Perancangan Aplikasi Pengamanan Pesan Teks dengan Menggunakan Metode Wake (Word Auto Key Encryption)

Nurlela Sinaga
Teknik Informatika STMIK Budi Darma Medan Jl. Sisingamangaraja No. 338 - Medan
e-mail : sinaga417@gmail.com

Abstrak

Banyak sekali kasus data yang tersimpan dalam computer yang tidak terjamin keamananny, kadang rusak / tidak terbaca bahkan juga yang hilang. Ini semua dikarenakan terinfeksi virus computer yang selalu ada yang terbaru.keadaan ini membuat para pemilik data tidak nyaman. Untuk mengatasi keresahan tersebut penulis merancang suatu perangkat lunak untuk mempelajari metode kriptografi, metode yang dipilih penulis adalah metode Word Auto Key Encryption (WAKE) karena metode ini cukup cepat dalam implementasinya pada perangkat lunak. Metode Word Auto Key Encryption juga merupakan salah satu metode yang telah digunakan secara komersial. Metode kriptografi dapat digunakan untuk mengamankan data yang bersifat rahasia agar data tersebut tidak diketahui oleh orang lain yang tidak berkepentingan.

Dewasa ini bidang ilmu kriptografi memiliki kemungkinan aplikasi yang sangat luas, mulai dari bidang militer, telekomunikasi, jaringan computer, keuangan dan perbankan, pendidikan dan singkatnya dimana suatu kerahasiaan data amat diperlukan, disitulah kriptografi memegang peranan penting. Produk – produk yang menggunakan kriptografi sebagai dasarnyapun cukup beragam, mulai dari kartu ATM, E-Commerce, Secure e-mail dan lain – lain.

Kata Kunci *: Pengamanan, Data, Perangkat Lunak, Word Auto Key Encryption*

Abstract

There are so many cases of data stored on computers that are not guaranteed to be safe, sometimes broken / illegible and even lost. This is all due to being infected with a computer virus which is always the latest. This situation makes the data owners uncomfortable. To overcome this anxiety, the author designed a software to study cryptographic methods, the method chosen by the author was the Word Auto Key Encryption (WAKE) method because this method was quite fast in its implementation in software. The Word Auto Key Encryption method is also one method that has been used commercially. Cryptographic methods can be used to secure confidential data so that the data is not known by others who are not interested.

Today the field of cryptography has a very wide possibility of applications, ranging from the fields of military, telecommunications, computer networks, finance and banking, education and in short where data confidentiality is very necessary, that's where cryptography plays an important role. Products that use cryptography as a basis are quite diverse, ranging from ATM cards, E-Commerce, Secure e-mail and others.

Keywords **:** *Security, Data, Software, Word Auto Key Encryption*.

1. PENDAHULUAN

Metode kriptografi dapat digunakan untuk mengamankan data yang bersifat rahasia agar data tersebut tidak diketahui oleh orang lain yang tidak berkepentingan. Metode *Word Auto Key Encryption* merupakan salah satu metode yang telah digunakan secara komersial. *WAKE* merupakan singkatan dari *Word Auto Key Encryption*. Metode *Word Auto Key Encryption* ditemukan oleh David Wheeler pada tahun 1993.

Metode *Word Auto Key Encryption* menggunakan kunci 128 bit, dan sebuah tabel 256 x 32 bit. Dalam algoritmanya, metode *Word Auto Key Encryption* menggunakan operasi XOR, AND, OR dan *Shift Right*. Metode *Word Auto Key Encryption* telah digunakan pada *program* Dr. Solomon Anti Virus versi terbaru. Metode *Word Auto Key Encryption* dapat dibagi menjadi beberapa proses yaitu proses pembentukan tabel dan kunci, enkripsi dan dekripsi. Proses penyelesaian metode *Word Auto Key Encryption* cukup rumit dan sulit untuk dikerjakan secara manual berhubung karena algoritmanya yang cukup panjang dan kompleks.

