Jurnal EurekaMatika

Journal homepage: https://ejournal.upi.edu/index.php/JEM
doi: https://doi.org/10.17509/jem.v14i1.142

Pengamanan *File* Excel Berformula dengan Menggunakan Kriptografi *Advanced Encryption Standard*-256 Dan *Encoding* Base64

Kesha Mauluddy Rachman, Rini Marwati, Imam Nugraha Albania*

Program Studi Matematika, Universitas Pendidikan Indonesia
*Correspondence: E-mail: rinimarwati@upi.edu

A B S T R A K

Perkembangan teknologi membuat pengolahan data semakin banyak dilakukan secara digital. Microsoft Excel banyak digunakan karena memudahkan perhitungan melalui formula otomatis. Namun, isi dan formula dalam file Excel sering kali bersifat rahasia, sehingga perlu diamankan agar tidak diakses oleh pihak yang tidak berwenang. Penelitian ini bertujuan untuk mengimplementasikan algoritma kriptografi *Advanced Encryption Standard* (AES-256) dan *encoding* Base64 dalam pengamanan *file* Excel berformula. Proses pengamanan dilakukan dengan mengonversi data plaintext dan kunci ke format heksadesimal menggunakan tabel ASCII, diikuti oleh enkripsi menggunakan algoritma AES-256. Hasil enkripsi kemudian dikodekan ke format Base64, yang menghasilkan teks yang sulit dipahami oleh manusia tanpa proses dekripsi. Program untuk implementasi ini dikembangkan menggunakan bahasa pemrograman Python. Implementasi ini memberikan solusi praktis dalam pengamanan data pada *file* Excel.

© 2025 Kantor Jurnal dan Publikasi UPI

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima 10 Maret 2026
Direvisi 8 April 2026
Disetujui 13 Mei 2026
Tersedia online 15 Mei 2026
Dipublikasikan 15 Mei 2026

Kata Kunci:
AES-256,
Base64,
File excel,
Kriptografi,
Pengamanan data.

A B S T R A C T

The rapid development of technology today has digitilized various aspects of human life, including data processing. Using Microsoft Excel, data processing becomes easier as it leverages formulas for automated calculations. However, Excel formulas and file contents often contain sensitive information, posing a security risk if accessed by unauthorized parties. This research aims to implement the Advanced Encryption Standard (AES-256) cryptographic algorithm and Base64 encoding for securing Excel files containing formulas. The security process involves converting plaintext data and keys into hexadecimal format using the ASCII table, followed by encryption using the AES-256 algorithm. The encrypted output is then encoded in Base64, resulting in text that is unreadable to humans and difficult to analyze without the decryption process. A program to implement this process was developed using the Python programming language. This implementation provides a practical solution for securing sensitive data.

© 2026 Kantor Jurnal dan Publikasi UPI

Keywords:
AES-256,
Base64,
Cryptography,
Excel files,
Securing data.

1. PENDAHULUAN

Perkembangan teknologi informasi di era modern telah memunculkan kebutuhan akan aplikasi pengolahan data yang efektif, salah satu software yang dapat melakukannya adalah Microsoft Excel. Aplikasi ini digunakan secara luas untuk berbagai keperluan, mulai dari laporan keuangan hingga berbagai kebutuhan administrasi (Astuti *et al*., 2023). Salah satu fitur utamanya adalah formula, yang memungkinkan perhitungan otomatis. Karena Excel sering digunakan untuk keperluan penting yang sering kali mengandung informasi atau formula khusus, file tersebut berpotensi menimbulkan risiko keamanan apabila diakses oleh pihak yang tidak berwenang. Oleh karena itu, keamanan *file* Excel yang mengandung data sensitif perlu ditingkatkan untuk mencegah potensi kebocoran informasi.

