

# Rancang Bangun Aplikasi Keamanan Data File Inventaris Dengan Menggunakan Cloud Storage Dan Algoritma Camellia (Studi Kasus: Divisi Kesehatan Lingkungan Rsu Imelda Pekerja Indonesia)

*Design and Development of an Inventory File Security Application Using Cloud Storage and the Camellia Algorithm  
(Case Study: Environmental Health Division, RSU Imelda Pekerja Indonesia)*

**Setyo Hadi<sup>1</sup>, Ommi Alfina<sup>2</sup>**

<sup>1,2</sup>Prodi Informatika

Universitas Potensi Utama, Jl. K.L Yos Sudarso Km 6,5 No 3A, Tj.Mulia Medan, 20241, Indonesia  
setyohadi394@gmail.com, ny.aeroen@gmail.com

---

## ABSTRAK

Penelitian ini membahas pengembangan aplikasi enkripsi, dekripsi dengan menggunakan algoritma camellia dan penyimpanan berbasis *cloud storage* untuk menjaga file inventaris pada divisi kesehatan lingkungan RSU Imelda Pekerja Indonesia. Beberapa masalah yang dihadapi adalah file inventaris masih menggunakan penyimpanan fisik yaitu flashdisk, belum adanya keamanan pada file inventaris dari akses tidak sah. Solusi yang ditawarkan dengan menggunakan algoritma camellia untuk keamanan data file inventaris dan *cloud storage* untuk masalah penyimpanan data file. Tujuan dari penelitian ini menerapkan metode kriptografi untuk meningkatkan keamanan data inventaris agar terlindung dari akses tidak sah dan juga membangun sistem penyimpanan data inventaris berbasis *cloud* yang efisien, aman, dan dapat diakses oleh pengguna yang berwenang. Hasil dari pengujian menunjukkan bahwa aplikasi dapat melakukan enkripsi dan dekripsi menggunakan algoritma camellia dan dapat menyimpan file inventaris dengan baik kedalam *cloud storage*.

**Kata Kunci :** Enkripsi, Dekripsi, Camellia, *Cloud Storage*, Kriptografi

## ABSTRACT

*This research discusses the development of an encryption and decryption application using the Camellia algorithm and cloud-based storage to secure inventory files in the Environmental Health Division of RSU Imelda Pekerja Indonesia. Several issues have been identified, including the reliance on physical storage devices such as flash drives and the absence of security measures to prevent unauthorized access to inventory files. The proposed solution involves implementing the Camellia algorithm to enhance data security and utilizing cloud storage for efficient data management. The objective of this study is to apply cryptographic methods to improve the security of inventory data, protecting it from unauthorized access while establishing a secure, efficient, and accessible cloud-based storage system for authorized users. The results of the testing phase indicate that the application successfully performs encryption and decryption using the Camellia algorithm and effectively stores inventory files in cloud storage.*

**Keywords :** Encryption, Decryption, Camellia, *Cloud Storage*, Cryptography

---

Info Artikel :

Disubmit: 24 Februari 2025

Direview: 05 Maret 2025

Diterima : 29 April 2025

---

Copyright © 2025 – Journal UPU. All rights reserved

---

## 1. PENDAHULUAN

Keamanan data merupakan aspek krusial dalam sistem informasi, terlebih pada sektor pelayanan kesehatan yang menangani data penting dan sensitif. Di divisi kesehatan lingkungan RSU imelda pekerja indonesia, file-file inventaris saat ini masih disimpan secara manual menggunakan media flashdisk tanpa adanya perlindungan data yang memadai. Hal ini berisiko tinggi terhadap kehilangan data akibat kerusakan fisik, pencurian, atau penyalahgunaan informasi oleh pihak yang tidak berwenang. Selain keterbatasan media penyimpanan, belum adanya penerapan sistem enkripsi juga menjadi celah keamanan yang signifikan. Penyimpanan data di flashdik

## 2. TINJAUAN PUSTAKA

### A. Kriptografi

Kriptografi merupakan ilmu dan seni untuk menjaga pesan agar aman. "*Crypto*" berarti "*secret*" (rahasia) dan "*graphy*" yang berarti "*writing*" (tulisan) [2]. Kriptografi telah berkembang dari sekadar teknik menulis pesan rahasia menjadi fondasi utama dalam sistem keamanan digital modern. Dalam penerapannya, kriptografimemungkinkan informasi dikodekan sedemikian rupa sehingga hanya pihak yang memiliki kunci tertentu yang dapat membaca isi pesan tersebut. Proses ini tidak hanya menjaga kerahasiaan pesan, tetapi juga memastikan bahwa informasi tidak dapat diubah sembarangan serta dapat memverifikasi identitas pengirim. Dengan demikian, kriptografi tidak hanya sebatas "tulisan rahasia", tetapi juga menjadi alat penting dalam menjaga integritas dan keaslian data di era komunikasi digital.

### B. Algoritma Camellia

Algoritma Camellia merupakan salah satu algoritma kriptografi modren yang termasuk ke dalam kunci simetrik dan juga blokcipher dengan jumlah blok 128 bit [3]. Algoritma ini menggunakan kunci dengan panjang 128, 192, atau 256 bit. Camellia dirancang oleh para ahli dari Mitsubishi dan NTT Jepang, serta telah diakui secara internasional dan termasuk dalam standar kriptografi seperti ISO/IEC. Salah satu keunggulan utama Camellia adalah struktur Feistelyang digunakannya, yang dikombinasikan dengan berbagai teknik pengacakan seperti rotasi bit, substitusi, dan penggunaan tabel S-box, sehingga mampu menjaga data tetap aman dari serangan kriptanalisis.

### C. Enkripsi

Enkripsi adalah sebuah proses untuk mengamankan data dengan menyembunyikannya atau mengubah data menjadi bentuk yang tidak dapat dibaca atau dimengerti [4]. Proses ini melibatkan penggunaan algoritma tertentu dan kunci enkripsi agar hanya pihak yang memiliki kunci yang sesuai dapat mengembalikan data tersebut ke bentuk aslinya. Enkripsi sangat penting dalam dunia digital saat ini, terutama untuk melindungi komunikasi, transaksi keuangan, dan penyimpanan data pribadi. Dengan enkripsi, informasi yang dikirimkan melalui jaringan terbuka tetap terlindungi dari akses atau manipulasi oleh pihak yang tidak diizinkan.

