APLIKASI PENGAMANAN DATABASE BERBASIS DESKTOP DENGAN ALGORITMA AES-128 DAN *RIVEST CODE (RC4)*

Melisa Dwi Wulandari[1)], Dewi Kusumaningsih[2)]
[1]Program Studi Teknik Informatika,Fakultas Teknologi Informasi, Universitas Budi Luhur
[1,2]Jl. Raya Ciledug, Petukangan Utara, Jakarta Selatan 12260
Telp. (021) 5853753, Fax. (021) 5866369
E-mail : Melisa.Ahara@gmail.com[1)], Dewi.kusumaningsih@budiluhur.ac.id[2)]

Abstrak

Setiap perusahaan memiliki jumlah karyawan yang relative banyak dengan data yang tersimpan di dalam database, meliputi data pribadi karyawan dan data overtime. Masih ada perusahaan yang memiliki database yang sama dengan teks-teks asli yang ditampilkan sebagai informasi terhadap pengguna. Hal ini dapat memberikan kesempatan orang lain yang tidak memiliki hak akses dapat mengetahui secara langsung isi dari database tersebut serta dapat memberikan peluang kepada mereka untuk melakukan pembocoran, mendistribusikan maupun melakukan modifikasi terhadap isi database tersebut. Oleh karena itu, dibutuhkan aplikasi yang dapat memudahkan pengguna untuk menginput dan menyimpan data-data tersebut dengan aman dan terjaga. Teknik pengamanan data ini dilakukan dengan menggunakan teknik kriptografi Advanced Encryption Standard 128 (AES-128) dan Rivest Code 4 (RC4) yang dapat dimanfaatkan untuk mengamankan data serta memberikan kemudahankepada user untuk mengamankan data agar isi dari data tersebut tidak diketahui oleh pihak yang tidak memiliki kepentingan terhadap data tersebut. Aplikasi pengamanan database dengan menggunakan algoritma kriptografi Advanced Encryption Standard 128 (AES-128) dan Rivest Code 4 (RC4) dapat diimplementasikan dalam bahasa VB.NET mampu mengamankan isi dari database dengan baik. Dengan adanya aplikasi pengamanan database, isi dari database tersebut dapat lebih terjaga kerahasiaannya dan keamanannya.

Kata kunci : Kriptografi, *Advanced Encryption Standard (AES-128),Rivest Code 4 (RC4),Database*

1. PENDAHULUAN

Pada perkembangan teknologi saat ini kehidupan manusia banyak bergantung pada teknologi informasi. Dengan semakin majunya perkembangan teknologi memungkinkan manusia untuk berkomunikasi, bertukar informasi, ataupun bertukar data dengan mudah dalam jarak jauh. Keuntungan yang diberikan oleh teknologi juga memiliki dampak negative, yaitu kejahatan pencurian data. Oleh sebab itu, keamanan data yang tersimpan sangatlah penting dijaga keamanan dari berbagai ancaman yang berupa pengaksesan, perubahan, dan penghapusan data oleh pihak yang tidak bertanggung jawab.

Database secara umum merupakan susunan record data operational lengkap dari suatu organisasi atau perusahaan yang diorganisir dan disimpan dalam media penyimpanan secara terintegrasi dengan sistem informasi yang berjalan yang dapat dijadikan sebagai salah satu sumber sehingga informasi terpenuhi secara optimal sesuai kebutuhan para pengguna. Dengan adanya *database*, maka proses pemutankhiran informasi dapat dilakukan.

Beberapa perusahaan yang memiliki karyawan yang relative banyak yang datanya disimpan di *database*yang berisi data karyawan dan data *overtime*. *Database* yang tersimpan di dalam *database* masih sama dengan teks-teks yang ditampilkan, memberikan kesempatan seorang yang tidak memiliki hak akses untuk dapat mengetahui isi dari *database* tersebut serta dapat memberikan peluang untuk dapat dilakukan pembocoran, pendistribusian, maupun modifikasi lain terhadap isi *database* tersebut.

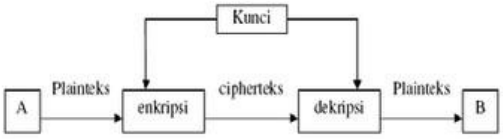
Untuk mengatasi masalah-masalah tersebut maka diperlukan cara untuk mengamankan data agar *database* tidak dapat disalahgunakan. Salah satu caranya adalah dengan melindungi data menggunakan teknik kriptografi. Kriptografi merupakan ilmu dan seni yang berkaitan dengan aspek keamanan informasi seperti integritas data, kerahasiaan, otentikasi, dan anti penyangkalan. Algoritma yang akan digunakan adalah algoritma AES-128 dan RC4 yang akan diimplementasikan pada kriptografi berbasis desktop berbasis data.Berdasarkan uraian diatas maka penulis skripsi ini membahas bagaimanakah mengamankan sebuah *database* yang bersifat rahasia agar terjaga keaslian dan kerahasiaannya dengan mengimplementasikan algoritma AES-128 dan RC4 ke dalam pengamanan *database*.