Dalam ilmu kriptografi, selain metode *Word Auto Key Encryption* masih banyak metode yang dapat digunakan untuk mengamankan data. Setiap metode memiliki kelebihan dan kekurangannya masing-masing. Namun, yang menjadi permasalahan dalam memilih metode kriptografi yang cocok adalah bagaimana mengetahui dan memahami cara kerja dari metode kriptografi tersebut. Oleh karena itu, diperlukan suatu perangkat lunak untuk mempelajari metode kriptografi tersebut. Penulis memilih metode *Word Auto Key Encryption* karena metode ini cukup cepat dalam implementasinya pada perangkat lunak.

2. METODOLOGI PENELITIAN

2.1. Kriptografi

Kriptografi mempunyai sejarah yang panjang. Secara historis, ada 4 kelompok orang yang berkontribusi terhadap perkembangan kriptografi, dimana mereka menggunakan kriptografi untuk menjamin kerahasiaan dalam komunikasi pesan penting, yaitu kalangan militer (termasuk intelijen dan mata-mata), kalangan diplomatik, penulis buku harian, dan pecinta (*lovers)*. Diantara keempat kelompok ini, kalangan militer yang memberikan kontribusi paling penting karena pengiriman pesan di dalam suasana perang membutuhkan teknik enkripsi dan dekripsi yang rumit [3].

Sejarah kriptografi sebagian besar merupakan sejarah kriptografi klasik, yaitu metode enkripsi yang menggunakan kertas dan pensil atau mungkin dengan bantuan alat mekanik sederhana. Secara umum algoritma kriptografi klasik dikelompokkan menjadi dua katagori yaitu, algoritma transposisi atau (*transposition chiper*) dan algoritma substitusi (*substitution chiper*). *Chiper* transposisi mengubah susunan huruf-huruf di dalam pesan, sedangkan *chiper* substitusi mengganti setiap huruf atau kelompok huruf lain [3].

Kriptografi atau yang sering dikenal dengan Sebutan ilmu penyandian data, adalah suatu bidang ilmu seni *(art and science)* yang bertujuan untuk menjaga kerahasiaan suatu pesan yang berupa data data dari akses oleh orang-orang atau pihak-pihak lain yang tidak berhak sehingga tidak menimbulkan kerugian. Bidang ilmu Kriptografi ini semula hanya populer dibidang militer dan bidang intelijen untuk menyandikan pesan- pesan panglima perang kepada pasukan yang berada di garis depan, akan tetapi seiring dengan semakin berkembangnya teknologi utamanya teknologi informasi dan semakin padatnya lalu lintas informasi yang terjadi tentu saja semakin menuntut adanya suatu komunikasi data yang aman, bidang ilmu ini menjadi semakin penting. Sekarang bidang ilmu ini menjadi salah satu isu suatu topik riset yang tidak habis-habisnya diteliti dengan melibatkan banyak peneliti.

Ilmu Kriptografi sebenarnya sudah mulai dipelajari manusia sejak tahun 400 SM, yaitu pada zaman Yunani kuno. Dari catatan bahwa "Penyandian Transposisi" merupakan sistem

kriptografi pertama yang digunakan atau dimanfaatkan. Bidang ilmu ini terus berkembang seiring dengan kemajuan peradaban manusia, dan memegang peranan penting dalam strategi peperangan yang terjadi dalam sejarah manusia, mulai dari sistem kriptografi "*Caesar Chiper*" yang terkenal pada zaman Romawi kuno, "*Playfair Cipher*" yang digunakan Inggris dan "ADFVGX Cipher" yang digunakan Jerman pada Perang Dunia I, hingga algoritma-algoritma kriptografi rotor yang populer pada Perang Dunia II , seperti Sigaba / M-134 (Amerika Serikat), Typex (Inggris), Purple (Jepang), dan mesin kriptografi legendaris Enigma (Jerman) [4]

Induk dari kriptografi sebenarnya adalah matematika, khususnya teori aljabar yang mendasar ilmu bilangan . Oleh karena itu kriptografi semakin berkembang ketika komputer ditemukan. Sebab dengan penemuan komputer memungkinkan dilakukannya perhitungan yang rumit dan komplek dalam waktu yang relatif sangat singkat, suatu hal yang sebelumnya tidak dapat dilakukan. Dari hal tersebut lahirlah banyak teori dan algoritma penyandian data yang semakin kompleks dan sulit dipecahkan.