### D. Dekripsi

Dekripsi diartikan sebagai proses mengubah data yang telah dienkripsi menjadi data aslinya sehingga dapat dibaca atau dimengerti kembali [4]. Proses ini hanya dapat dilakukan oleh pihak yang memiliki kunci atau informasi yang sesuai dengan metode enkripsi yang digunakan sebelumnya. Dekripsi berperan penting dalam memastikan bahwa data yang diterima masih utuh dan dapat dibaca dengan benar oleh penerima yang sah. Dalam sistem keamanan informasi, dekripsi menjadi bagian yang tak terpisahkan karena memungkinkan pengguna yang berwenang untuk mengakses data yang telah diamankan, tanpa membuka peluang bagi pihak yang tidak memiliki izin untuk membacanya.

### E. Cloud Storage

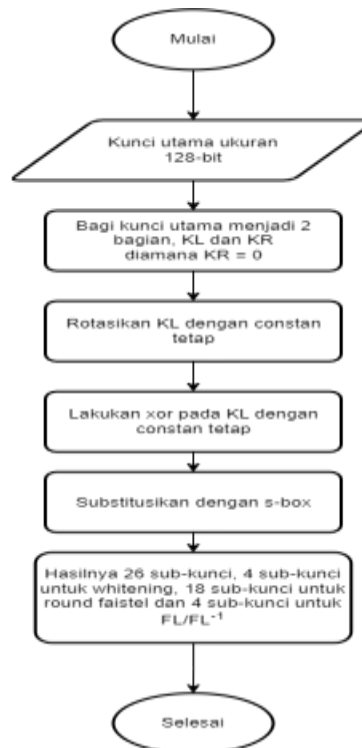
*Cloud storage* merupakan layanan penyimpanan file berbasis dengan jaringan internet dimana file yang disimpan dapat dikelola dari berbagai tempat selama pengguna dapat terhubung dengan cloud storage

melalui internet [5]. Teknologi ini menawarkan fleksibilitas karena data tidak lagi terbatas pada satu perangkat fisik, melainkan tersimpan di server milik penyedia layanan yang dapat diakses kapan saja dan dari mana saja. Selain itu, cloud storage juga mendukung kolaborasi jarak jauh, pencadangan otomatis, serta penghematan ruang penyimpanan lokal. Keamanan data menjadi prioritas utama dalam layanan ini, dengan penerapan enkripsi dan kontrol akses guna melindungi informasi dari ancaman yang tidak diinginkan.

### 3. METODE

#### A. Penjadwalan kunci Algoritma Camellia

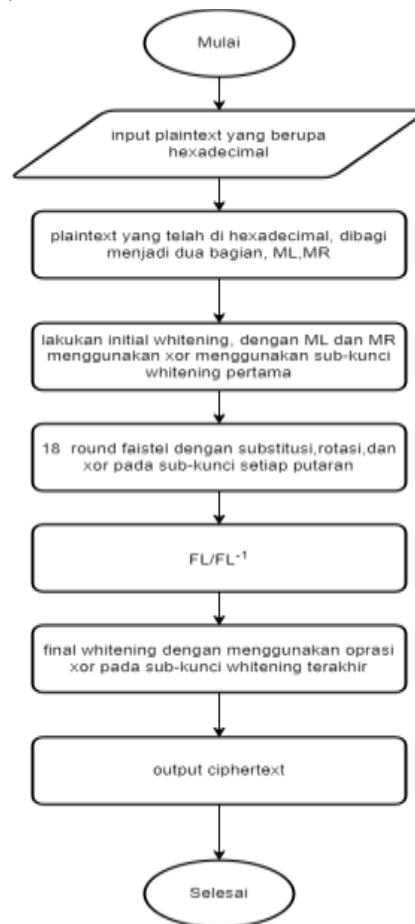
Berdasarkan *flowchart* pada Gambar 1, proses penjadwalan kunci pada algoritma Camellia untuk panjang kunci 128-bit dimulai dengan membagi kunci utama menjadi dua bagian, yaitu KL dan KR. Pada jenis kunci 128-bit, bagian KR tidak digunakan dan diisi dengan nilai nol. Tahapan selanjutnya adalah pembentukan nilai KA dan KB melalui serangkaian operasi transformasi, termasuk fungsi FL,  $FL^{-1}$ , serta fungsi F yang menggunakan tabel substitusi (S-box). Nilai KA dan KB ini menjadi landasan dalam pembangkitan sub-kunci utama. Sub-kunci yang dihasilkan dikelompokkan ke dalam beberapa jenis, antara lain sub-kunci untuk proses enkripsi dan dekripsi utama (kw, k, dan kl), sub-kunci untuk fungsi FL dan  $FL^{-1}$  (ke), serta sub-kunci untuk proses whitening. Seluruh sub-kunci tersebut diperoleh melalui operasi rotasi bit terhadap nilai-nilai KA, KB, KL, dan KR yang telah dibentuk sebelumnya. Proses rotasi ini dirancang secara hati-hati guna mencegah terjadinya kelemahan kriptografi, seperti pola sub-kunci yang berulang atau korelasi yang tinggi antar sub-kunci, yang dapat dimanfaatkan oleh pihak yang tidak berwenang.



Gambar 1. Flowchart Penjadwalan Kunci Camellia

#### B. Enkripsi Algoritma Camellia

Gambar 2 menggambarkan alur proses enkripsi pada algoritma Camellia. Proses ini dimulai dengan membagi blok plainteks berukuran 128-bit menjadi dua bagian yang masing-masing berukuran 64-bit, yaitu bagian kiri (L) dan kanan (R). Sebelum memasuki tahapan utama enkripsi, terlebih dahulu dilakukan proses *whitening* dengan menggunakan sub-kunci awal (kw). Proses ini bertujuan untuk menambahkan lapisan awal perlindungan dengan menggabungkan plainteks dan kunci melalui operasi XOR. Setelah proses *whitening*, data kemudian diproses melalui 18 ronde utama untuk kunci sepanjang 128-bit, atau 24. Setiap ronde mengikuti struktur Feistel dan menerapkan fungsi F yang memanfaatkan tabel substitusi (S-box), operasi rotasi bit, serta operasi logika. Fungsi F berperan penting dalam menciptakan difusi dan non-linieritas yang dibutuhkan untuk meningkatkan kompleksitas hasil enkripsi dan memperkuat ketahanan terhadap serangan kriptografi. Pada ronde ke-6 dan ke-12 (serta ronde ke-18 untuk kunci yang lebih panjang), disisipkan fungsi FL dan  $FL^{-1}$ . Fungsi FL merupakan transformasi linier yang mengandalkan operasi logika seperti AND, OR, dan rotasi bit, sedangkan  $FL^{-1}$  merupakan invers dari FL. Penyisipan fungsi-fungsi ini dilakukan secara bergantian pada masing-masing jalur data dan bertujuan untuk memperkuat struktur algoritma dari potensi serangan yang mengeksplorasi pola atau simetri internal. Setelah seluruh ronde selesai dijalankan, dilakukan proses *whitening* terakhir menggunakan sub-kunci yang berbeda dari awal. Tahapan ini berfungsi untuk menyamarkan hasil akhir dari proses enkripsi dan memastikan bahwa *ciphertext* yang dihasilkan benar-benar tidak menunjukkan hubungan langsung dengan plainteks maupun kunci yang digunakan. Hasil akhirnya adalah *ciphertext* sepanjang 128-bit yang siap dikirim atau disimpan secara aman.

