# IMPLEMENTASI ALGORITMA ENKRIPSI RSA DAN KOMPRESI LZW PADA DATABASE NASABAH DI PT. CENTRAL CAPITAL FUTURES

**Arief Fahrizon**[1)], **Muhammad Ainur Rony**[2)]
Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur
Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260
E-mail : arieffahrizon@gmail.com[1)], ainur.rony@gmail.com[2)]

***Abstrak***

*PT. Central Capital Futures menjaga baik data para nasabahnya agar tidak terjadi kesalahan atau kebocoran data pribadi maupun kecurangan dana dalam bertransaksi berbagai produk yang didagangkan. Namun data tidak benar-benar aman jika hanya disimpan dalam bentuk data yang dapat dibaca oleh manusia, dibutuhkan enkripsi agar data tidak lagi dapat dibaca oleh manusia, dan dibutuhkan dekripsi agar data bisa diterjemahkan kembali ke bahasa yang mudah dimengerti, dan juga dekripsi ditujukan hanya untuk orang tertentu yang bersangkutan. Maka dari itu selain untuk tujuan penulisan Tugas Akhir ini penulis juga membantu mengembangkan aplikasi perusahaan untuk pihak nasabah dengan bantuan teknik kriptografi. Kriptografi yang digunakan dalam aplikasi tersebut menggunakan metode RSA (Rivest-Shamir-Adleman) dan kompresi LZW (Lempel-ZivWelch). Aplikasi ini juga menggunakan perhitungan matematika yang cukup rumit disertai dengan private key dan public ke) dengan bertujuan dapat mengamankan database yang terenkripsi sehingga akan menyulitkan hacker untuk menembus. Berdasarkan pengujian program, telah disimpulkan bahwa program ini mudah untuk digunakan oleh user atau nasabah itu sendiri, data yang dikirim dan diterima melalui aplikasi ini aman dan terjaga kerahasiaannya karena sudah melalui proses enkripsi terlebih dahulu dan didekripsi kemudian. Penulis mengambil kesimpulan dengan adanya proses pengamanan ini dapat memecahkan masalah keamanan komunikasi data pada PT. Central Capital Futures.*

**Kata Kunci :** Kriptografi, Enkripsi, RSA, LZW, Database, Nasabah

## 1. PENDAHULUAN

Bertukar informasi ataupun data memberikan keuntungan serta dampak positif dalam kemudahan berinteraksi dengan apapun, namun juga memiliki dampak negatif, yaitu kejahatan pencurian data. Maka dari itu keamanan data yang tersimpan sangat penting dijaga keamanan dan orisinalitas dari berbagai ancaman yang berupa pengaksesan, perubahan, dan pengrusakan data oleh pihak yang tidak bertanggung jawab.

PT. Central Capital Futures mempunyai database para nasabah-nasabah yang melakukan investasi di perusahaan ini yang berisi biodata lengkap serta akun trading yang dimiliki. Sering kali *record* yang tersimpan di dalam basis data masih sama seperti teks yang ditampilkan sebagai informasi bagi pengguna akhir atau end-user. Hal ini dapat mempermudah orang lain mengetahui isi dari database tersebut.

Untuk mengatasi masalah-masalah di atas maka diperlukan cara untuk mengamankan data tersebut supaya tidak dapat disalah gunakan. Salah satu cara yang dapat dilakukan untuk melindungi data adalah dengan menggunakan teknik kriptografi. Algoritma yang akan digunakan adalah algoritma RSA (Rivest Shamir Adleman) dan kompresi LZW (Lempel Ziv Welsh) yang akan diimplementasikan pada aplikasi kriptografi berbasis website untuk mengamankan basis data di PT. Central Capital Futures.

Dari uraian diatas, maka penulis merumuskan masalah yang akan dibahas adalah tentang bagaimana mengimplementasikan algoritma RSA (*Rivest Shamir Adleman*) dan kompresi LZW (*Lempel Ziv Welsh*) dalam aplikasi pengamanan *database* dan bagaimana cara mengamankan isi dari *database* yang bersifat rahasia agar terjaga orisinil dan kerahasiaannya di PT. Central Capital Futures.