Salah satu cara menjaga keamanan data dan formula dalam *file* Excel adalah dengan kriptografi, yang melindungi kerahasiaan dan keaslian data serta meningkatkan keamanannya (Rahman, 2019). Salah satu algoritma kriptografi yang bernama *Advanced Encryption Standard*-256 (AES-256), adalah solusi efektif untuk menjaga kerahasiaan data. AES-256 dipilih karena efisiensi dan keamanannya yang tinggi dalam menangani data kompleks seperti *file* Excel berformula (Nuari & Ratama, 2020). Penelitian oleh Mahajan dan Sachdeva (2013) mengonfirmasi bahwa AES lebih unggul dibandingkan algoritma lainnya seperti RSA dalam hal kecepatan dan keamanan

Namun, tantangan muncul ketika hasil enkripsi AES-256 yang berupa data biner harus disimpan atau dipertukarkan dalam format yang dapat dibaca dan tetap aman, seperti pada file Excel. Untuk mengatasi hal tersebut, encoding Base64 dapat digunakan sebagai solusi. *Encoding* Base64 sendiri merupakan metode yang mengubah data biner menjadi teks ASCII, sehingga mempermudah pengolahan dan penyimpanan data pada berbagai sistem. Menurut Azlin dkk. (2018), Base64 dapat mengonversi data besar menjadi teks yang lebih mudah ditangani tanpa kehilangan integritas data. Efendi *et al*. (2021) juga menunjukkan bahwa Base64 sering digunakan dalam pengamanan *file* multimedia, termasuk *file* video, karena fleksibilitasnya. Dalam konteks transmisi data, hasil penelitian oleh Halimi *et al*., (2024) menunjukkan bahwa kombinasi AES-256 dengan *encoding* Base64 memberikan perlindungan optimal dengan hasil enkripsi yang dapat disimpan dalam format teks yang kompatibel.

Berdasarkan temuan-temuan sebelumnya, dalam penelitian ini dirancang suatu aplikasi kriptografi yang menggabungkan AES-256 dengan Base64 untuk menciptakan sistem pengamanan *file* Excel berformula yang lebih aman dan efisien. Pendekatan ini diharapkan dapat memberikan solusi praktis dalam melindungi kerahasiaan dan integritas data pada *file* Excel, baik untuk penggunaan pribadi maupun institusional.

2. METODE

Metode yang digunakan dalam pembuatan aplikasi pengamanan *file* Excel berformula pada artikel ini adalah penggabungan algoritma AES-256 dan *encoding* Base64.

2.1 *Advanced Encryption Standard*-256

Algoritma Rijndael, yang dikembangkan oleh Joan Daemen dan Vincent Rijmen, merupakan standar algoritma kriptografi baru yang ditetapkan sebagai *Advanced Encryption Standard* (AES) oleh US-National Institute of Standards and Technology (NIST) pada November 2001 sebagai pengganti Data Encryption Standard (DES) yang sudah berhasil dibobol kuncinya. AES adalah algoritma kriptografi simetri yang berarti kunci pada proses enkripsi dan dekripsinya sama. AES memiliki tiga variasi panjang kunci, yaitu 128, 192, dan 256 bit, serta dikenal sebagai AES-128, AES-192, AES-256. Meskipun panjang kunci berbeda,

ukuran blok selalu tetap, yaitu 128 bit. Blok-blok data masukan dan kunci dioperasikan dalam bentuk array. Operasi AES dilakukan terhadap array of *byte* yang disebut state. Setiap state akan mengalami proses yang secara garis besar terdiri atas empat tahap sebagai berikut.

1. *AddRoundKey*

   Pada proses ini, subkey digabungkan dengan state. Dalam proses penggabungan ini digunakan operasi XOR untuk setiap *byte* dari subkey dengan *byte* dari state. Untuk seluruh tahap, subkey dibangkitkan dari kunci utama dengan menggunakan proses key schedule. Setiap subkey berukuran sama dengan state yang bersangkutan.
2. *SubBytes*

   Proses Sub*Byte*s merupakan operasi yang akan melakukan substitusi dengan cara mengganti setiap *byte* state dengan *byte* pada sebuah tabel yang dinamakan tabel S-Box. Suatu tabel S-Box terdiri atas 16 baris dan 16 kolom dengan masing-masing berukuran 1 *byte*.
3. *ShiftRows*

   Proses ini berfungsi untuk mengolah setiap baris dalam tabel state. Dalam proses ini, byte pada tiga baris terakhir (baris 1, 2, dan 3) akan digeser dengan jumlah pergeseran yang berbeda. Baris 1 akan digeser satu posisi, baris 2 akan digeser dua posisi, dan baris 3 akan digeser tiga posisi. Sementara itu, baris 0 tidak akan mengalami pergeseran.
4. *MixColumns*