2. LANDASAN TEORI

2.1. Kriptografi

Kata kriptografi berasal dari Yunani yang terdiri dari dua kata yaitu *crypto* dan *graphia*. Kata *crypto* berarti rahasia sedangkan *graphia* berarti tulisan. Orang-orang Mesir menggunakan kriptografi untuk menyampaikan suatu pesan kepasukan militer yang berada di lapangan agar pesan yang dikirim tidak terbaca oleh pihak musuh walaupun prajurit pembawa pesan tertangkap. Secara umum kriptografi merupakan teknik keamanan informasi yang dilakukan dengan cara merubah informasi yang asli (*plaintext*) menggunakan kunci tertentu dengan suatu metode

enkripsi sehingga dapat menghasilkan suatu informasi yang tidak dapat terbaca secara langsung (*chipertext)*[1].



Gambar 1.Proses Enkripsi dan Dekripsi Dengan Kunci

2.1.1 Kriptografi Klasik

Kriptografi klasik pertama ditemukan pada ukiran *hieroglyph* yang tidak lazim dalam pyramid meskipun arti atau definisi dari tulisan tersebut belum dapat diperjelaskan. Untuk mendapatkan pengertian bagaimana suatu algoritma kriptografi berkembang, dapat ditelusuri satu per satu algoritma kriptografi klasik dari teknik substitusi dan teknik transposisi. Teknik Substitusi adalah penggantian setiap karakter *plaintext* dengan karakter lain sedangkan teknik transposisi (permutasi) adalah teknik dengan pesan yang asli tidak dapat terbaca kecuali menggunakan kunci untuk mengembalikan pesan tersebut ke bentuk semula atau disebut dengan dekripsi[1].

2.1.2 Kriptografi Modern

Enkripsi modern berbeda dengan enkripsi konvensional dikarenakan pada enkripsi modern sudah menggunakan komputer dalam pengoperasiannya, yang berfungsi mengamankan data baik yang ditransfer melalui jaringan maupun tidak. Kriptografi modern merupakan suatu perbaikan yang diacak pada kriptografi klasik. Untuk kerahasiaan algoritma, kriptografi modern sudah tidak bergantung pada keamanannya melainkan pada diketahui oleh siapa saja, akan tetapi tanpa dibekali oleh pengetahuan mengenai kunci, data tersandi tetap saja tidak dapat terbuka[1].

Terdapat perbedaan yang signifikan antara kriptografi klasik dan kriptografi modern, yang pertama adalah kriptografi modern bekerja dalam bentuk bit sedangkan kriptografi klasik beroperasi dalam bentuk karakter.

2.1.3 Algoritma Kriptografi

Untuk menyelesaikan masalah secara sistematis, menurut definisi terminologinya algoritma menggunakan langkah-langkah logis bagaimana pesan tidak dapat terbaca oleh orang-orang yang tidak memiliki hak atas pesan tersebut[2].

Algoritma kriptografi terdiri dari lima fungsi dasar, yaitu :

a. Enkripsi adalah merubah *plaintext*kedalam bentuk *chiertext* menggunakan algoritma yang dapat mengkodekan data yang diinginkan agar tidak dapat dimengerti oleh orang lain yang tidak berkepentingan.
b. Deskripsi adalah kebalikan dari proses enkripsi yaitu menterjemahkan sandi-sandi yang telah diacak ke bentuk aslinya
c. Plaintext adalah pesan yang hendak dikirim yang merupakan data asli
d. Kunci adalah suatu bilangan yang dirahasiakan untuk proses enkripsi dan dekripsi.
e. Chipertext adalah pesan yang tersandi atau telah terenkripsi

2.2 Jenis Kriptografi

Algoritma kriptografi terdiri 3 bagian berdasarkan dari kunci yang digunakannya :

a. Algoritma Simetri
   Algoritma yang disebut juga dengan algoritma klasik,karena kegiatan enkripsi dan dekripsinya menggunakan kunci yang sama.
b. Algoritma Asimetri
   Algoritma asimeteri sering disebut juga dengan algoritma kunci public, dengan arti kata kunci uang digunakan untuk melakukan enkripsi dan dekripsi berbeda
c. Hash Function
   Mengamankan pesan agar tetap terjaga sampai ke orang yang diinginkan dengan sidik jari.