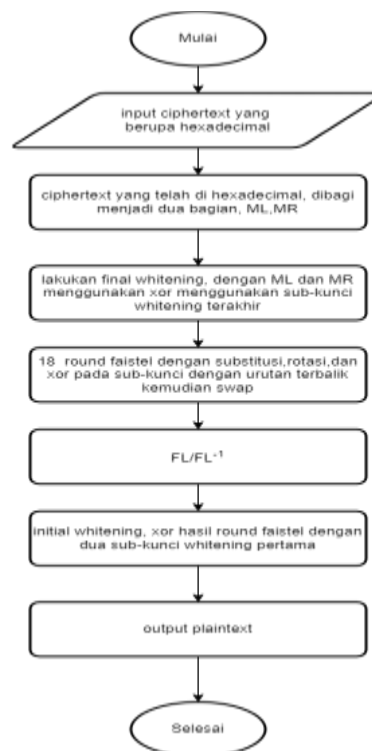


Gambar 2. Flowchart Enkripsi Camellia

### C. Dekripsi Algoritma Camellia

Gambar 3 memperlihatkan alur proses dekripsi pada algoritma Camellia, yang secara prinsip serupa dengan proses enkripsi namun dijalankan secara terbalik. Dekripsi dimulai dengan membagi blok ciphertext berukuran 128-bit menjadi dua bagian masing-masing 64-bit, yang disebut bagian kiri (Left) dan kanan (Right). Sebelum memasuki ronde utama, dilakukan proses *whitening* awal menggunakan sub-kunci khusus dekripsi ( $k_w$ ). Langkah ini bertujuan untuk menghapus efek dari *whitening* pada saat enkripsi. Ciphertext selanjutnya diproses melalui sejumlah ronde utama dengan urutan yang dibalik dari proses enkripsi, untuk kunci sepanjang 128-bit, terdapat 18 ronde. Setiap ronde menggunakan struktur Feistel dan fungsi F, yang bersifat simetris, sehingga algoritma dekripsi tetap dapat menggunakan struktur yang sama dengan enkripsi, hanya urutan sub-kuncinya yang dibalik. Pada ronde tertentu, yaitu ronde ke-6 dan ke-12 (serta ronde ke-18 untuk kunci yang lebih panjang), disisipkan kembali fungsi FL dan  $FL^{-1}$ . Dalam proses dekripsi, fungsi-fungsi ini diterapkan dalam urutan yang berlawanan dibandingkan saat enkripsi. Meski demikian, fungsi dan prinsip kerjanya tetap sama, yaitu berperan sebagai transformasi linier yang memperkuat ketahanan terhadap serangan kriptanalisis. Setelah semua ronde selesai dilalui, proses

diakhiri dengan *whitening* akhir menggunakan sub-kunci yang berbeda. Langkah ini bertujuan untuk memastikan bahwa hasil dekripsi berupa plaintext yang sesuai dengan input asli sebelum dienkripsi. Hasil akhirnya adalah blok data 128-bit yang telah dikembalikan ke bentuk semula secara utuh.



Gambar 3. Flowchart Dekripsi Camellia

## 4. HASIL DAN PEMBAHASAN

### A. Proses Enkripsi

Contoh:

Sampel *plaintext* = data rahasia tyo

$M_{(128)} = 6461746120726168617369612074796F$   $M_L = 6461746120726168.$

$M_R = 617369612074796F.$

$L_0 = M_L \text{ Xor } Kw1.$

$= 6461746120726168 \wedge 6B65792D72616861.$

$= 0F040D4C52130909.$

$R_0 = M_R \text{ Xor } Kw2.$

$= 617369612074796F \wedge 7369612D68616469 = 121A084C48151D06.$

1). Round  $\rightarrow 1$

$L_1 = R_0 \text{ Xor } F(L_0, k1).$

$R_1 = L_0.$

$L_1 = 121A084C48151D06 \wedge F(0F040D4C52130909, 0B1149E121134BEB).$

$F(L_0, k1) = 0F040D4C52130909 \wedge 0B1149E121134BEB.$

$F(L_0, k1) = 041544AD730042E2.$

$= P(S1(04)) = 179.$

$= P(S4(15)) = 220.$

$= P(S3(44)) = 253.$

$= P(S2(AD)) = 254.$

$= P(S4(73)) = 172.$

$= P(S3(00)) = 56.$

$= P(S2(42)) = 53.$

$= P(S1(E2)) = 185.$

$z'1 = z1 \wedge z3 \wedge z4 \wedge z6 \wedge z7 \wedge z8.$

$z'1 = 185 \wedge 56 \wedge 172 \wedge 253 \wedge 220 \wedge 179 = 191.$

$z'2 = z1 \wedge z2 \wedge z4 \wedge z5 \wedge z7 \wedge z8.$

$z'2 = 185 \wedge 53 \wedge 172 \wedge 254 \wedge 220 \wedge 179 = 177.$

$z'3 = z1 \wedge z2 \wedge z3 \wedge z5 \wedge z6 \wedge z8.$

$z'3 = 185 \wedge 53 \wedge 56 \wedge 254 \wedge 220 \wedge 179 = 37.$

$z'4 = z2 \wedge z3 \wedge z4 \wedge z5 \wedge z6 \wedge z7.$

$z'4 = 53 \wedge 56 \wedge 172 \wedge 254 \wedge 253 \wedge 220 = 126.$

$z'5 = z1 \wedge z2 \wedge z6 \wedge z7 \wedge z8.$

$z'5 = 185 \wedge 53 \wedge 253 \wedge 220 \wedge 179 = 30.$

$z'6 = z2 \wedge z3 \wedge z5 \wedge z7 \wedge z8.$

$z'6 = 53 \wedge 56 \wedge 254 \wedge 220 \wedge 179 = 156.$

$z'7 = z3 \wedge z4 \wedge z5 \wedge z6 \wedge z8.$

$z'7 = 56 \wedge 172 \wedge 254 \wedge 253 \wedge 179 = 36.$

$z'8 = z1 \wedge z4 \wedge z5 \wedge z6 \wedge z7.$

$z'8 = 179 \wedge 172 \wedge 254 \wedge 253 \wedge 220 = 192.$

$z' = BFB1257E1E9C24C0.$

$L_1 = 121A084C48151D06 \wedge BFB1257E1E9C24C0$

$L_1 = ADAB2D32568939C6.$

$R_1 = 0F040D4C52130909.$

2). Round  $\rightarrow 2$

$L_2 = R_1 \text{ Xor } F(L_1, k2).$

$R_2 = L_1.$

$L_2 = 0F040D4C52130909 \wedge F(ADAB2D32568939C6, 65E8250E7E1554E9).$

$F(L_1, k2) = 0F040D4C52130909 \wedge 0B1149E121134BEB.$

$$\begin{aligned}F(L_1, k2) &= C843083C289C6D2F. \\&= P(S1(C8)) = 41. \\&= P(S4(43)) = 98. \\&= P(S3(08)) = 118.\end{aligned}$$