Berdasarkan rumusan masalah yang telah didapatkan sebelumnya, maka tujuan yang dapat diharapkan bisa terwujud dari penelitian ini adalah mengamankan isi dari *database* pada PT. Central Capital Futures agar tidak dapat diketahui dan dimodifikasi oleh pihak yang tidak berwenang; dapat mengimplementasikan metode algoritma kriptografi RSA (*Rivest Shamir Adleman*) dan kompresi LZW (*Lempel Ziv Welsh*) dalam bentuk aplikasi; menghasilkan aplikasi pengamanan *database* berbasis website yang user friendly, mudah dimengerti dan digunakan oleh pengguna.

## 2. LANDASAN TEORI

### 2.1 *Rivest-Shamir-Adleman* (RSA)

RSA (Rivest-Shamir-Adleman) adalah salah satu sistem kripto dengan kunci publik pertama dan banyak digunakan untuk transmisi data yang aman. Dalam kriptosistem seperti itu, kunci enkripsi bersifat publik dan berbeda dengan kunci dekripsi yang dirahasiakan (privat). Di RSA, asimetri ini didasarkan pada kesulitan praktis dari faktorisasi produk dua bilangan prima besar, yaitu "factoring problem".

a. Proses Pembentukan Kunci

1) Buat dua bilangan prima acak besar, p dan q, dengan ukuran yang hampir sama sehingga produk mereka n = pq adalah panjang bit yang diperlukan, misalnya 1024 bit.
2) Hitunglah n = pq dan (phi) φ(n) = (p-1) (q-1).
3) Pilih bilangan bulat e, 1 <e <phi, seperti gcd (e, phi) = 1.
4) Hitunglah eksponen rahasia d, 1 <d <phi, seperti ed ≡ 1 (mod phi).
5) Kunci publik adalah (n, e) dan kunci privat (d, p, q). Jauhkan semua nilai d, p, q dan phi secret.

Penjelasan dari proses pembentukan kunci di atas adalah :

a) Langkah pertama berarti, angka dapat diuji secara probabilistik untuk primitif.
b) Langkah kedua, PKCS#1 v2.1 menggunakan λ(n) = lcm(p-1, q-1) selain daripada φ(n) = (p-1)(q-1).
c) Langkah ketiga, Pilihan populer untuk eksponen publik adalah e = $2^{16}$ + 1 = 65537. Beberapa aplikasi memilih nilai yang lebih kecil seperti e = 3, 5, atau 35 sebagai gantinya. Hal ini dilakukan untuk membuat verifikasi enkripsi dan tanda tangan lebih cepat pada perangkat kecil seperti kartu pintar namun eksponen publik kecil dapat menyebabkan risiko keamanan yang lebih besar.
d) Langkah ketiga dan keempat dapat dilakukan dengan algoritma *Euclidean* yang diperluas.

Pada public key terdiri dari :

a) N, modulus yang digunakan.
b) e, eksponen publik atau eksponen enkripsi.

Sedangkan private key terdiri dari :

a) N, modulus yang digunakan, digunakan pula pada public key.
b) d, eksponen privat atau eksponen dekripsi yang harus dijaga kerahasiaannya.

Biasanya, berbeda dari bentuk kunci privat (termasuk parameter CRT) :

a) p dan q, bilangan prima dari pembangkitan kunci.
b) d mod (p-1) dan d mod (q-1) (dikenal sebagai dmp1 dan dmq1).
c) (1/q) mod p (dikenal sebagai iqmp).

b. Proses Enkripsi

$$c = n^e \bmod N$$

c. Proses Dekripsi

$$n = c^d \bmod N$$

### 2.2 *Lempel-Ziv-Welch* (LZW)

*Lempel Ziv Welch* (LZW) adalah algoritma kompresi *loseless* yang ditemukan oleh Abraham Lempel, Jacob Ziv, dan Terry Welch. Algoritma LZW dirancang untuk cepat dalam implementasi tetapi biasanya tidak optimal karena hanya melakukan analisis pada data.