2.3 Algoritma AES

2.3.1 Sejarah Singkat Algoritma

Pada tahun 2001 *Advanced Encryption Standard (AES)* dipublikasikan oleh *NIST (National Institute of Standard and Technology)*. Simetris block chipper *DES (Data Encryption Standard)*digantikan oleh *AES*. Selama puluhan tahun DES mengjadi algoritma yang aman di dunia, tapi tahun 1990 DES dinilai memiliki kunci yang pendek dan pada tahun 1998, dalam tempo waktu 96 hari 70 ribu PC di Internet berhasil dibuka dengan satu kunci DES, sedangkan tahun 1999 dalam waktu 22 hari. Karena algoritma enkripsi DES sudah berhasil dipecahkan pada tahun 1998 dalam 96 hari, maka dibuat mesin khusus untuk memecahkan algoritma DES dan mesin tersebut dapat memecahkan 25% kunci DESdalam waktu 2,3 hari dan seluruh kunci DES dalam waktu rata-rata 4,5 hari[2].

2.3.2 Pengertian Algoritma AES

Advanced Encryption Standard (AES) merupakan algoritma kriptografi yang dapat dimanfaatkan untuk mengamankan data. Algoritma AES adalah dapat mengenkripsi (*enchiper*) dan mendekripsi (*dechiper*) informasi dengan chipertext simetrik. Algoritma AES ini menggunakan kunci kriptografi 128, 192, dan 256 bits untuk mengenkrip dan dekrip data pada blok cepatan, dan karakteristik algoritma beserta implementasinya[3].

Kriteria penilaian algoritma DES yang dianggap masa berlakunya telah usai oleh NIST didasarkan pada 3 kriteria utama berikut:

a. Aspek Keamanan

   Keamanan merupakan aspek terpenting dalam penilaian yang mengacu pada ketahanan algoritma terhadap serangan, kompleksitas penghitungan matematis, *output* yang dihasilkan, dan perbandingan aspek keamanan satu sama lain.

b. Aspek Biaya

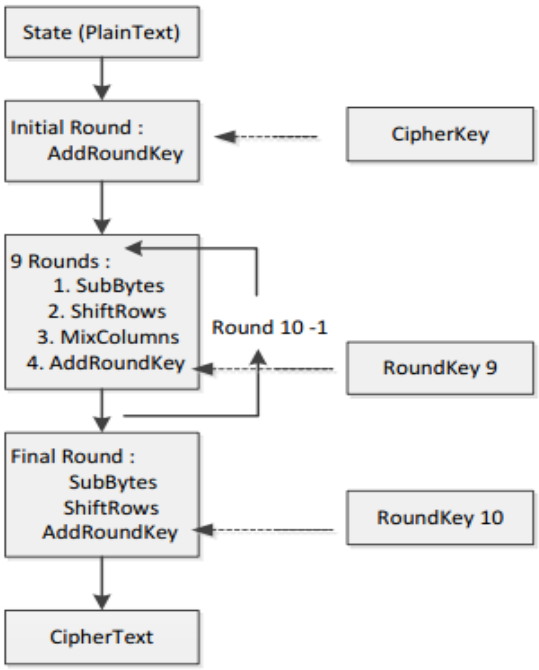
   Mengacu pada lisensi, efisiensi komputasional di berbagai *platform*, dan kebutuhan *memory* sesuai dengan tujuan NIST yang menginginkan agar algoritma AES dipublikasikan di internet secara gratis, dan juga pada smart card yang memiliki memori yang kecil dapat diimplementasikan dengan biaya yang murah

c. Aspek Implementasi dan Karakteristik Algoritma.

   Aspek ini mengacu pada fleksibilitas, kesesuaian terhadap perangkat lunak maupun keras, serta kesederhanaan algoritma. Pada algoritma AES, jumlah blok *input*, blok *output*, dan *state* adalah 128-bit. Dengan besar data 128-bit, berarti Nb = 4 (Nb = panjang blok *plaintext* dibagi 32 dan Nk = panjang kunci dibagi 32) yang menunjukkan panjang data tiap baris adalah 4 *byte*. Dengan blok *input* atau blok data sebesar 128-bit, *key* yang digunakan pada algoritma AES-128 tidak harus mempunyai besar yang sama dengan blok *input.Cipherkey* pada algoritma AES-128 dapat menggunakan kunci dengan panjang 128-bit, 192-bit, atau 256-bit. Jumlah *round* yang akan diimplementasikan pada algoritma AES-128 ini dapat dibedaan oleh panjang kunci. Berikut ini merupakan tabel yang menginfomasikan jumlah *round* (Nr) yang harus diimplementasikan pada masing-masing panjang kunci.