$$\begin{aligned}&= P(S2(3C)) = 162. \\&= P(S4(28)) = 223. \\&= P(S3(9C)) = 132. \\&= P(S2(6D)) = 48. \\&= P(S1(2F)) = 26.\end{aligned}$$

$$\begin{aligned}z'1 &= z1 \wedge z3 \wedge z4 \wedge z6 \wedge z7 \wedge z8. \\z'1 &= 26 \wedge 132 \wedge 223 \wedge 118 \wedge 98 \wedge 41 = 124. \\z'2 &= z1 \wedge z2 \wedge z4 \wedge z5 \wedge z7 \wedge z8. \\z'2 &= 26 \wedge 48 \wedge 223 \wedge 162 \wedge 98 \wedge 41 = 28. \\z'3 &= z1 \wedge z2 \wedge z3 \wedge z5 \wedge z6 \wedge z8. \\z'3 &= 26 \wedge 48 \wedge 132 \wedge 162 \wedge 98 \wedge 41 = 71. \\z'4 &= z2 \wedge z3 \wedge z4 \wedge z5 \wedge z6 \wedge z7. \\z'4 &= 48 \wedge 132 \wedge 223 \wedge 162 \wedge 118 \wedge 98 = 221. \\z'5 &= z1 \wedge z2 \wedge z6 \wedge z7 \wedge z8. \\z'5 &= 26 \wedge 48 \wedge 118 \wedge 98 \wedge 41 = 23. \\z'6 &= z2 \wedge z3 \wedge z5 \wedge z7 \wedge z8. \\z'6 &= 48 \wedge 132 \wedge 162 \wedge 98 \wedge 41 = 93. \\z'7 &= z3 \wedge z4 \wedge z5 \wedge z6 \wedge z8. \\z'7 &= 132 \wedge 223 \wedge 162 \wedge 118 \wedge 41 = 166. \\z'8 &= z1 \wedge z4 \wedge z5 \wedge z6 \wedge z7. \\z'8 &= 26 \wedge 223 \wedge 162 \wedge 118 \wedge 98 = 115. \\z' &= 7C1C47DD175DA638. \\L_2 &= 0F040D4C52130909 \wedge 7C1C47DD175DA638 \\L_2 &= 7C1C47DD175DA673 \\R_2 &= ADAB2D32568939C6.\end{aligned}$$

3). Round  $\rightarrow$  3

$$L_3 = R_2 \text{ Xor } F(L_2, k3).$$

$$R_3 = L_2.$$

$$L_3 = ADAB2D32568939C6 \wedge F(7C1C47DD175DA673, C9E52DC95A875422).$$

$$F(L_2, k3) = 7C1C47DD175DA673 \wedge C9E52DC95A875422.$$

$$F(L_2, k3) = B5F96A144DDAF251.$$

$$\begin{aligned}&= P(S1(B5)) = 55. \\&= P(S4(F9)) = 98. \\&= P(S3(6A)) = 78. \\&= P(S2(14)) = 138. \\&= P(S4(4D)) = 230. \\&= P(S3(DA)) = 191. \\&= P(S2(F2)) = 167. \\&= P(S1(51)) = 76.\end{aligned}$$

$$\begin{aligned}z'1 &= z1 \wedge z3 \wedge z4 \wedge z6 \wedge z7 \wedge z8. \\z'1 &= 76 \wedge 191 \wedge 230 \wedge 78 \wedge 98 \wedge 55 = 14. \\z'2 &= z1 \wedge z2 \wedge z4 \wedge z5 \wedge z7 \wedge z8.\end{aligned}$$

$$z'2 = 76 \wedge 167 \wedge 230 \wedge 138 \wedge 98 \wedge 55 = 210.$$

$$z'3 = z1 \wedge z2 \wedge z3 \wedge z5 \wedge z6 \wedge z8.$$

$$z'3 = 76 \wedge 167 \wedge 191 \wedge 138 \wedge 78 \wedge 55 = 167.$$

$$z'4 = z2 \wedge z3 \wedge z4 \wedge z5 \wedge z6 \wedge z7.$$

$$z'4 = 167 \wedge 191 \wedge 230 \wedge 138 \wedge 78 \wedge 98 = 88.$$

$$z'5 = z1 \wedge z2 \wedge z6 \wedge z7 \wedge z8.$$

$$z'5 = 76 \wedge 167 \wedge 78 \wedge 98 \wedge 55 = 240.$$

$$z'6 = z2 \wedge z3 \wedge z5 \wedge z7 \wedge z8.$$

$$z'6 = 167 \wedge 191 \wedge 230 \wedge 98 \wedge 55 = 171.$$

$$z'7 = z3 \wedge z4 \wedge z5 \wedge z6 \wedge z8.$$

$$z'7 = 191 \wedge 230 \wedge 138 \wedge 78 \wedge 55 = 170.$$

$$z'8 = z1 \wedge z4 \wedge z5 \wedge z6 \wedge z7.$$

$$z'8 = 76 \wedge 230 \wedge 138 \wedge 78 \wedge 98 = 12.$$

$$z' = 8E652A633811750C.$$

$$L_3 = ADAB2D32568939C6 \wedge 8E652A633811750C.$$

$$L_3 = 23CE07516E984CCA.$$

$$R_3 = 7C1C47DD175DA673$$

langkah round dilakukan sampai 18 round, setelah cipertext di dapat dalam bentuk hexadesimal, kemudian

ciphertext dikonvermasi dalam karakter yaitu, datY ra—asic tyl.