Dewasa ini bidang ilmu kriptografi memiliki kemungkinan aplikasi yang sangat luas, mulai dari bidang militer, telekomunikasi, jaringan komputer, keuangan dan perbankan, pendidikan dan singkatnya dimana suatu kerahasiaan data amat diperlukan, disitulah kriptografi memegang peranan penting. Produk-produk yamg menggunakan kriptografi sebagai dasarnyapun cukup beragam, mulai dari kartu ATM, *E-Commerce*, *secure e-mail* dan lain-lain.

Menurut *Stalling (Stalling,William, Ph.D, Network and Internetwork Security Prentice Hall, 1995)*, Ada beberapa tuntutan yang terkait dengan isu keamanan data yaitu :

a. Confidentiality
Menjamin bahwa data-data tersebut hanya bisa diakses oleh pihak-pihak tertentu saja.

b. Authentication
Baik pada saat mengirim atau menerima informasi, kedua belah pihak perlu mengetahui bahwa pengirim dari pesan tersebut adalah orang yang sebenarnya seperti yang diklaim.

c. Integrity
Tuntutan ini berhubungan dengan jaminan setiap pesan yang dikirim pasti sampai pada penerimanya tanpa ada bagian dari pesan tersebut yang diganti, diduplikasi, dirusak, diubah urutannya dan ditambahkan.

d. Nonrepudiation
Nonrepudiation mencegah pengirim maupun penerima mengingkari bahwa mereka telah mengirimkan atau menerima suatu pesan/informasi. Jika sebuah pesan dikirim, penerima dapat membuktikan bahwa pesan tersebut memang dikirim oleh pengirim yang tertera. Sebaliknya, jika sebuah pesan diterima, pengirim dapat membuktikan bahwa pesannya telah diterima oleh pihak yang ditujunya.

e. Access Control
Membatasi sumber-sumber data hanya kepada orang-orang tertentu.

f. Availability
Jika diperlukan setiap saat semua informasi pada sistem komputer harus tersedia bagi semua pihak yang berhak atas informasi tersebut.

Dari keenam aspek keamanan data tersebut, empat diantaranya dapat diatasi dengan menggunakan kriptografi yaitu *confidentiality, integrity, authentication,* dan *nonrepudiation.*

Kriptografi dapat didefenisikan sebagai ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek-aspek pada keamanan informasi misalnya kerahasiaan, integritas data, otentikasi pengirim / penerima data, dan otentikasi data. *Cryptanalysis* adalah bidang ilmu dan seni untuk memecahkan *chiperteks. Cryptanalysis merupakan* studi tentang bagaimana mengalahkan (memecahkan) mekanisme kriptografi, dan *cryptology* yang berasal dari kata kryptos dan logos (bahasa Yunani) yang artinya kata tersembunyi, adalah penggabungan disiplin *cryptography* dan *cryptanalysis* [3].

2.2. Sistem Kriptografi

Berdasarkan jumlah kunci yang digunakan, ada dua jenis sistem kriptografi yaitu sistem kriptografi simetris dan sistem kriptografi asimetris. Enkripsi simetris sering juga disebut sebagai enkripsi konvensional atau enkripsi kunci-tunggal (*single key*), karena kunci untuk enkripsi sama dengan kunci untuk dekripsi [4].

Pada model enkripsi simetris ini digunakan algoritma yang sama untuk proses enkripsi/dekripsi dengan memakai satu kunci yang sama. Gambar 2.1 dibawah ini memperlihatkan skema kriptografi simetris.

Sistem kriptografi asimetris biasanya lebih dikenal dengan kriptografi kunci-publik (*public-key cryptography)*. Pada kriptografi asimetris kunci untuk enkripsi tidak rahasia dan dapat diketahui oleh siapapun (diumumkan ke publik), sedangkan kunci untuk dekripsi hanya diketahui oleh penerima pesan (karena itu rahasia). Pada kriptografi ini, setiap orang yang berkomunikasi mempunyai sepasang kunci, yaitu kunci privat dan kunci publik.