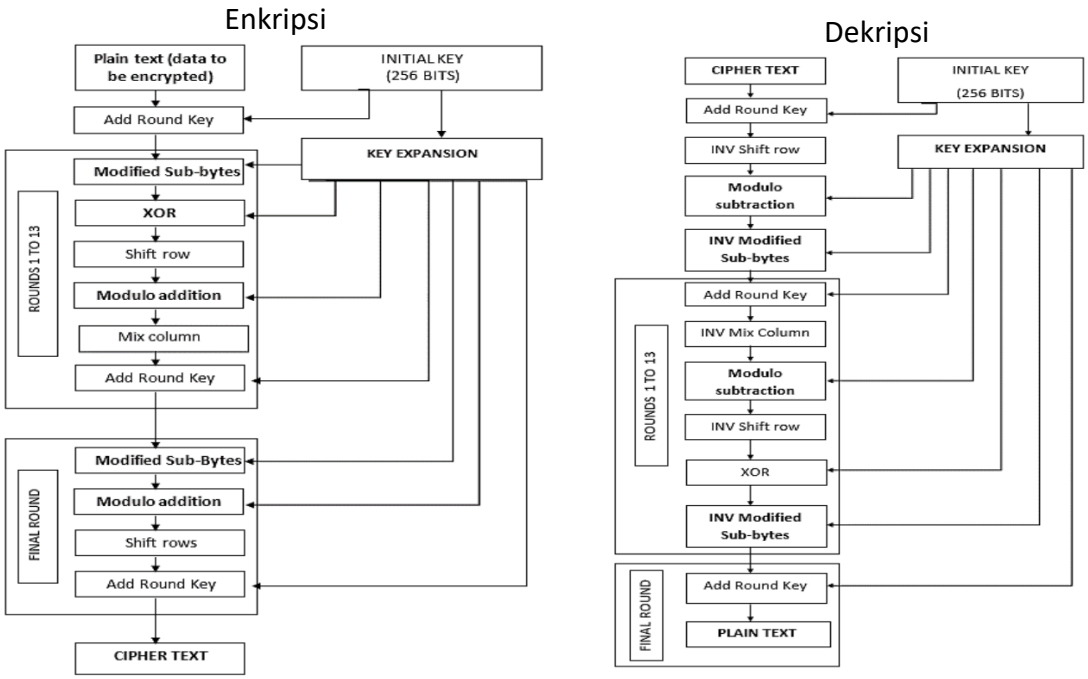
   Proses MixColumns akan beroperasi pada tiap kolom dari tabel state. Proses ini mengoperasikan 4 bytes dari setiap kolom tabel state dengan matriks *MixColumns* dalam AES.
5. *KeyExpansion*

   Untuk mengembangkan jumlah kunci dari kunci utama, digunakan proses key schedule. Proses ini mencakup beberapa operasi sebagai berikut.
   1) *Rot Word*, merupakan operasi perputaran setiap *bytes* dari kunci secara siklik.
   2) *Sub Bytes*, pada operasi ini 8-bit dari subkey disubstitusikan dengan nilai yang ada pada tabel dari S-Box.
   3) Operasi XOR dengan $w[i - N_k]$ yaitu *word* yang berada pada N_k sebelumnya.
   4) Operasi *Rcon*, merupakan operasi XOR dengan nilai pada Tabel Rcon sesuai dengan round yang dijalankan. Tabel *Rcon* dapat dilihat pada Tabel 1. Nilai-nilai dari Rcon kemudian akan di XOR dengan hasil operasi sebelumnya.