Algoritma kompresi LZW tertentu mengambil setiap urutan masukan bit dengan panjang tertentu (misalnya 12 bit) dan membuat entri dalam tabel (kadang-kadang disebut "dictionary" atau "codebook") untuk pola bit tertentu, yang terdiri dari pola itu sendiri dan kode yang lebih pendek. Sebagai masukan dibaca, setiap pola yang telah dibaca sebelum menghasilkan substitusi kode yang lebih pendek, secara efektif menekan jumlah input ke sesuatu yang lebih kecil.

a. Proses Kompresi

Prinsip kompesi tercapai jika referensi dalam bentuk pointer dapat disimpan dalam jumlah bit yang lebih dibandingkan string aslinya. Sebagai contoh, *string* “ABBABABAC” akan dikompresi dengan LZW. Isi *dictionary* pada awal proses diatur dengan tiga karakter dasar yang ada: “A”, “B”, “C”. Tahapan proses kompresi ditunjukkan pada tabel.

Tabel 1, Tahapan Kompersi LZW

| Step | Posisi | Karakter | Dictionary | Output |
|---|---|---|---|---|
| 1 | 1 | A | [4] A B | A |
| 2 | 2 | B | [5] B B | B |
| 3 | 3 | B | [6] B A | B |
| 4 | 4 | A | [7] A B A | AB |
| 5 | 6 | C | [8] A B A C | A B A |
| 6 | 9 | C | ------ | C |

b. Dekompresi LZW

Proses dekompresi pada *Lempel Ziv Welch* (LZW) dilakukan dengan prinsip yang sama seperti proses kompresi. Tahapan dekompresi ditunjukkan pada Tabel

Tabel 2, Tahapan Dekompersi LZW

| Step | Posisi | Output | Dictionary |
|---|---|---|---|
| 1 | 1 | A | ------ |
| 2 | 2 | B | [4] A B |
| 3 | 2 | B | [5] B B |
| 4 | 4 | A B | [6] B A |
| 5 | 7 | A B A | [7] A B A |
| 6 | 3 | C | [8] A B A C |

Dalam proses dekompresi, algoritma LZW tidak menyimpan *string* tabel yang berisi indek-indek dari setiap *code word* yang dihasilkan dalam proses kompresi ke memori akan tetapi menggunakan beberapa informasi yang telah disimpan sebelumnya antara lain 256 karakter ASCII, karakter pertama dari inputan dan *code word* terakhir.

### 2.3 Basis Data

a. Definisi Basis Data

Basis Data adalah koleksi data yang terorganisir. Sebuah basis data relasional, yang lebih ketat, adalah kumpulan skema, tabel, queri, laporan, dan elemen lainnya. Perancang basis data biasanya mengatur data untuk memodelkan aspek dengan cara mendukung proses yang membutuhkan informasi.

b. Tahapan Perancangan Basis Data

Untuk merancang basis data diperlukan beberapa tahap yang diperlukan agar keperluannya sesuai terhadap kebutuhan, seperti berikut :

1) Pengumpulan data dan analisa
2) Perancangan basis data secara konseptual
3) Pemilihan DBMS (*Database Management System*)
4) Perancangan basis data secara logika (data model *mapping*)
5) Perancangan basis data secara fisik
6) Implementasi sistem basis data

## 3. HASIL DAN PEMBAHASAN

### 3.1 Tampilan Layar

Tampilan layar berikut ini merupakan tampilan form fungsi utama yang digunakan dari aplikasi ini.

a. Tampilan Layar *Form Sign In*

Pada tampilan *form Sign In* terdapat dua inputan untuk dapat masuk ke halaman utama yaitu lewat *email* dan *password*. Berikut ini adalah tampilan *form Sign In*.