Sebuah *block cipher* adalah sebuah fungsi yang memetakan n-bit blok *plaintext* menjadi n-bit *ciphertext*. Fungsi tersebut terdiri dari sebuah algoritma dan sebuah kunci. Hasil pemetaan dari *plaintext* ke *ciphertext* akan berbeda-beda tergantung pada kunci yang digunakan. Baik *cryptography* simetris maupun *cryptography* asimetris bisa merupakan *block cipher*.

Untuk *plaintext* yang panjangnya lebih besar dari n-bit perlu dipilih mode operasi untuk menentukan cara enkripsi/ dekripsi *plaintext* tersebut. Ada beberapa pilihan mode operasi yang bisa diterapkan antara lain *Electronic CodeBook* (*ECB*), *Cipher Block Chaining* (*CBC*), *Cipher FeedBack* (*CFB*), *Output FeedBack* (*OFB*). Keempat mode operasi ini memiliki kelebihan dan kekurangan masing-masing. Untuk penelitian dalam skripsi ini mode operasi yang digunakan hanya ECB dan CBC saja.

2.3. Aplikasi Kriptografi

2.3.1. Privacy

Privacy (kerahasiaan) mungkin merupakan aplikasi paling nyata dari kriptografi. Kriptografi dapat digunakan untuk mengimplementasikan *privacy* hanya dengan mengenkrip informasi yang diinginkan untuk tetap *private*. Agar seseorang dapat membaca data *private* ini dia harus mendekrip terlebih dahulu. Kadang-kadang informasi tertentu bukan untuk diakses oleh siapapun juga, dan dalam hal ini informasi dapat disimpan sedemikian rupa sehingga membalik proses merupakan sesuatu yang secara virtual tidak mungkin.

Misalnya, dalam sistem *multi-user*, tidak ada satu orangpun dimungkinkan untuk mengetahui daftar *password* dari masing-masing *user* dalam sistem. Biasanya nilai hash dari *password* yang disimpan bukan *password* itu sendiri. Hal ini memungkinkan *user* dari sistem yakin betul tentang informasi pribadi disimpan betul-betul aman dari gangguan orang lain karena dengan memasukkan *password* harus diverifikasi terlebih dahulu (dengan menghitung fungsi *hash*-nya dan membandingkan dengan nilai *hash* yang tersimpan).

2.3.2. Digital Signature dan Authentication

Authentication adalah suatu proses untuk membuktikan dan memverifikasi informasi tertentu. Kadang-kadang seseorang ingin memverifikasi asal dokumen, indentitas pengirim, waktu dan tanggal penandatanganan dan/atau pengiriman, identitas komputer atau user dan lain-lain. Suatu *digital signature* adalah cara *cryptography* dimana dengan cara tersebut beberapa hal di atas dapat diverifikasi. Tanda tangan dijital dari suatu dokumen adalah potongan informasi yang didasarkan kepada dokumen dan kunci rahasia penanda-tangan. Tanda tangan ini biasanya diciptakan melalui penggunaan fungsi hash dan fungsi tanda tangan privat (enkripsi kunci rahasia penanda tangan), tetapi masih ada metode lain.

Setiap hari orang menandatangani nama mereka dalam surat, bukti penggunaan kartu kredit, dan dokumen lainnya, yang menunjukkan bahwa mereka setuju dengan isi dokumen tersebut. Dalam hal ini, mereka melakukan otentikasi bahwa mereka dalam kenyataannya adalah pemilik atau pengirim atau sumber dari dokumen. Hal ini memungkinkan orang lain untuk melakukan verifikasi bahwa pesan khusus benar-benar berasal dari si penanda-tangan dokumen. Akan tetapi, cara ini bukanlah bebas dari kemungkinan pencurian atau pemalsuan dari pihak ketiga, karena orang dapat "mengangkat" tanda-tangan dari dalam dokumen dan menempatkannya ke dokumen lain, dengan demikian menghasilkan dokumen asli tapi palsu (aspal). Tanda tangan konvensional (dengan tinta) juga tidak aman dari pemalsuan karena dimungkinkan untuk mereproduksi sebuah tanda tangan pada dokumen lain atau mengubah dokumen setelah dokumen ditanda-tangani. Tanda tangan digital dan tulisan tangan tergantung pada fakta bahwa sulit untuk mendapatkan dua orang dengan tanda tangan sama.