2.3.3 Proses Enkripsi AES

Ada 4 jenis transformasi byte pada proses enkripsi algoritma AES, yaitu *SubBytes, ShiftRows Mixcolumns*, dan *AddRoundKey*. *Input* yang telah diduplikasikan ke dalam *state* akan mengalami transformasi byte *AddRoundKey* pada awal proses enkripsi. Setelah itu, state akan melalui proses transformasi *SubBytes, ShiftRows, MixColumns* dan *AddRoundKey* secara berulang-ulang sebanyak Nr. *Round function* merupakan sebutan dari proses dari algoritma AES. *Round* yang terakhir berbeda dengan *round-round* sebelumnya dimanapada *round* terakhir,state tidak melalui proses transformasi *MixColumn*s, dibawah ini adalah gambar proses enkripsi AES[3].



Gambar 2. Proses Enkripsi AES-128

Tabel 1. Perbandingan Panjang Kunci Aes

Tipe	Panjang kunci (NK)	Ukuran Blok (NB)	Jumlah Putaran (NR)
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

2.3.4 Proses Dekripsi

Transformasi cipher dapat dibalikkan dalam arah yang berlawanan untuk menghadilkan inverse chipper yang mudah dimengerti dan diiplementasikan untuk algoritma AES. *InvShiftRows, InvSubBytes, InvMixColumns*, dan *AddRoundKey* adalah Transformasi *byte* yang digunakan pada *invers chiper[3]*.

2.4 Algoritma Rivest Code 4 (RC4)

2.4.1 Sejarah Singkat RC4

Pada tahun 1987 Ron Rivest yang berasal dari laboratorium RSA pertama kali mendesain RC4, singkatan resmi RC itu sendiri yaitu "*RivestChiper*", tetapi dikenal juga sebagai "*Ron'sCode*". RC4 sudah memiliki hak paten, untuk menghindari pemetaan sering disebut dengan "ARCFOUR" atau "ARC4" (AllegedRC4). Secara tidak resmi RSA *Security* merilis algoritma tersebut, namun Wikipedia Inggris dihubungi oleh Rivest secara pribadi untuk merilisnya dengan catatan-catatan yang dimilikinya. RC4 sudah menjadi bagian dari protocol enkripsi yang standart dan sering digunakan. Kecepatannya dan kesederhanaannya dalam menangani banyak aplikasi menjadi faktor utama dalam kesuksesannya, sehingga mudah untuk mengembangkan implementasi yang effisien ke software dan

hardware, RC4 disebut sebagai algoritma kriptografi simetris karena menggunakan kunci yang sama untuk mengenkripsi atau dekripsi suatu pesan data, ataupun informasi[4].

2.4.2 Pengertian RC4

Salah satu jenis stream Cipher yaitu memproses unit atau *input* data pada satu saat adalah RC4. Dengan cara ini pada Panjang yang variable dapat melaksanakan proses enkripsi atau dekripsi. Sebelum diproses algoritma tidak wajib menunggu jumlah *input* data tertentu, atau untuk mengenkrip dapat menambahkan byte tambahan.

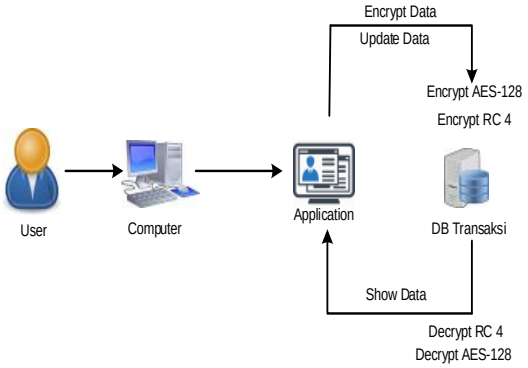
RC4 menggunakan panjang variabel untuk menginisialisasi state label dengan kunci 1 s.d. 256 byte. Untuk pengurutan menghasilkan byte pseudo random yang berikutnya menjadi pseudo-random menggunakan state table. Setelah di-XOR dengan plainteks sehingga didapat ciphertext. Tiap element pada state tabel di swap sedikitnya sekali. Kunci RC4 dibatasi sampai dengan 40 bit. Tetapi besar kemungkinan untukmenggunakan kunci 128 bit. Penggunaan kunci antara 1 ssmpsi 2048 bit merupakan kemampuan yang dimiliki RC4. Faktor utama dalam sekuritas data adalah panjang kunci. RC4 bisa mempunyai kunci sampai dengan 128 bit[4]