Tabel 1. Nilai *Rcon*

Round	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Rcon[]	01	02	04	08	10	20	40	80	1b	36
	00	00	00	00	00	00	00	00	00	00
	00	00	00	00	00	00	00	00	00	00
	00	00	00	00	00	00	00	00	00	00

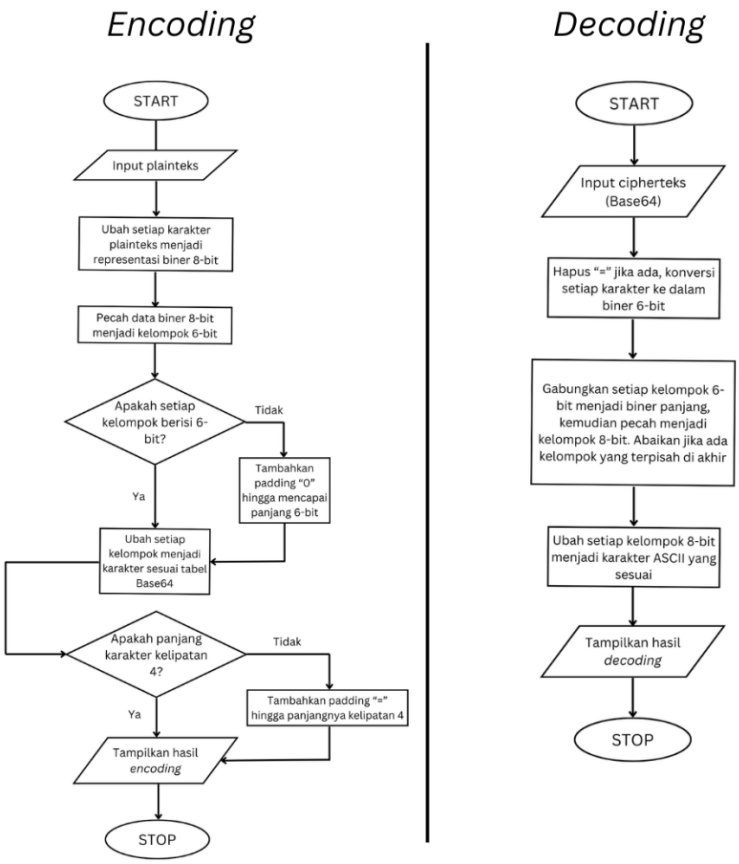
Proses dekripsi merupakan kebalikan dari enkripsi, di mana transformasi dilakukan secara terbalik dan diimplementasikan dalam arah yang berlawanan untuk mendapatkan cipher inversenya (Utami, *et al*., 2023). Proses ini berulang sebanyak 14 putaran. Skema enkripsi AES-256 yang dikemukakan oleh Manikandaprabhu dan Samreetha, (2024) dapat dilihat pada Gambar 1.



Gambar 1. Skema Enkripsi dan Dekripsi AES-256

2.2 Base64

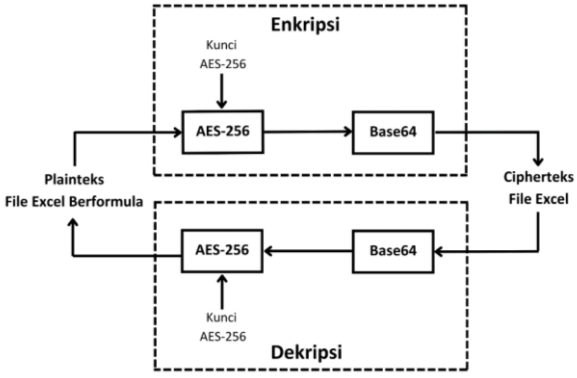
Base64 merupakan salah satu metode *encoding* dan *decoding* data ke dalam format ASCII, yang didasarkan pada bilangan basis 64. Algoritma ini berguna untuk mengonversi data biner menjadi teks. Algoritma ini akan menghasilkan data yang terdiri atas karakter A-Z, a-z, angka 0-9, serta dua karakter simbol "+" dan "/" (Azlin dkk., 2018). Skema encoding & decoding Base64 (Wen & Dang, 2016) dapat Pada Gambar 2 adalah skema dari proses *encoding* dan *decoding* Base64 yang dikemukakan oleh Wen & Dang (2016).