3. HASIL DAN PEMBAHASAN

3.1 Proses Pembentukan Tabel S-Box

Proses pembentukan tabel *S-Box* terdiri atas 8 (delapan) proses utama. Dalam prosesnya, pembentukan tabel *S-Box* memerlukan *input kunci* dengan panjang 128 bit biner atau 16 karakter *ASCII*. Untuk lebih jelas, proses ini dapat dilihat pada contoh berikut :
Misalkan *input key* = 'STMIKBUDIDARMA', maka proses pembentukan tabel *S-Box* dalam heksadesimal adalah sebagai berikut :

A. Inisialisasi nilai TT[0] ... TT[7].
   TT[0] = 726A8F3B (dalam heksadesimal)
   TT[1] = E69A3B5C
   TT[2] = D3C71FE5
   TT[3] = AB3C73D2
   TT[4] = 4D3A8EB3
   TT[5] = 0396D6E8
   TT[6] = 3D4C2F7A
   TT[7] = 9EE27CF3

B. Pecah kunci menjadi 4 kelompok dan masukkan pada T[0] ... T[3].
   Kunci : "STMIKBUDIDARMA"
   Kode *ASCII* dari "' = 39 = 27
   Kode *ASCII* dari 'S' = 83 = 53
   Kode *ASCII* dari 'T' = 84 = 54
   Kode *ASCII* dari 'M' = 77 = 4D
   Kode *ASCII* dari 'I' = 73 = 49
   Kode *ASCII* dari 'K' = 75 = 4B
   Kode *ASCII* dari 'B' = 66 = 42
   Kode *ASCII* dari 'U' = 85 = 55
   Kode *ASCII* dari 'D' = 68 = 44
   Kode *ASCII* dari 'I' = 73 = 49
   Kode *ASCII* dari 'D' = 68 = 44
   Kode *ASCII* dari 'A' = 65 = 41
   Kode *ASCII* dari 'R' = 82 = 52
   Kode *ASCII* dari 'M' = 77 = 4D
   Kode *ASCII* dari 'A' = 65 = 41
   Kode *ASCII* dari "' = 39 = 27
   Kunci (dalam heksa) = 2753544D494B425544494441524D4127
   T[0] = K[0] = 2753544D
   T[1] = K[1] = 494B4255

T[2] = K[2] = 44494441
T[3] = K[3] = 524D4127

C. Untuk n = 4 sampai 255, lakukan prosedur berikut :
n = 4
-> X = T[0] + T[3] = 2753544D + 524D4127 = 79A09574
-> X >> 3 (Shift Right 3 bit) = 79A09574 >> 3 = 0F3412AE
X AND 7 = 79A09574 AND 7(10) = 4
T[4] = X >> 3 XOR TT[X AND 7] = 0F3412AE XOR TT[4] = 420E9C1D
n = 5
-> X = T[1] + T[4] = 494B4255 + 420E9C1D = 8B59DE72
-> X >> 3 (Shift Right 3 bit) = 8B59DE72 >> 3 = 116B3BCE
X AND 7 = 8B59DE72 AND 7(10) = 2
T[5] = X >> 3 XOR TT[X AND 7] = 116B3BCE XOR TT[2] = C2AC242B
n = 6
-> X = T[2] + T[5] = 44494441 + C2AC242B = 06F5686C
-> X >> 3 (Shift Right 3 bit) = 06F5686C >> 3 = 00DEAD0D
X AND 7 = 06F5686C AND 7(10) = 4
T[6] = X >> 3 XOR TT[X AND 7] = 00DEAD0D XOR TT[4] = 4DE423BE (dan seterusnya hingga n = 255).