#### A. Proses Dekripsi

Contoh:

Sampel *ciphertext* = datY ra—asic tyl.

$$C_{(128)} = 6461749F20726197617369632074796C.$$

$$C_{(128)} = 6461749F20726197617369632074796C \text{ XOR } (KW3 \parallel KW4).$$

$$C_{(128)} = 6461749F20726197617369632074796 \wedge (A512EC764AB819F5 \parallel 9E315CA7CB847A43).$$

$$C_{(128)} = C17398E96ACA7862FF4235C4EBF0032F.$$

1). Round  $\rightarrow$  1

$$R_{18} = C17398E96ACA7862.$$

$$L_{18} = FF4235C4EBF0032F.$$

$$R_{17} = L_{18} \text{ Xor } F(R_{18}, k18).$$

$$L_{17} = R_{18}$$

$$R_{17} = FF4235C4EBF0032F \text{ Xor } F(C17398E96ACA7862, 8C5B12AC7DA983EF).$$

$$F(R_{18}, k18) = C17398E96ACA7862 \wedge 8C5B12AC7DA983EF.$$

$$F(R_{18}, k18) = 73192768965980C0$$

$$= P(S1(73)) = 178.$$

$$= P(S4(19)) = 57.$$

$$= P(S3(27)) = 103.$$

$$= P(S2(68)) = 100.$$

$$= P(S4(96)) = 197.$$

$$= P(S3(59)) = 219.$$

$$= P(S2(80)) = 85.$$

$$= P(S1(C0)) = 233.$$

$$z'1 = z1 \wedge z3 \wedge z4 \wedge z6 \wedge z7 \wedge z8.$$

$$z'1 = 233 \wedge 219 \wedge 197 \wedge 103 \wedge 57 \wedge 178 = 27.$$

$$z'2 = z1 \wedge z2 \wedge z4 \wedge z5 \wedge z7 \wedge z8.$$

$$z'2 = 233 \wedge 85 \wedge 197 \wedge 100 \wedge 57 \wedge 178 = 150.$$

$$z'3 = z1 \wedge z2 \wedge z3 \wedge z5 \wedge z6 \wedge z8.$$

$$z'3 = 233 \wedge 85 \wedge 219 \wedge 100 \wedge 103 \wedge 178 = 100.$$

$$z'4 = z2 \wedge z3 \wedge z4 \wedge z5 \wedge z6 \wedge z7.$$

$$z'4 = 85 \wedge 219 \wedge 197 \wedge 100 \wedge 103 \wedge 57 = 113.$$

$$z'5 = z1 \wedge z2 \wedge z6 \wedge z7 \wedge z8.$$

$$z'5 = 233 \wedge 85 \wedge 103 \wedge 57 \wedge 178 = 80.$$

$$z'6 = z2 \wedge z3 \wedge z5 \wedge z7 \wedge z8.$$

$$z'6 = 85 \wedge 219 \wedge 100 \wedge 57 \wedge 178 = 97.$$

$$z'7 = z3 \wedge z4 \wedge z5 \wedge z6 \wedge z8.$$

$$z'7 = 219 \wedge 197 \wedge 100 \wedge 103 \wedge 178 = 175.$$

$$z'8 = z1 \wedge z4 \wedge z5 \wedge z6 \wedge z7.$$

$$z'8 = 233 \wedge 197 \wedge 100 \wedge 103 \wedge 57 = 22.$$

$$z' = 1B9664715061AF16.$$

$$R_{17} = FF4235C4EBF0032F \wedge 1B9664715061AF16.$$

$$R_{17} = E4D451B5BB91AC39.$$

$$L_{17} = C17398E96ACA7862.$$

2). Round  $\rightarrow$  2

$$R_{16} = L_{17} \text{ Xor } F(R_{17}, k17)$$

$$L_{16} = R_{17}$$

$$R_{16} = C17398E96ACA7862 \text{ Xor } F(E4D451B5BB91AC39, 95BE8727A29E684D).$$