Gambar 2. Skema *Encoding* dan *Decoding* Base64

2.3 Pengembangan Model

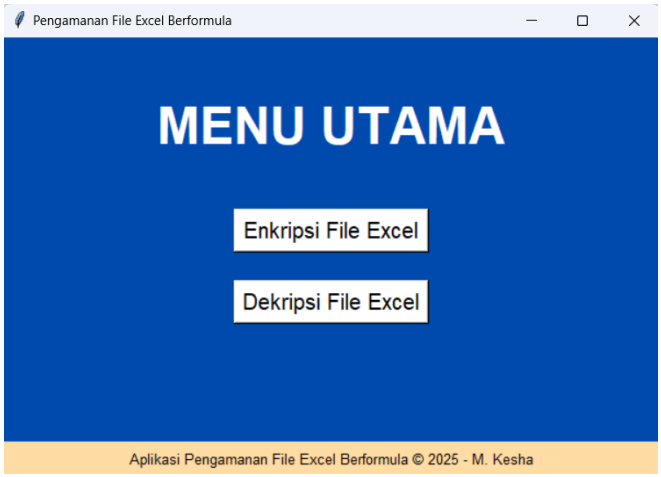
Penelitian ini menggabungkan algoritma enkripsi simetris AES-256 dan *encoding* Base64. AES-256 digunakan untuk mengamankan data dan formula dalam *file* Excel, sementara Base64 menyandikan hasil enkripsi ke format string ASCII yang kompatibel dan mudah disimpan. Kombinasi ini memastikan data tetap terlindungi meski *file* dipindahkan antar sistem. Hasil enkripsi disimpan dalam *file* Excel sebagai lapisan keamanan tambahan untuk mencegah akses tidak sah. Skema pengembangan model dapat dilihat pada Gambar 3, Enkripsi diterapkan pada plaintext berupa file Excel berformula dengan terlebih dulu menggunakan AES-256, kemudian encoding Base64, sehingga diperoleh cipherteks berupa file Excel. Dekripsi diterapkan pada ciphertext berupa file Excel dengan menggunakan Base64, dilanjutkan dengan AES-256, sehingga diperoleh kembali plainteks berupa file Excel berformula.



Gambar 3. Skema Pengembangan Model

3. HASIL DAN PEMBAHASAN

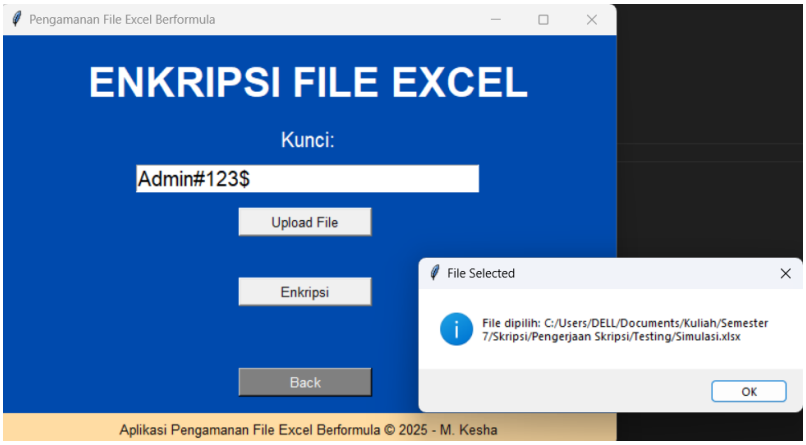
Program aplikasi untuk implementasi algoritma AES-256 dan *encoding* Base64 dikembangkan dengan mengonversi model matematis algoritma ke dalam bahasa pemrograman Python. Program ini dirancang untuk pengamanan *file* Excel berformula, dengan menggunakan Python versi 3.9.6 pada komputer yang memiliki spesifikasi sistem operasi Windows 11 64-bit, prosesor Intel Core i5-6300U, dan RAM 8 GB. Program yang telah dikonstruksi memiliki dua menu utama yaitu enkripsi dan dekripsi. Tampilan awal dari program ditunjukkan pada Gambar 4.