D. Untuk n = 0 sampai 22, lakukan prosedur berikut :
n = 0
T[0] = T[0] + T[89] = 2753544D + 264A0F22 = 4D9D636F
n = 1
T[1] = T[1] + T[90] = 494B4255 + C306E074 = 0C5222C9
n = 2
T[2] = T[2] + T[91] = 44494441 + F63F71DD = 3A88B61E
n = 3
T[3] = T[3] + T[92] = 524D4127 + C420D8E6 = 166E1A0D

(dan seterusnya hingga n = 22).

E. Set nilai untuk beberapa variabel di bawah ini.
X = AA953396
Z = T[59] OR 01000001 = E6202648 OR 01000001 = E7202649
Z = Z AND FF7FFFFF = E7202649 AND FF7FFFFF = E7202649
X = X AND FF7FFFFF = AA953396 AND FF7FFFFF = 913559DF

F. Untuk n = 0 sampai 255, lakukan prosedur berikut :
n = 0
X = (913559DF AND FF7FFFFF) + E7202649 = 78558028
T[0] = 4D9D636F] AND 00FFFFFF XOR 78558028 = 78C8E347
n = 1
X = (78558028 AND FF7FFFFF) + E7202649 = 5F75A671
T[1] = 0C5222C9] AND 00FFFFFF XOR 5F75A671 = 5F2784B8
n = 2
X = (5F75A671 AND FF7FFFFF) + E7202649 = 4695CCBA
T[2] = 3A88B61E] AND 00FFFFFF XOR 4695CCBA = 461D7AA4
(dan seterusnya hingga n = 255).

G. Set nilai untuk beberapa variabel berikut.
T[256] = T[0] = 78C8E347

X = X AND 255(10) = 915BA2DF AND 255(10) = 000000DF

H. Untuk n = 0 sampai 255, lakukan prosedur berikut.
n = 0
Temp = T[223] XOR X AND 255 = B1DDC264 XOR 000000DF AND 255 = 000000BB
T[0] = T[187] = 35E0E343
T[223] = T[1] = 5F2784B8
n = 1
Temp = T[222] XOR X AND 255 = CA53220D XOR 000000DF AND 255 = 000000D2
T[1] = T[210] = F6EE3574
T[223] = T[2] = 461D7AA4
n = 2
Temp = T[221] XOR X AND 255 = E3B16DBC XOR 000000DF AND 255 = 00000063
T[2] = T[99] = CD1A008E
T[223] = T[3] = 2D5BE90E
(dan seterusnya hingga n = 255).

3.2 Proses Pembentukan Kunci

Proses pembentukan kunci memerlukan *input kunci* dengan panjang 128 bit biner atau 16 karakter *ASCII*. Pertama – tama, *input* kunci dipecah menjadi 4 kelompok dan di-*set* sebagai nilai awal dari variabel A_0, B_0, C_0, D_0. Kemudian isi variabel A, B, C dan D dan ulangi sebanyak n-putaran yang di-*input*.

$A_{i+1} = M(A_i, D_i)$
$B_{i+1} = M(B_i, A_{i+1})$
$C_{i+1} = M(C_i, B_{i+1})$
$D_{i+1} = M(D_i, C_{i+1})$

Fungsi M(X, Y) = (X + Y)>>8 XOR T[(X + Y) AND 255]. Nilai dari D_i merupakan nilai dari kunci K_i. Proses ini dapat dilihat pada contoh berikut :
Misalkan *input key* : 'STMIKBUDIDARMA' dan putaran kunci sebanyak 5 kali putaran, maka proses pembentukan kunci dalam heksadesimal adalah sebagai berikut :

Kunci 'STMIKBUDIDARMA' , diubah dalam bentuk heksa =
2753544D494B425544494441524D4127
Pecah kunci menjadi 4 kelompok dan masukkan ke A(0), B(0), C(0) dan D(0).
A(0) = 2753544D
B(0) = 494B4255
C(0) = 44494441
D(0) = 524D4127