3. RANCANGAN SISTEM DAN APLIKASI

Tahapan-tahapan yang terjadi dalam proses sistem aplikasi ini dapat dijelaskan sebagai berikut :

3.1 Skema Proses

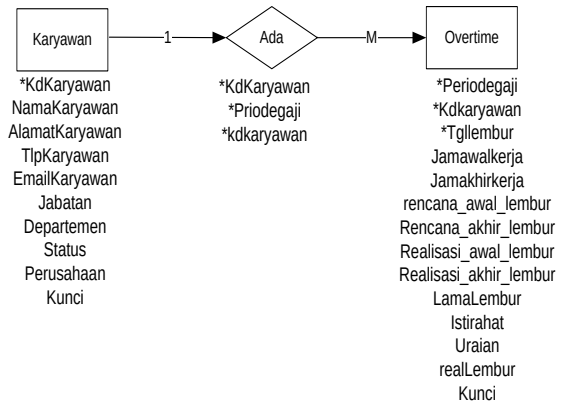
Berikut ini merupakan skema proses aplikasi, untuk dapat memahami konsep aplikasi yang akan dibangun. Pada gambar skema proses aplikasi menggambarkan secara garis besar proses dari keseluruhan sistem:



Gambar 3. Skema Proses Aplikasi

3.2 ERD (*EntityRelationship Diagram)*

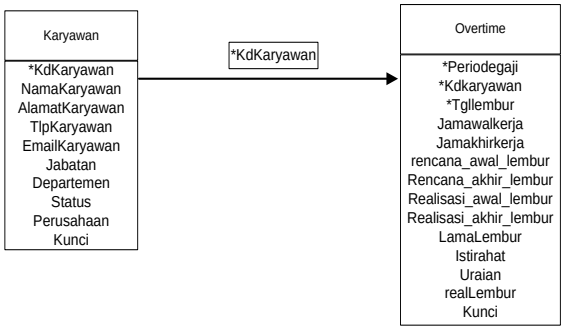
Entity Relationship Diagram (ERD) ini berisi himpunan relasi dan komponen-komponen himpunan entitas. Seluruh data yang ada diwakili dengan masing-masing yang telah dilengkapi dengan atribut-atribut. Seperti pada Gambar 4 adalah gambar rancangan ERD:



Gambar 4 : Entity Relationship Diagram

3.2 *Logical RecordStructure (LRS)*

Berikut gambar LRS yang diusulkan untuk aplikasi:

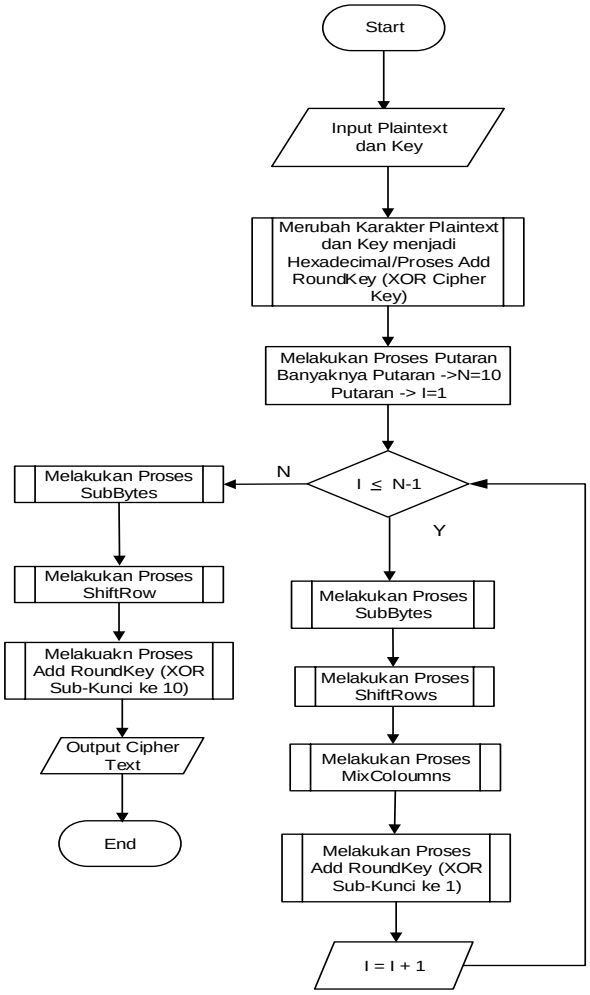


Gambar 5 : Logical Record Structure (LRS)