Gambar 4. Tampilan Awal Program

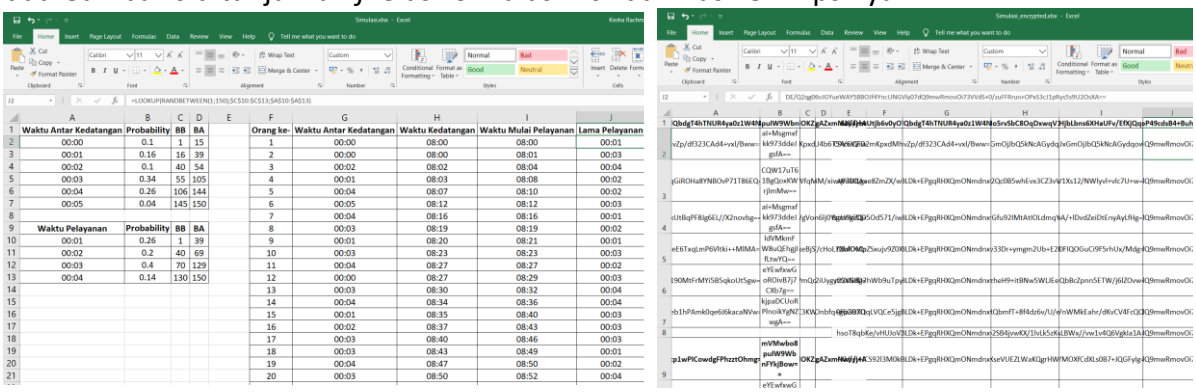
3.1 Enkripsi

Pada program enkripsi, pengguna menginput kunci berupa karakter ASCII dan memilih *file* Excel berformula mana yang akan dienkripsi. Ditunjukkan contoh *input* kunci "Admin#123$" dan *file* Excel Simulasi.xlsx yang akan dienkripsi pada Gambar 5.



Gambar 5. Tampilan Menu Enkripsi *File* Excel

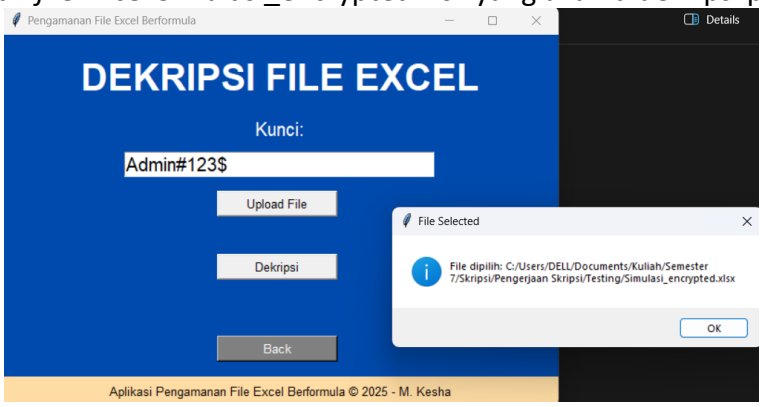
Pada Gambar 6 ditunjukkan *file* asli Simulasi.xlsx dan hasil enkripsinya.



Gambar 6. Contoh Hasil Enkripsi

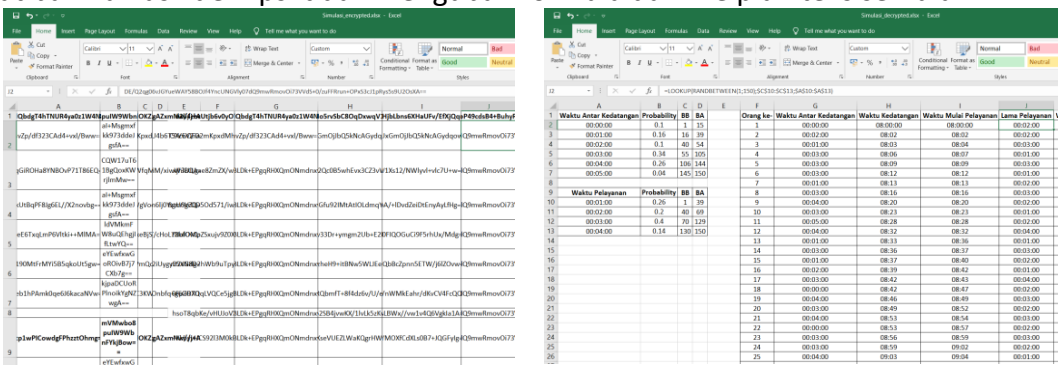
3.2 Dekripsi

Pada program dekripsi, pengguna menginput kunci yang sama seperti saat mengenkripsi dan memilih *file* Excel cipherteks yang akan didekripsi. Ditunjukkan contoh *input* kunci "Admin#123$" dan *file* Excel Simulasi_encrypted.xlsx yang akan didekripsi pada Gambar 7.