KUNCI PUTARAN 1

FungsiM(A[0],D[0]) = FungsiM(2753544D,524D4127) = (2753544D + 524D4127)>>8 XOR T[(2753544D + 524D4127) AND 255(10)] = 79A09574>>8 XOR T[116] = 0079A095 XOR 9E6B9FC1 = 9E123F54
A[1] = 9E123F54

FungsiM(B[0],A[1]) = FungsiM(494B4255,9E123F54) = (494B4255 + 9E123F54)>>8 XOR T[(494B4255 + 9E123F54) AND 255(10)] = E75D81A9>>8 XOR T[169] = 00E75D81 XOR 9E6B9FC1 = 9E8CC240
B[1] = 9E8CC240

FungsiM(C[0],B[1]) = FungsiM(44494441,9E8CC240) = (44494441 + 9E8CC240)>>8 XOR T[(44494441 + 9E8CC240) AND 255(10)] = E2D60681>>8 XOR T[129] = 00E2D606 XOR C6338DEF = C6D15BE9
C[1] = C6D15BE9

FungsiM(D[0],C[1]) = FungsiM(524D4127,C6D15BE9) = (524D4127 + C6D15BE9)>>8 XOR T[(524D4127 + C6D15BE9) AND 255(10)] = 191E9D10>>8 XOR T[16] = 00191E9D XOR DAF9D741 = DAE0C9DC
D[1] = DAE0C9DC

KUNCI PUTARAN 2

FungsiM(A[1],D[1]) = FungsiM(9E123F54,DAE0C9DC) = (9E123F54 + DAE0C9DC)>>8 XOR T[(9E123F54 + DAE0C9DC) AND 255(10)] = 78F30930>>8 XOR T[48] = 0078F309 XOR 368BFE76 = 36F30D7F
A[2] = 36F30D7F

FungsiM(B[1],A[2]) = FungsiM(9E8CC240,36F30D7F) = (9E8CC240 + 36F30D7F)>>8 XOR T[(9E8CC240 + 36F30D7F) AND 255(10)] = D57FCFBF>>8 XOR T[191] = 00D57FCF XOR B39BE81E = B34E97D1
B[2] = B34E97D1
FungsiM(C[1],B[2]) = FungsiM(C6D15BE9,B34E97D1) = (C6D15BE9 + B34E97D1)>>8 XOR T[(C6D15BE9 + B34E97D1) AND 255(10)] = 7A1FF3BA>>8 XOR T[186] = 007A1FF3 XOR BAAB72E3 = BAD16D10
C[2] = BAD16D10

FungsiM(D[1],C[2]) = FungsiM(DAE0C9DC,BAD16D10) = (DAE0C9DC + BAD16D10)>>8 XOR T[(DAE0C9DC + BAD16D10) AND 255(10)] = 95B236EC>>8 XOR T[236] = 0095B236 XOR 2113B64D = 2186047B
D[2] = 2186047B

KUNCI PUTARAN 3

FungsiM(A[2],D[2]) = FungsiM(36F30D7F,2186047B) = (36F30D7F + 2186047B)>>8 XOR T[(36F30D7F + 2186047B) AND 255(10)] = 587911FA>>8 XOR T[250] = 00587911 XOR CA53220D = CA0B5B1C
A[3] = CA0B5B1C

FungsiM(B[2],A[3]) = FungsiM(B34E97D1,CA0B5B1C) = (B34E97D1 + CA0B5B1C)>>8 XOR T[(B34E97D1 + CA0B5B1C) AND 255(10)] = 7D59F2ED>>8 XOR T[237] = 007D59F2 XOR 029A3569 = 02E76C9B
B[3] = 02E76C9B

FungsiM(C[2],B[3]) = FungsiM(BAD16D10,02E76C9B) = (BAD16D10 + 02E76C9B)>>8 XOR T[(BAD16D10 + 02E76C9B) AND 255(10)] = BDB8D9AB>>8 XOR T[171] = 00BDB8D9 XOR 0A4631DE = 0AFB8907
C[3] = 0AFB8907