$$F(R_{17}, k17) = E4D451B5BB91AC39 \wedge 95BE8727A29E684D.$$

$$F(R_{17}, k17) = 716AD692190FC474.$$

$$= P(S1(71)) = 178.$$

$$= P(S4(6A)) = 57.$$

$$= P(S3(D6)) = 103.$$

$$= P(S2(92)) = 100.$$

$$= P(S4(19)) = 197.$$

$$= P(S3(0F)) = 219.$$

$$= P(S2(C4)) = 85.$$

$$= P(S1(74)) = 233.$$

$$z'1 = z1 \wedge z3 \wedge z4 \wedge z6 \wedge z7 \wedge z8$$

$$z'1 = 195 \wedge 160 \wedge 57 \wedge 184 \wedge 231 \wedge 68 = 65$$

$$z'2 = z1 \wedge z2 \wedge z4 \wedge z5 \wedge z7 \wedge z8$$

$$z'2 = 195 \wedge 63 \wedge 57 \wedge 100 \wedge 231 \wedge 68 = 102$$

$$z'3 = z1 \wedge z2 \wedge z3 \wedge z5 \wedge z6 \wedge z8$$

$$z'3 = 195 \wedge 63 \wedge 160 \wedge 100 \wedge 184 \wedge 68 = 160$$

$$z'4 = z2 \wedge z3 \wedge z4 \wedge z5 \wedge z6 \wedge z7$$

$$z'4 = 63 \wedge 160 \wedge 57 \wedge 100 \wedge 184 \wedge 231 = 249$$

$$z'5 = z1 \wedge z2 \wedge z6 \wedge z7 \wedge z8$$

$$z'5 = 195 \wedge 63 \wedge 184 \wedge 231 \wedge 68 = 231$$

$$z'6 = z2 \wedge z3 \wedge z5 \wedge z7 \wedge z8$$

$$z'6 = 63 \wedge 160 \wedge 100 \wedge 231 \wedge 68 = 60$$

$$z'7 = z3 \wedge z4 \wedge z5 \wedge z6 \wedge z8$$

$$z'7 = 160 \wedge 57 \wedge 100 \wedge 184 \wedge 68 = 101$$

$$z'8 = z1 \wedge z4 \wedge z5 \wedge z6 \wedge z7$$

$$z'8 = 195 \wedge 57 \wedge 100 \wedge 184 \wedge 231 = 165$$

$$z' = 4166A0F9E73C65A5$$

$$R_{16} = C17398E96ACA7862 \wedge 4166A0F9E73C65A5.$$

$$R_{16} = 801538108DF61DC7.$$

$$L_{16} = E4D451B5BB91AC39.$$

3). Round  $\rightarrow$  3

$$R_{15} = L_{16} \text{ Xor } F(R_{16}, k_{16})$$

$$L_{15} = R_{16}$$

$$R_{15} = E4D451B5BB91AC39 \text{ Xor } F(801538108DF61DC7, 731DA837B529C24F)$$

$$F(R_{16}, k_{16}) = 801538108DF61DC7 \wedge 731DA837B529C24F.$$

$$F(R_{16}, k_{16}) = F308902738DFDF88.$$

$$= P(S1(F3)) = 123.$$

$$= P(S4(08)) = 35.$$

$$= P(S3(90)) = 8.$$

$$= P(S2(27)) = 157.$$

$$= P(S4(38)) = 254.$$

$$= P(S3(DF)) = 125.$$

$$= P(S2(DF)) = 245.$$

$$= P(S1(88)) = 84.$$

$$z'1 = z1 \wedge z3 \wedge z4 \wedge z6 \wedge z7 \wedge z8.$$

$$z'1 = 84 \wedge 125 \wedge 254 \wedge 8 \wedge 35 \wedge 123 = 135.$$

$$z'2 = z1 \wedge z2 \wedge z4 \wedge z5 \wedge z7 \wedge z8.$$

$$z'2 = 84 \wedge 245 \wedge 254 \wedge 157 \wedge 35 \wedge 123 = 154.$$

$$z'3 = z1 \wedge z2 \wedge z3 \wedge z5 \wedge z6 \wedge z8$$

$$z'3 = 84 \wedge 245 \wedge 125 \wedge 157 \wedge 8 \wedge 123 = 50.$$

$$z'4 = z2 \wedge z3 \wedge z4 \wedge z5 \wedge z6 \wedge z7.$$

$$z'4 = 245 \wedge 125 \wedge 254 \wedge 157 \wedge 8 \wedge 35 = 192.$$

$$z'5 = z1 \wedge z2 \wedge z6 \wedge z7 \wedge z8.$$

$$z'5 = 84 \wedge 245 \wedge 8 \wedge 35 \wedge 123 = 210.$$

$$z'6 = z2 \wedge z3 \wedge z5 \wedge z7 \wedge z8.$$

$$z'6 = 245 \wedge 125 \wedge 157 \wedge 35 \wedge 123 = 77.$$

$$z'7 = z3 \wedge z4 \wedge z5 \wedge z6 \wedge z8.$$

$$z'7 = 125 \wedge 254 \wedge 157 \wedge 8 \wedge 123 = 109.$$

$$z'8 = z1 \wedge z4 \wedge z5 \wedge z6 \wedge z7.$$

$$z'8 = 84 \wedge 254 \wedge 157 \wedge 8 \wedge 35 = 28.$$

$$z' = 879A32C0D24D6D1C$$

$$R_{15} = E4D451B5BB91AC39 \wedge 879A32C0D24D6D1C.$$

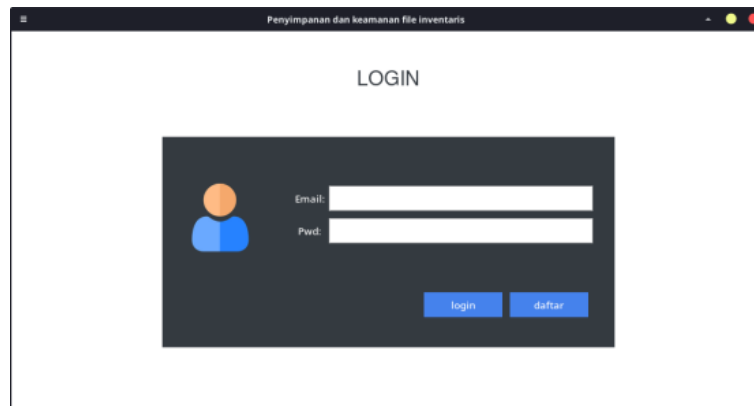
$$R_{15} = 634E637569DCC125.$$

$$L_{15} = 801538108DF61DC7.$$

proses dekripsi dilakukan sampai dengan round 18, setelah hasil dari plaintext di dapat, plaintext hexadesimal di konversikan ke dalam bentuk karakter menjadi data rahasia tyo.

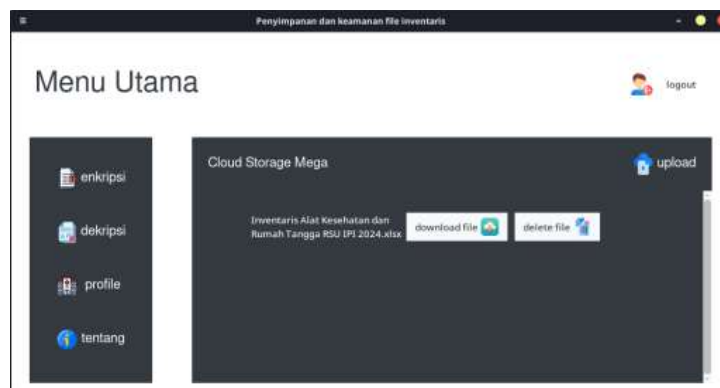
## B. Pengujian

Pengguna atau *user* harus melakukan proses autentifikasi terlebih dahulu dengan memasukan email ke entry box untuk email, kemudian masukan password ke entry box untuk password, email atau password harus sudah di daftarkan di *Mega Cloud Storage* dengan cara menekan tombol daftar, saat menekan tombol daftar *user* atau pengguna akan di alihkan ke website dari *cloud storage* mega, untuk masuk kedalam menu utama aplikasi tekan tombol login. Gambar 4 adalah tampilan dari *form login*



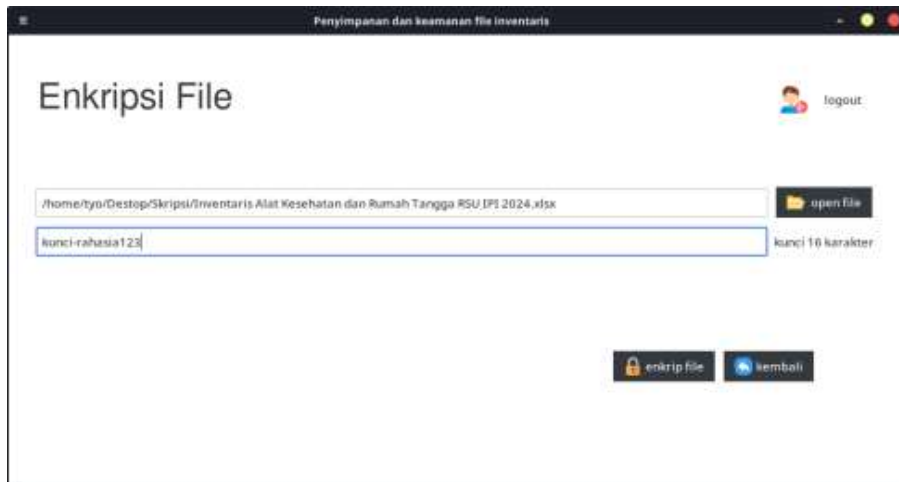
**Gambar 4.** Tampilan *form login*

Jika password dan email benar, maka pengguna atau *user* akan ditampilkan menu utama dari aplikasi, yang memiliki beberapa komponen, tombol dengan logo *person* dengan tulisan *logout* untuk menghapus autentifikasi kemudian menampilkan menu logi, untuk gambar *cloud* dengan tulisan *upload* digunakan untuk mengupload atau mengunggah file kedalam *cloud storage* mega, untuk tombol. *download* file digunakan untuk mendownload file yang ada dalam *cloud storage* mega, sedangkan untuk tombol *delete* file digunakan untuk menghapus file yang ada dalam *cloud storage* mega, untuk enkripsi, dekripsi, profile, tentang adalah sub menu dari menu utama, jika diklik akan membawa *user* atau pengguna ke menu enkripsi, dekripsi, profile, tentang. Gambar 5 adalah tampilan dari menu utama aplikasi