Gambar 7. Dekripsi *File* Excel

Pada Gambar 8 ditunjukkan *file* asli Simulasi.xlsx dan hasil enkripsinya. Pada Gambar 8 terlihat bahwa hasil dekripsi tidak mengubah formula dari file plainteks semula.



Gambar 8. Contoh Hasil Dekripsi

4. KESIMPULAN

Program pengamanan *file* Excel berformula dengan algoritma AES-256 dan *encoding* Base64 dirancang untuk melindungi teks dan formula dalam *file* Excel dengan tingkat keamanan tinggi. Proses enkripsi dimulai dengan mengubah teks dan formula menjadi byte yang tidak dapat dibaca tanpa kunci enkripsi menggunakan AES-256, kemudian hasil enkripsi

ini di*encoding* ke format string ASCII dengan Base64 agar dapat disimpan dan ditampilkan dalam format *file* Excel. Program yang dikembangkan dengan menggunakan Python ini dilengkapi dengan fitur enkripsi dan dekripsi, memungkinkan data yang terenkripsi dikembalikan ke bentuk aslinya dengan aman, sehingga kombinasi algoritma ini mampu menjaga kerahasiaan dan integritas data dalam *file* Excel secara efektif. Program aplikasi yang dikembangkan dalam penelitian ini belum mengakomodasi file Excel yang memuat gambar. Hal ini dapat menjadi kajian dalam penelitian berikutnya.

5. DAFTAR PUSTAKA

Astuti, E., Yunita, P., Tambunan, F., Wahyuni, F. S., & Setiyawati, R. I. (2023). Pelatihan pengenalan dan penerapan aplikasi komputer microsoft excel pada SMU Swasta Dharmawangsa Medan. *Abdikan: Jurnal Pengabdian Masyarakat Bidang Sains Dan Teknologi*, 2(1), 50-57.

Azlin, A., Musadat, F., & Nur, J. (2018). Aplikasi kriptografi keamanan data menggunakan algoritma base64. *Jurnal Informatika*, 7(2), 1-5.

Efendi, M., Sihombing, V., & Parulian, S. (2021). Implementation and use of base64 algorithm in video *file* security. *Sinkron: Jurnal dan Penelitian Teknik Informatika*, *6*(1), 243-247.

Halimi, A., Tholib, A., & Yaqin, M. A. (2024). Optimasi keamanan data penerimaan mahasiswa menggunakan aes-256, sha-256, dan base64. *JUSTIFY: Jurnal Sistem Informasi Ibrahimy*, 3(1), 38-45.

Mahajan, P., & Sachdeva, A. (2013). A study of encryption algorithms AES, DES and RSA for security. *Global Journal Of Computer Science And Technology, 13*(15), 15-22.

Manikandaprabhu, P., & Samreetha, M. (2024). A review of encryption and decryption of text using the AES algorithm. *International Journal of Scientific Research & Engineering Trends*, *10*(2). 400-404.

Nuari, R., & Ratama, N. (2020). Implementasi algoritma kriptografi AES (Advanced Encryption Standard) 128 bit untuk pengamanan dokumen shipping. *Journal Of Artificial Intelligence And Innovative Applications, 1*(2), 37-44.

Rahman, A. (2019). Perancangan aplikasi pengamanan *file* pada memory card handphone menggunakan algoritma kunci asimetris Elgamal. *JURIKOM (Jurnal Riset Komputer)*, *6*(5), 531-537.

Utami, W. C., Marwati, R., & Gozali, S. M. (2023). Penggabungan kriptografi Rivest Shamir Adleman (RSA) dan advanced encryption standard (AES) pada aplikasi pengirim e-mail. *Interval: Jurnal Ilmiah Matematika*, 3(2), 92-101.

Wen, S., & Dang, W. (2018). Research on base64 encoding algorithm and PHP implementation. In *2018 26th International Conference on Geoinformatics* (pp. 1-5). IEEE.