digunakan untuk menghasilkan nilai *hash* dengan panjang sama dengan total bit dari *capacity (c)* (J. Kelsey) (M.J. Dworkin).

State pada *keccak* adalah bit-bit yang dapat dilihat sebagai bit *array* dengan bentuk tiga dimensi (J. Kelsey) (M.J. Dworkin). Setiap sumbu dari *array* di representasikan dengan sumbu x, y dan z. *State* dari keccak ditunjukkan pada Gambar 7.



Gambar 7 State Pada Keccak (Kelsey) (Dworkin)

2. Perancangan skema

   Pada tahap ini, akan dibuat skema untuk melakukan proses penanda tanganan secara digital dan proses validasi bagi penerima sehingga integritas data dokumen bisa tetap terjaga dan penerima meyakini jika pesan berasal dari pengirim yang sah.

3. Implementasi perancangan skema algoritma

   Pada tahap ini, hasil dari perancangan skema akan di implementasikan dengan flow dari proses unggah dokumen dan unduh dokumen. Prototyping merupakan metode pengembangan perangat lunak yang berupa model fisik kerja sistem dan berfungsi sebagai versi awal dari sistem. Metode prototyping ini akan menghasilkan prototype sistem sebagai perantara pengembang dan pengguna agar dapat berinteraksi dalam proses kegiatan pengembangan sistem informasi (Purnomo, D). Model prototyping pada Gambar 8 setidaknya mempunyai 6 tahapan sebagai berikut: 1) Requirements Gathering and Analysis (Analisis Kebutuhan), Desain cepat, Bangun *prototype*, Evaluasi pengguna, Perbaikan *prototype*, Implementasi produk beserta pemeliharaan

Gambar 8 Metode *Prototyping*

4. Pengujian brute force attack

   Pada tahap ini hasil enkripsi akan diujikan dengan menggunakan metode brute force attack dimana akan dilihat celah bagi penyerang untuk mendapatkan isi dokumen pdf yang terenkripsi.

5. Hasil pengujian dan kesimpulan

   Pada tahap ini akan disampaikan hasil akhir dari penelitian ini.

III. ANALISIS DAN PERANCANGAN

Pada bab ini disampaikan perancangan skema untuk proses enkripsi dan dekripsi serta pengujian avalanche effect untuk melihat keacakan hasil enkripsi.

III.1 Skema unggah dokumen (enkripsi)

Skema dari penelitian ini terdiri pemilik dokumen dan sistem yang menyimpan dokumen dari pengguna yang sah. Gambar 9 merupakan proses Unggah (enkripsi).

Gambar 9 Skema Unggah (Enkripsi)

Pada skema unggah dokumen, pengguna bebas menentukan kunci sendiri untuk menyandikan dokumen dengan ketentuan Panjang kunci minimal 8 digit, dengan kombinasi huruf angka dan special karakter. Pengguna juga harus mengunggah dokumen pdf. Proses enkripsi akan terjadi pada sistem dimana isi dari dokumen pdf akan di enkripsi dengan kunci yang di inputkan pengguna, selanjutnya hasil hashing kunci dengan SHA3-512 akan disimpan pada basisdata sehingga harus mengetahui kunci asli untuk bisa melakukan dekripsi dokumen yang terenkripsi. Skema unggah (enkripsi) dokumen bisa dilihat pada Gambar 9.

III.2 Skema unduh dokumen (dekripsi)

Pada skema unduh dokumen, pemilik harus login terlebih dahulu ke dalam sistem lalu memilih dokumen yang ada pada menu list dokumen yang di unggah, lalu masukkan kunci dari dokumen yang akan diunggah. Saat kunci yang di masukkan sama dengan kunci dokumen yang diunggah maka isi dokumen pdf akan di ekstraksi lalu akan dilakukan dekripsi dengan kunci yang diinputkan dan setelah selesai proses dekripsi maka hasil dekripsi akan disimpan pada dokumen pdf sehingga dokumen pdf bisa diunduh dengan format teks asli yang bisa terbaca, jika tidak mengetahui kunci dan memaksa mengunduh dokumen tersebut maka isi dokumen akan teracak karena hasil enkripsi. Sehingga orang yang bukan pemilik yang sah tidak bisa membaca dokumen tersebut dengan mudah. Proses unduh dokumen (dekripsi) oleh pemilik bisa dilihat pada Gambar 10.