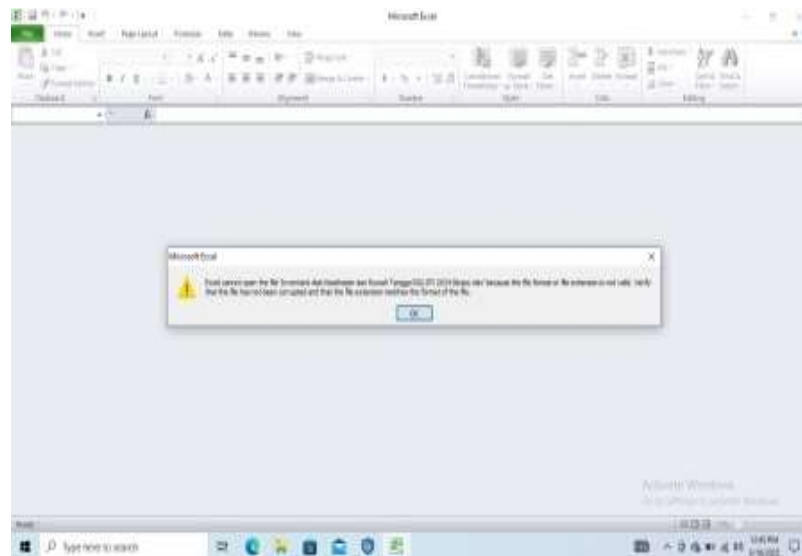
**Gambar 5.** Tampilan Menu Utama

Tampilan untuk menu enkripsi terdiri dari beberapa komponen, tombol *open* file untuk mengambil file inventaris yang berformat *xlsx* kemudia *path* file akan ditampilkan di entry box disamping tombol *open* file, *user* atau pengguna juga bisa memasukan jalur atau *path* dari file inventaris tanpa harus menekan tombol *open* file dengan entry box, selanjutnya *user* atau pengguna memasukan kunci untuk proses enkripsi dengan panjang 16 karakter, kemudian tombol *enkrip* file digunakan untuk proses enkripsi file inventaris dan menyimpan kunci untuk proses dekripsi, untuk kembali ke menu utama maka tekan tombol *kembali*. Gambar 6 adalah tampilan enkripsi aplikasi.



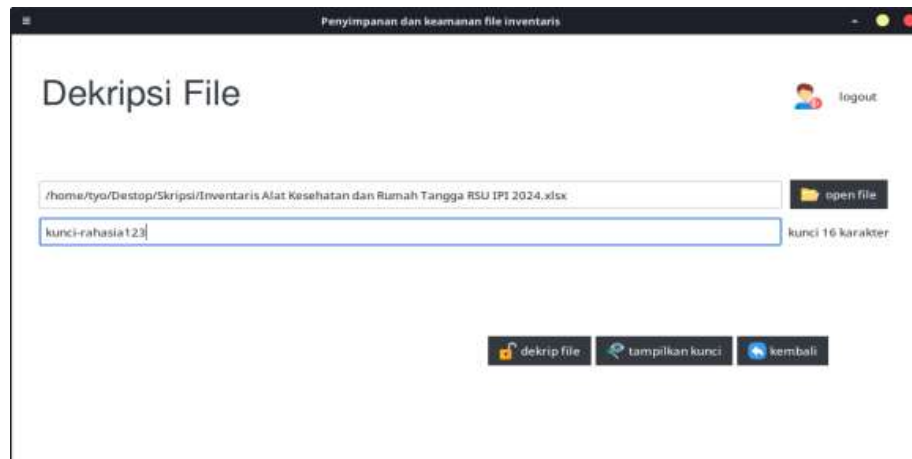
**Gambar 6.** Tampilan Menu Enkripsi

File yang telah selesai dienkripsi tidak dapat dibuka di dalam microsoft excel. Gambar 7 adalah hasil enkripsi file inventaris.



**Gambar 7.** Hasil Enkripsi File Inventaris

Tampilan untuk menu dekripsi terdiri dari beberapa komponen, tombol open file untuk mengambil file inventaris yang sudah di enkripsi berformat xlsx, kemudia *path* file akan ditampilkan di entry box disamping tombol open file, *user* atau pengguna juga dapat memasukan jalur atau *path* file inventaris tanpa harus menekan tombol open file dengan entry box, selanjutnya *user* atau pengguna memasukan kunci untuk proses dekripsi dengan panjang 16 karakter, jika *user* atau pengguna lupa kunci untuk melakukan dekripsi maka tekan tombol tampilkan kunci, untuk kembali ke menu utama maka tekan tombol kembali. Gambar 8 adalah tampilan dari menu dekripsi.



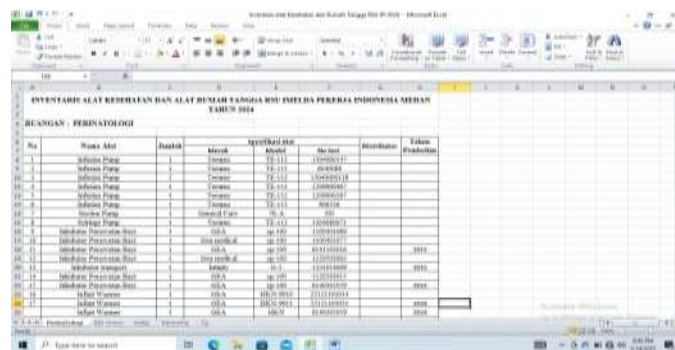
**Gambar 8.** Tampilan Dekripsi

Jika Anda lupa kunci enkripsi yang digunakan sebelumnya, Anda dapat menggunakan fitur ini untuk melihat kembali kunci yang telah disimpan sebelumnya. Kunci ini diperlukan untuk melakukan proses dekripsi dan mengembalikan data ke bentuk semula. Gambar 9. adalah penyimpanan kunci untuk proses dekripsi.