Gambar 1. Tampilan Layar *Form Sign In*

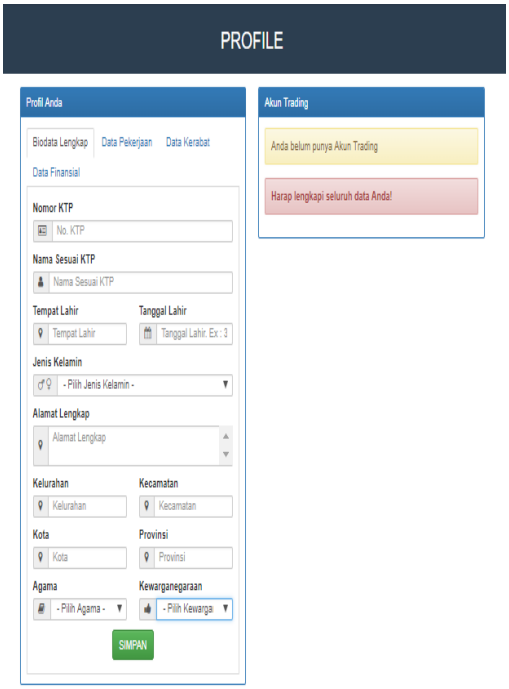
b. Tampilan Layar *Form Sign Up*

Tampilan layar *form Sign Up* digunakan *user* untuk mendaftarkan akun baru sehingga bisa masuk pada aplikasi. Dan nanti *user* dapat menggunakan fasilitas yang ada pada pada halaman utama. Berikut ini adalah tampilan layar *form Sign Up.*


Gambar 2. Tampilan Layar *Form Sign Up*

c. Tampilan Layar Halaman *Profile* dan Biodata Lengkap Sebelum *Trading*

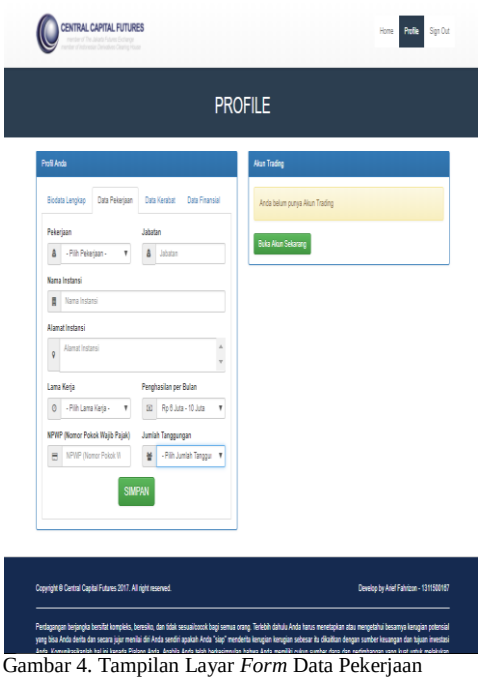
Tampilan Layar Halaman *Profile* saat pertama kali dibuka akan menampilkan data *user* yang baru membuat akun *user*, di sini ditampilkan beberapa *form* pengisian data yang nanti akan diisi oleh *user* untuk melengkapi data diri sebelum nanti *user* meng-klik pilihan simpan untuk melanjutkan ke *form* berikutnya. Pada tampilan yang terlihat di bawah, dimana akun *trading* masih kosong atau belum dibuat *user* akan diminta untuk membuat akun setelah melengkapi data pada *form* yang ada.




Gambar 3. Tampilan Layar Halaman *Profile* dan Biodata Lengkap Sebelum *Trading*

d. Tampilan Layar *Form* Data Pekerjaan

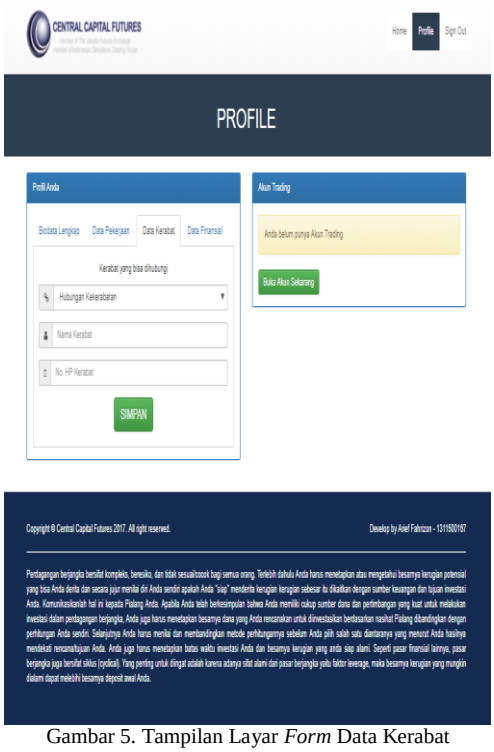
Tampilan Layar Halaman *Profile* di sini ditampilkan *form* Data Pekerjaan untuk *user* melakukan pengisian data yang nanti akan diisi untuk melengkapi data diri sebelum nanti *user* meng-klik pilihan simpan untuk melanjutkan ke *form* berikutnya. Pada tampilan yang terlihat dibawah, dimana akun *trading* masih kosong atau belum dibuat *user* akan diminta untuk membuat akun setelah melengkapi data pada *form* yang ada.