Gambar 10 Skema Unduh (Dekripsi)

III.3 Pengujian brute force attack

Skema yang diajukan pada pengujian ini adalah attacker mencoba membobol data dokumen dan mencoba melakukan dekripsi terhadap file pdf yang terenkripsi. Meskipun attacker sudah mendapatkan dokumen pdf, namun isi dokumen pdf tersebut masih terenkripsi dan attacker butuh mengetahui kunci dokumen tersebut dengan cara brute force attack kunci yang dicoba pad dokumen pdf atau mencoba mencari tahu kunci dengan membandingkan hasil hashing kunci yang di tebak dengan kunci yang di has kan pada dokumen.

Peluang penyerang untuk mendapatkan kunci dengan brute force adalah $1/2^{l}$ dimana l merupakan panjang kunci rahasia. Proses untuk menebak kunci rahasia penanda tangan bisa dilihat pada persamaan 1.

K, hashing_kunci
K = 1
while hashing kunci != h(K){

K = K + 1 } (1)

Jika penyerang mencoba menebak kunci dan cara melakukan brute force terhadap hasil enkripsi adalah $1/2^{l}$ dimana l merupakan panjang kunci rahasia yang bisa dilihat pada persamaan 2.

file_terenkripsi, k
k = 1
while error = tmp {
tmp = {file_terenkripsi}$_k$
k = k + 1 } (2)

Pengujian algoritma dilakukan pada 3 file pdf. File-file pdf tersebut bisa dilihat pada tabel 1.

Tabel 1. File Pdf Untuk Pengujian Algoritma

No	Nama file	Ukuran file	Jumlah kata
1	Large.pdf	28.13 KB	2976 kata
2	Medium.pdf	18.41 KB	1780 kata
3	Small.pdf	08.36 KB	450 kata

Pada tabel 1 bisa dilihat bahwa pengujian algoritma AES-256 dan SHA3-512 dengan 3 ukuran file pdf.

Pengujian waktu juga dilakukan untuk menghitung waktu proses enkripsi dan dekripsi algoritma SHA3-512 dan AES-256. Tabel 2 menunjukan waktu proses enkripsi dan dekripsi dalam satuan waktu mikroseken.

Tabel 2. Waktu Tanda Tangan Dan Validasi Dokumen

No	Dokumen pdf	Waktu enkripsi	Waktu dekripsi
1	Large	0.0070 µs	0.0070 µs
2	Medium	0.0060 µs	0.0063 µs
3	Small	0.0046 µs	0.0049 µs
4	Large	0.0050 µs	0.0052 µs
5	Medium	0.0040 µs	0.0057 µs
6	Small	0.0060 µs	0.0060 µs
7	Large	0.0070 µs	0.0052 µs
8	Medium	0.0040 µs	0.0041 µs
9	Small	0.0050 µs	0.0061 µs

Dari percobaan pada Tabel 1 dapat dilihat bahwa proses waktu enkripsi paling lama pada dokumen pdf large adalah 0.0070 µs smentara untuk prose dekripsi umumnya lebih lama dari pada proses enkripsi.