**Gambar 9.** Menu Penyimpanan Kunci

File yang telah di dekripsi akan kembali ke awal sebelum proses enkripsi, dimana file inventaris yang berformat xls berisi 4 sheet di dalamnya. Gambar 10. adalah hasil dari dekripsi.



| ID | Nama       | Jumlah | Satuan | Keterangan              |
|----|------------|--------|--------|-------------------------|
| 01 | Buku       | 1      | buah   | untuk referensi         |
| 02 | Kertas     | 1      | lembar | untuk menulis           |
| 03 | Pensil     | 1      | buah   | untuk menulis           |
| 04 | Penggaris  | 1      | buah   | untuk mengukur          |
| 05 | Kalkulator | 1      | buah   | untuk menghitung        |
| 06 | Komputer   | 1      | buah   | untuk pengolahan data   |
| 07 | Printer    | 1      | buah   | untuk mencetak          |
| 08 | Mouse      | 1      | buah   | untuk mengontrol kursor |
| 09 | Keyboard   | 1      | buah   | untuk memasukkan data   |

Gambar 10. Hasil Dekripsi

## 5. KESIMPULAN

Berdasarkan penelitian yang dilakukan oleh penulis, maka penulis mengambil beberapa kesimpulan sebagai berikut:

1. Bahwa penggunaan algoritma kriptografi Camellia mampu meningkatkan keamanan data inventaris dengan melakukan proses enkripsi sebelum data disimpan ke cloud. Algoritma Camellia memberikan perlindungan yang kuat terhadap akses tidak sah dan serangan kriptografi karena kemampuannya menangani kunci 128-bit, 192-bit, dan 256-bit. Selain itu, pemanfaatan teknologi cloud storage sebagai media penyimpanan data inventaris dapat mengatasi kelemahan dari penyimpanan manual seperti flashdisk, yang rentan terhadap kerusakan fisik dan kehilangan. Dengan penyimpanan terpusat melalui cloud, data menjadi lebih mudah diakses oleh pengguna yang berwenang dan lebih terlindungi dari kehilangan permanen.
2. Menggunakan algoritma Camellia yang memiliki tingkat keamanan tinggi dan tahan terhadap berbagai teknik cryptanalysis.
3. Menyimpan data inventaris secara online di cloud storage (MEGA), menghindari risiko kehilangan akibat kerusakan atau kehilangan perangkat fisik.
4. Implementasi kunci enkripsi masih terbatas pada 128-bit.
5. Mengembangkan dukungan untuk panjang kunci Camellia lainnya, yaitu 192-bit dan 256-bit.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Ibu Ommi Alfina, M.Kom, selaku dosen pembimbing yang telah dengan sabar membimbing, memberikan arahan, serta motivasi dalam penyusunan jurnal ini. Segala masukan dan ilmu yang diberikan sangat berarti dalam proses penyusunan hingga selesainya tulisan ini. Tanpa bimbingan dan dukungan beliau, jurnal ini tidak akan terselesaikan dengan baik.

## REFERENSI

- [1] Kumiawan, D., Afyenni, R., & Hidavat, R. (2018). Implementasi Algoritma AES dalam Mengenkripsi Berkas Terintegrasi dengan Layanan Cloud Storage Berbasis Android. *Prosiding SISFOTEK*, 2(1), 237-245.
- [2] Haryono, W. (2024). Teori Kriptografi dan Aplikasi.
- [3] Yanti, N. R., & Zebua, T. (2020). Implementasi Algoritma Camellia Pada Penyandian Record Database. *Journal of Computer System and Informatics (JoSYC)*, 1(3), 107-116.
- [4] Nugraha, S. N. (2024). Penerapan Algoritma Kriptografi Elgamal Pada Aplikasi Pengamanan Pesan Berbasis Website. *Jurnal Informatika dan Teknik Elektro Terapan*, 12(3).

- 
- [5] Tantowi, L., & Wijayanti, L. (2023). Peluang dan Tantangan Penyimpanan Cloud Storage Pada Dokumen Digital. *Shaut Al-Maktabah: Jurnal Perpustakaan, Arsip dan Dokumentasi*, 15(1), 118-131.
- [6] Matsui, M., Nakajima, J., & Tokita, S. (2004). A Description of the Camellia Encryption Algorithm (RFC 3713). Internet Engineering Task Force (IETF). Retrieved from <https://www.rfc-editor.org/rfc/rfc3713>.
- [7] Ani, H., & Nasution, S. D. (2020). Implementasi Algoritma Camellia Dalam Penyandian Pesan SMS Pada Telepon Seluler. *Journal of Information System Research (JOSH)*, 2(1), 61-70.
- [8] Wibowo, Z. D., Adam, I. F., & Saputra, W. A. (2020). Implementasi Algoritma Camellia Untuk Keamanan Citra Medis Pada Sistem Radiologi Berbasis Web. *Jurnal Informatika Polinema*, 6(4), 1-8.
- [9] Zainul, Z., & Romadhan, N. H. (2023). Cloud Storage Sebagai Pengganti Arsip Manual Dalam Penunjang Aktifitas Sehari-Hari. *Kohesi: Jurnal Sains dan Teknologi*, 1(6), 20-30.
- [10] Maharani, F. D. M. (2020). Implementasi Algoritma Camellia Pada Penyandian Citra USG. *Journal of Information System Research (JOSH)*, 2(1), 14-25.
- [11] Irwandi, H. (2017). *Implementasi Hybrid Cryptosystem Algoritma Elgamal dan Algoritma Camellia dalam Aplikasi Email* (Doctoral dissertation).
- [12] Hidayati, L. N., Fitriana, G. F., & Adam, I. F. (2021). Perbandingan Keacakan Citra Enkripsi Algoritma AES dan Camelia Uji NPCR dan UACI. *JURIKOM (Jurnal Riset Komputer)*, 8(6), 274.
- [13] Mulyani, S., Styorini, W., & Khabzli, W. (2016). Sistem Keamanan Aplikasi Chatting Menggunakan Algoritma Camelia. *Jurnal ELEMENTER (Elektro dan Mesin Terapan)*, 2(2), 42-49.
- [14] Malau, E. P. (2019). Pengamanan Aplikasi Menggunakan Algoritma Camelia. *Jurnal Informatika Kaputama (JIK)*, 3(1), 44-50.
- [15] Admi, A. (2017). CRYPTOGRAPHIC RANDOMNESS TESTING PADA ALGORITMA BLOCK CIPHER CAMELLIA MENGGUNAKAN UJI COVERAGE.