3.3 *Flowchart* Aplikasi

3.3.1 *Flowchart*Proses Enkripsi AES-128

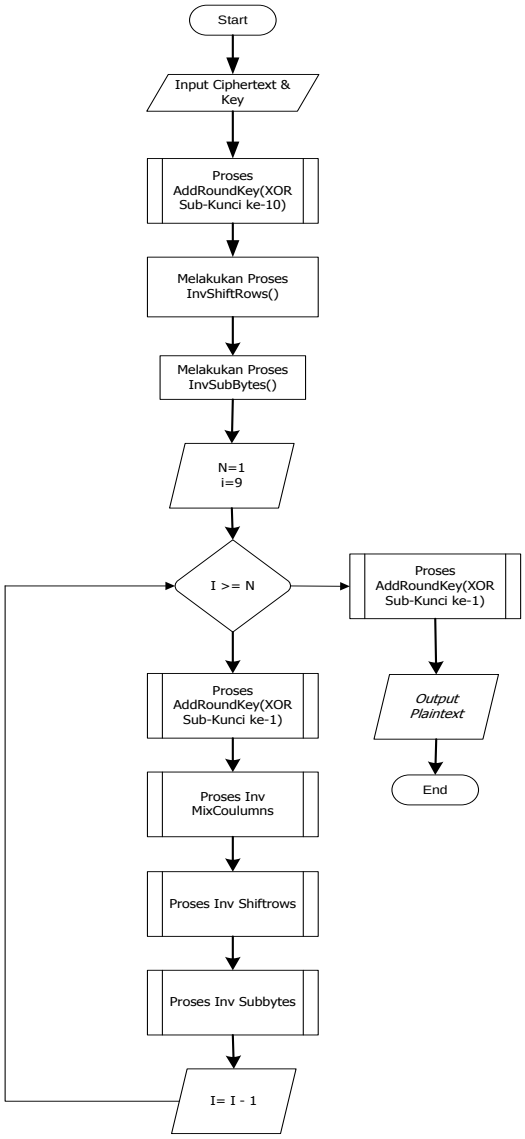
*Flowchar*t pada *Gambar.6*menjelaskan alur proses atau cara kerja algoritma AES-128 untuk menghasilkan *ciphertext.* Untuk lebih jelasnya berikut adalah proses dari *flowchart* proses enkripsi AES-128.



Gambar. 6: Flowchar Proses Enkripsi AES-128

3.3.2 *Flowchart Proses Deskripsi AES-128*

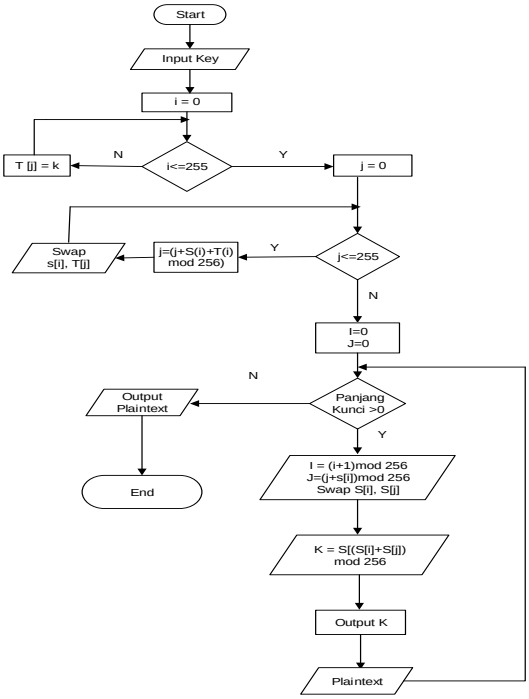
Flowchart pada Gambar.7 menjelaskan alur proses atau cara kerja algoritma AES-128 untuk menghasilkan *Plaintext.* Untuk lebih jelasnya berikut adalah proses dari *flowchart* proses dekripsi AES-128.