IV. KESIMPULAN

Tingkat keamanan yang ditawarkan oleh AES-256 dalam kombinasi dengan SHA3-512 terbukti memadai. Pengujian menunjukkan bahwa proses enkripsi dan dekripsi algoritma menggunakan kombinasi ini stabil dan dapat diandalkan, menjamin integritas dan kerahasiaan dokumen secara efektif. Kekuatan algoritma AES bergantung pada keacakan kunci, sehingga kunci yang digunakan ditingkatkan kecakannya dengan menggunakan fungsi hash SHA-512, meskipun butuh waktu yang lebih Panjang karena kunci untuk enkripsi di haskan terlebih dahulu namun dalam hal kecepatan pemrosesan, kombinasi algoritma ini menunjukkan efisiensi waktu yang baik, bahkan dengan ukuran *file* yang bervariasi. Hal ini menegaskan bahwa metode ini tidak hanya aman tetapi juga cepat, ini membuat kombinasi AES-128 dan SHA-512 sebagai solusi yang praktis untuk pengamanan dokumen berbasis *website*.

Pada penelitian selanjutnya diharapkan kombinasi ini bisa digunakan dalam perangkat internet of things atau mobile. Dengan ekstensi dokumen yang lebih beragam. Pada penelitian selanjutnya juga diharapkan menggunakan kombinasi lain dari fungsi hash dan algortima simetris atau asimetris.

REFERENSI

Stallings, W. (2013). “Cryptography and Network Security: Principles and Practice”. New Jersey: Prentice Hall Press.

Stallings, William. (2005). “Cryptography and Network Security Principles and Practices (4th ed.)”. Prentice Hall.

Dang, Quynh H. (2012). Federal Inf. Process. Stds. (NIST FIPS), National Institute of Standards and Technology, Gaithersburg, MD, [online], https://doi.org/10.6028/NIST.FIPS.180-4

B. Bhushan, G. Sahoo and A. K. Rai, "Man-in-the-middle attack in wireless and computer networking — A review," 2017 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA) (Fall), 2017, pp. 1-6, doi: 10.1109/ICACCAF.2017.8344724.

W. Diffie, M. Hellman, New directions in cryptography. IEEE Transactions on Information Theory, 22, (1976), pp. 644–654.

Secure Hash Standard (SHS). Federal Information Processing Standards Publication, FIPS PUB 202, 2015

Nasution, A. B. (2023). Implementasi Kriptografi dengan Metode Caesar Cipher untuk Mengamankan Data File di Javanetbeans. *Jurnal Pendidikan, Sains Dan Teknologi*, *2*(1), 17-21

I. G. Indra, "Peningkatan Pengamanan Data File Menggunakan Algoritma Kriptografi AES Dari Serangan Brute Force," *Jurnal Media Informatika*, vol. 4, no. 2, pp. 102–109, 2023.

Dharmawan, A., & Munandar, H. (2023). PENERAPAN ALGORITME KRIPTOGRAFI SHA-256 DAN AES-256 UNTUK PENGAMANAN FILE PADA PT PELANGI SENTRAL KREASI. *Prosiding Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)*, *2*(2), 186–195.

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.

El Anwar, Y., Habibi, R., & riza, N. (2022). PENERAPAN METODE KRIPTOGRAFI AES UNTUK MENGAMANKAN FILE DOKUMEN. *Jurnal Tekno Insentif*, *16*(2), 92-104.

Al-gohany, N. A., & Almotairi, S. (2019). Comparative Study of Database Security In Cloud Computing Using AES and DES Encryption Algorithms. Journal of Information Security and Cybercrimes Research, 2(1).

Ariyus, D., & Amikom, U. (n.d.). Pengantar Ilmu Kriptografi: Teori Analisis & Implementasi. Penerbit Andi.

Purnomo, D. (2017). Model prototyping pada pengembangan sistem informasi. *JIMP-Jurnal Informatika Merdeka Pasuruan*, *2*(2).

Kelsey, J., Change, S., & Perlner, R. (2016). *SHA-3 derived functions: cSHAKE, KMAC, TupleHash and ParallelHash*. https://doi.org/10.6028/NIST.SP.800-185

Dworkin, M. J. (2015). *SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions*. https://doi.org/10.6028/NIST.FIPS.202

Upadhyay, D., Gaikwad, N., Zaman, M., & Sampalli, S. (2022). Investigating the avalanche effect of various cryptographically secure Hash functions and Hash-based applications. *IEEE Access*, *10*, 112472-112486.