Gambar 4. Tampilan Layar *Form* Data Pekerjaan

e. Tampilan Layar *Form* Data Kerabat

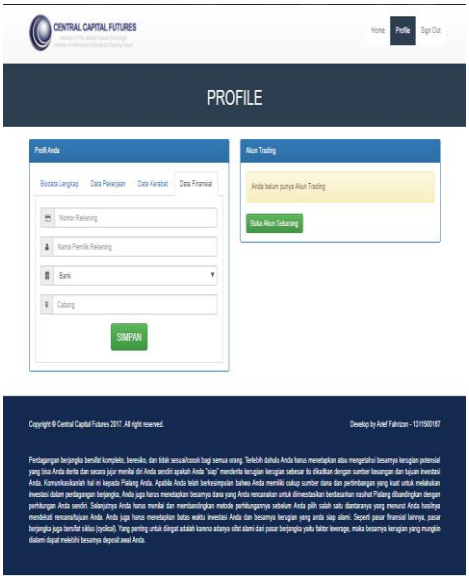
Tampilan Layar Halaman *Profile* di sini ditampilkan *form* Data Kerabat untuk *user* melakukan pengisian data yang nanti akan diisi untuk melengkapi data diri sebelum nanti *user* meng-klik pilihan simpan untuk melanjutkan ke *form* berikutnya. Pada tampilan yang terlihat dibawah, dimana akun *trading* masih kosong atau belum dibuat *user* akan diminta untuk membuat akun setelah melengkapi data pada *form* yang ada.




Gambar 5. Tampilan Layar *Form* Data Kerabat

f. Tampilan Layar *Form* Data Finansial

Tampilan layar halaman terakhir pada *form* pengisian data *user* pada halaman *Profile*. Di sini ditampilkan *form* Data Finansial untuk *user* melakukan pengisian data yang nanti akan diisi untuk melengkapi data diri sebelum nanti *user* meng-klik pilihan simpan untuk melanjutkan ke *form* berikutnya. Pada tampilan yang terlihat dibawah, dimana akun *trading* masih kosong atau belum dibuat *user* akan diminta untuk membuat akun setelah melengkapi data pada *form* yang ada.




Gambar 6. Tampilan Layar *Form* Data Finansial

g. Tampilan Layar *Form Trading*

Tampilan layar *form Open Live Account* yang berfungsi untuk *user* membuat akun *trading*. Di sini dijelaskan beberapa syarat dan ketentuan yang harus dibaca oleh *user* dan disetujui *user* dengan meceklis setiap *user* sudah membaca syarat dan ketentuan yang ada. Dan di akhir sebelum *user* menyimpan data atau membuat akun *trading*, *user* diminta mengisi kolom data yang ada dan menyetujui akan mengikuti semua syarat ketentuan yang ada dengan menceklis sekali lagi.


Gambar 7. Tampilan Layar *Form Trading*

### 3.2 Tabel Pengujian

Dalam pengujian kali ini dibahas hasil *ciphertext* yang diambil dari enkripsi *plaintext* kemudian terdapat juga waktu proses enkripsi dan juga *size* atau ukuran data dari keseluruhan data yang diwakilkan oleh masing-masing *plaintext.*