Gambar 7: Flowchart Proses Dekripsi AES-128

3.3.3 *Flowchart Proses Enkripsi RC4*

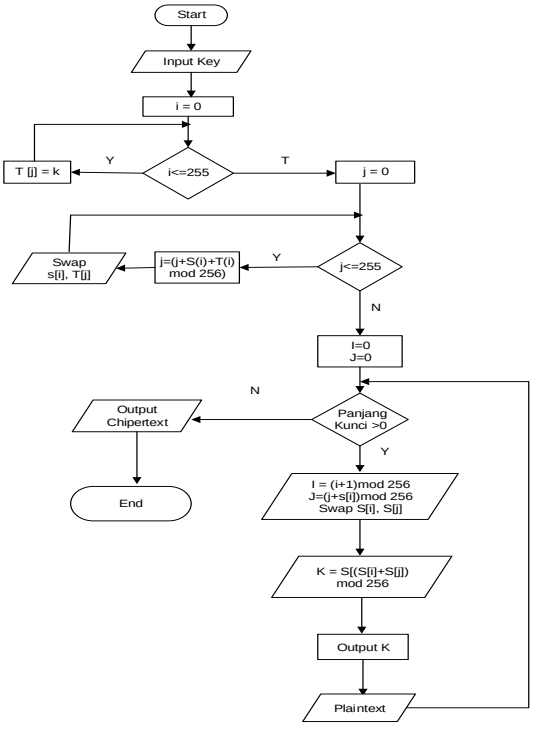
Flowchart pada Gambar .8 menjelaskan alur proses atau cara kerja algoritma RC 4 untuk menghasilkan *ciphertext.* Untuk lebih jelasnya berikut adalah proses dari *flowchart* proses enkripsi RC 4.



Gambar 8: Flowchart Proses Enkripsi RC4

3.3.3 *Flowchart Proses Dekripsi RC4*

Flowchart pada Gambar 9 menjelaskan alur proses atau cara kerja algoritma RC 4 untuk menghasilkan *Plaintext*. Untuk lebih jelasnya berikut adalah proses dari *flowchart* proses dekripsi RC 4:

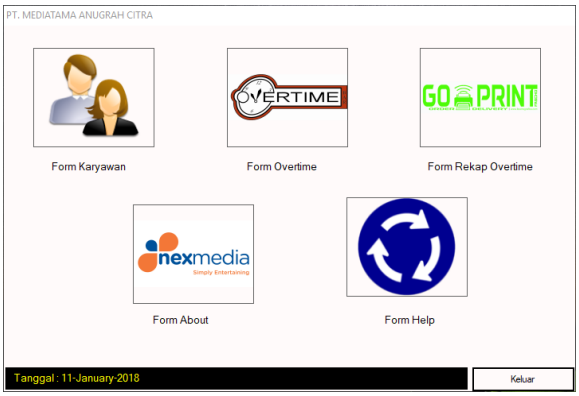


Gambar 9: Flowchart Proses Enkripsi RC4

4. HASIL DAN PEMBAHASAN

4.1 Tampilan Halaman Menu Utama

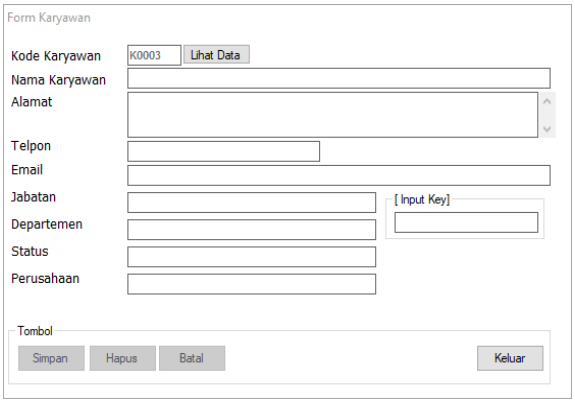
Tampilan layar menu utama adalah *form* yang secara otomatis tampil pada saat aplikasi dibuka dan *form* tersebut terdiri dari 5 menu yaitu *Form* karyawan, *FormOvertime*, *Form* Rekap *Overtime*, *FormAbout*, *FormHelp* dan satu tombol keluar, seperti pada Gambar 10:



Gambar 10: Tampilan Menu Utama

4.2 Tampilan Form Menu Karyawan

Pada *form* Karyawan terdapat kode karyawan yang secara otomatis terinput sesuai dengan urutan kode karyawan, terdapat juga field penginputan Nama Karyawan, Alamat, Telepon, Jabatan, Departemen, Status, Prusahaan, dan input Key, dan Terdapat 5 *button* yaitu simpan, ubah, hapus, batal, dan keluar dan terdapat 1 *button* yaitu lihat data, dapat dilihat pada Gambar 11 berikut ini:

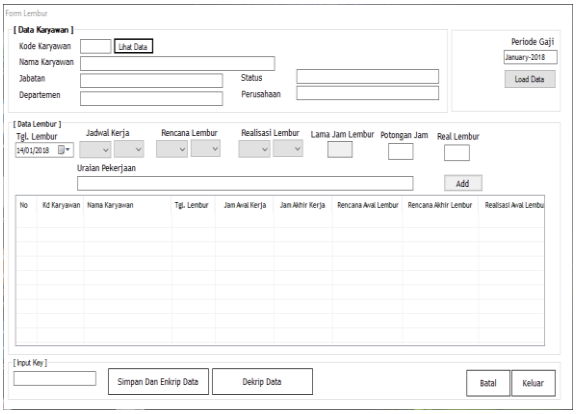


Gambar.11: Tampilan Form Menu Karyawan

4.3 Tampilan Form Menu Overtime

Tampilan ini akan muncul ketika *user* memilih menu *overtime.* Di dalam menu *overtime* ini *user* harus menginput terlebih dahuku kode karyawan yang berada di dalam *database* dengan cara menekan tombol lihat, lalu diikuti dengan meng-*input* tanggal lembur, jadwal kerja, rencana lembur, realisasi lembur, lama jam lembur, potongan jam, *real* lembut, dan uraian pekerjaan dengan menekan tombol *Add.* Setelah menekan tombol *add* kemudian *input key*untuk mengenkrip data yang sudah ditambahkan ke table *view* sebelum disimpan ke dalam *database.* Terdapat 7 *button* yaitu *button* lihat

data, *button load* data, *button* simpan dan enkrip data yang berfungsi untuk menyimpan dan mengenkrip data dan *button* batal untuk membatalkan proses *input*-an dan button keluar, dapat dilihat pada Gambar 12.



Gambar 12 Tampilan Form Menu Overtime

4.4 Evaluasi Sistem

Setelah dilakukan analisa dari hasil pengujian aplikasi dapat ditemukan beberapa kelebihan dan kekurangan dari aplikasi , yaitu:

a. Kelebihan Aplikasi
   1) Tampilan aplikasi *user friendly* sehingga mudah digunakan
   2) *Database Overtime* dan *database* Karyawan menjadi lebih aman karena telah dilakukan proses enkripsi
   3) Hasil Rekap *Overtime* bias langsung dilihat oleh admin dan dapat melihat langsung data *Overtime*
   4) Mempermudah pencarian data pada *database*

b. Kekurangan Aplikasi
   1) Tidak dapat diakses dimana saja karena berbasis *desktop*
   2) Ukuran *file* enkripsi menjadi lebih besar karena menggunakan algoritma AES-128 yang melakukan proses putaran sebanyak sembilankali putaran
   3) Aplikasi ini hanya mengamankan isi *field*

5. KESIMPULAN

Berdasarkan analisis yang telah dilakukan terhadap permasalahan yang terjadi, terdapat kesimpulan dan saran yang diperlukan dalam pengembangan aplikasi mendatang.

a. Kesimpulan

   Dari hasil perancangan dan percobaan aplikasi ini. Dapat diambil kesimpulan sebagai berikut :

   1) Enkripsi Algoritma AES-128 dan *Rivest Code* 4 (RC 4) dapat diimpementasikan pada aplikasi pengamanan *database* dengan menggunakan bahasa VB.net dan *database* HeidiSQL
   2) Aplikasi ini dapat mengamankan data yang masuk kedalam *database* dengan teknik kriptografi menggunakan metode AES-128 dan *Rivest Code* 4 (RC 4) sehingga data yang tersimpan kedalam *database* akan sulit untuk dibaca. Aplikasi ini dapat dijalankan sesuai dengan spesifikasi teknik yang dirancang

b. Saran

Selain menarik kesimpulan, penulis juga membuat saran-saran agar dapat memperbaiki kekurangaan aplikasi sehingga menjadi aplikasi yang lebih baik lagi. Berikut saran-saran dari pembuatan aplikasi yaitu:

   a. Diharapkan dilakukan pelatihan terlebih dahulu kepada *user* agar *user* benar-benar memahami sistem dan cara penggunaan sekaligus pemeliharaannya sehingga sistem dapat digunakan dengan optimal untuk jangka waktu yang lama
   b. Program atau perangkat lunak ini dapat dikembangkan dengan menambahkan penjelasan yang lebih detail dan lebih baik.

6. DAFTAR PUSTAKA

[1] Ariyus, D. 2006. Kriptografi Keamanan Data Dan Komunikasi. Yogyakarta: Graha Ilmu

[2] Haryanto, H., Wiryadinata, R. dan Afif, M. (2014) "Implementasi Kombinasi Algoritma Enkripsi Aes 128 Dan Algoritma Kompresi Shannon-Fano," *Setrum*, 3(1), hal. 16–25

[3] Ilyas, I. A. dan Widodo, S. (2014) Kriptografi *File* Menggunakan Metode Aes Dual Password, *Prosiding Seminar Ilmiah Nasional Komputer dan Sistem Intelijen (KOMMIT 2014)*, 8(2302–3740), hal. 263–270

[4] Jumrin, Sutardi & Subardin, 2016. Aplikasi Sistem Keamanan Basis Data Dengan Teknik Kriptografi RC4. semanTIK, 2(1), pp.59–64. ISSN: 2502-8928.