Tabel 3, Tabel Pengujian

| Plaintexks | Cipherteks | Size | Waktu Proses Enkripsi |
|---|---|---|---|
| Arief Fahrizon | fHgdROnCGGG04FYx1M2xgSG6ySStFYRGm4zKRG+HCIP3g0R6Zy+cWs+zO6TceVayFAw2uD0EM3qYhgU186VkGH6Zn0jHyvzOJ2A/wCATQqjYOGUmRAeT6jy8QxS4CMwjK6RIEkoHVK2VofAWm2iuwixne9l2L04dhegHYmDM7zYljacEQTCWpmgO2mdWyx1s | 5 KB | 0.145366907 |
| Rizki Muhammad Hanafi | MAEZzwlDaF0KsF+02eLHYT2Umz2rmm0gm7wmuFSqA08SK1lWe0UGDUrFIrUqeSU7DivwWmAiy0iIDeWTi4yEIVqzQ6IRGCXACkajiSXiKdnIGBwPxAYxC8HOIDo1QE0lqumiaDMzDyKGOjVgFwSdG0FgSR0AdVIrxGMyGXnAOxkwR8szqo3mGi26EwGkKXly | 6 KB | 0.155503034 |

### 3.3 Evaluasi Program

a. Kelebihan Aplikasi

1) Aplikasi mudah digunakan karena tampilan yang sederhana sehingga memudahkan *user* dalam menggunakan aplikasi.
2) Data yang telah terenkrip tidak dapat dibuka, sehingga meminimalkan kebocoran informasi.
3) Terdapat keamanan ganda pada seluruh data *user* yang sudah pasti aman.
4) Data hasil dekripsi tidak mengalami perubahan (bentuk, ukuran, maupun nama) atau kerusakan dan dapat dibaca kembali oleh pengguna.
5) Terdapat validasi setiap *user* melakukan *input* data pada *Front-End.*

b. Kekurangan Aplikasi

1) Apabila koneksi internet yang lemah akan membuat aplikasi yang berjalan lambat atau dapat berhenti melakukan proses.
2) Karena menggunakan *server database* sebagai pihak ketiga, maka proses sewaktu. *Sign In* pada aplikasi perlu mengaktifkan *MySQL* dan *Apache Server* sebagai jembatan penghubung antara aplikasi dengan *database server*.
3) Tidak terintegrasi langsung dengan *Meta Trader* 4, karena program dibuat sebagai *prototype.*

## 4. KESIMPULAN

Berdasarkan serangkaian hasil uji coba pada aplikasi dan tanggapan dari *user* atau pengguna, maka ditemukan kesimpulan, sebagai berikut :

a. Dengan menggunakan dua algoritma yaitu RSA dan LZW ini dapat mengamankan data atau informasi yang ada dikantor PT. Central Capital Futures supaya dapat lebih aman kerahasiaan.
b. Kecepatan penggunaan aplikasi sangat tergantung dengan data dan proses yang akan dienkrip maupun didekrip.
c. Data *user* yang telah terenkripsi dapat dikembalikan menjadi data asli tanpa mengalami kerusakan atau perubahan karakter, sehingga informasi dari data *user* yang disampaikan tetap terjamin.
d. Aplikasi ini hanya dapat digunakan oleh *user* yang memiliki hak akses oleh admin.

## 5. DAFTAR PUSTAKA

[1] Putra, N.A. & Busron, D., 2014. Rekayasa perangkat lunak kriptografi menggunakan algoritma rsa pada sistem keamanan file berbasis java. *Jurnal TEKNOIF*, 2(1), pp.7-17.
[2] Rahajoeningroem, T. & Aria, M., 2011. Studi dan Implementasi Algoritma RSA untuk pengamanan Data transkip Mahasiswa. *Majalah Ilmiah Unikom*, 8(1), pp.77–90.
[3] Sudirman, 2017. ANALISIS ENKRIPSI CITRA DIGITAL MENGGUNAKAN ALGORITMA LOGISTIC MAP DENGAN ALGORITMA KOMPRESI LAMPEL-ZIV-WELCH ( LZW ). *Jurnal Nasional Informatika dan Teknologi Jaringan*, (374), pp.95–99.
[4] Suhastra, F., 2014. IMPLEMENTASI ALGORITMA KOMPRESI LAMPEL ZIV WELCH ( LZW ) PADA BERKAS DIGITAL. *Pelita Informatika Budi Darma*, VI(3), pp.54–57.
[5] Wahyadyatmika, A.P., Isnanto, R.R. & Somantri, M., 2014. IMPLEMENTASI ALGORITMA KRIPTOGRAFI RSA PADA SURAT ELEKTRONIK ( E-Mail ). *Transient*, 3(4), pp 1-9.