FungsiM(D[2],C[3]) = FungsiM(2186047B,0AFB8907) = (2186047B + 0AFB8907)>>8 XOR T[(2186047B + 0AFB8907) AND 255(10)] = 2C818D82>>8 XOR T[130] = 002C818D XOR F7847E4C = F7A8FFC1
D[3] = F7A8FFC1

KUNCI PUTARAN 4

FungsiM(A[3],D[3]) = FungsiM(CA0B5B1C,F7A8FFC1) = (CA0B5B1C + F7A8FFC1)>>8 XOR T[(CA0B5B1C + F7A8FFC1) AND 255(10)] = C1B45ADD>>8 XOR T[221] = 00C1B45A XOR 029A3569 = 025B8133
A[4] = 025B8133

FungsiM(B[3],A[4]) = FungsiM(02E76C9B,025B8133) = (02E76C9B + 025B8133)>>8 XOR T[(02E76C9B + 025B8133) AND 255(10)] = 0542EDCE>>8 XOR T[206] = 000542ED XOR 50310F19 = 50344DF4
B[4] = 50344DF4

FungsiM(C[3],B[4]) = FungsiM(0AFB8907,50344DF4) = (0AFB8907 + 50344DF4)>>8 XOR T[(0AFB8907 + 50344DF4) AND 255(10)] = 5B2FD6FB>>8 XOR T[251] = 005B2FD6 XOR 6D317F71 = 6D6A50A7
C[4] = 6D6A50A7

FungsiM(D[3],C[4]) = FungsiM(F7A8FFC1,6D6A50A7) = (F7A8FFC1 + 6D6A50A7)>>8 XOR T[(F7A8FFC1 + 6D6A50A7) AND 255(10)] = 65135068>>8 XOR T[104] = 00651350 XOR 04591BF2 = 043C08A2
D[4] = 043C08A2

KUNCI PUTARAN 5

FungsiM(A[4],D[4]) = FungsiM(025B8133,043C08A2) = (025B8133 + 043C08A2)>>8 XOR T[(025B8133 + 043C08A2) AND 255(10)] = 069789D5>>8 XOR T[213] = 00069789 XOR CA53220D = CA55B584
A[5] = CA55B584

FungsiM(B[4],A[5]) = FungsiM(50344DF4,CA55B584) = (50344DF4 + CA55B584)>>8 XOR T[(50344DF4 + CA55B584) AND 255(10)] = 1A8A0378>>8 XOR T[120] = 001A8A03 XOR FE015D5F = FE1BD75C
B[5] = FE1BD75C
FungsiM(C[4],B[5]) = FungsiM(6D6A50A7,FE1BD75C) = (6D6A50A7 + FE1BD75C)>>8 XOR T[(6D6A50A7 + FE1BD75C) AND 255(10)] = 6B862803>>8 XOR T[3] = 006B8628 XOR 1B52176B = 1B399143
C[5] = 1B399143

FungsiM(D[4],C[5]) = FungsiM(043C08A2,1B399143) = (043C08A2 + 1B399143)>>8 XOR T[(043C08A2 + 1B399143) AND 255(10)] = 1F7599E5>>8 XOR T[229] = 001F7599 XOR B51E74B3 = B501012A
D[5] = B501012A

KUNCI = D[5] = B501012A

3.3. Proses Enkripsi

Proses enkripsi dari metode WAKE untuk menghasilkan *ciphertext* adalah berupa hasil operasi XOR dari *plaintext* dan 32 bit kunci yang dihasilkan dari proses pembentukan kunci.
Contoh :
Plain Text : 'STMIK BUDIDARMA,TESTING AJA...'
Kunci : AGUS MARISAK..AJA

Plain Text : 'STMIK BUDIDARMA,TESTING AJA...'
Kode *ASCII* dari ''' = 27
Kode *ASCII* dari 'S' = 53
Kode *ASCII* dari 'T' = 54
Kode *ASCII* dari 'M' = 4D
Kode *ASCII* dari 'I' = 49
Kode *ASCII* dari 'K' = 4B
Kode *ASCII* dari ' ' = 20
Kode *ASCII* dari 'B' = 42
Kode *ASCII* dari 'U' = 55
Kode *ASCII* dari 'D' = 44
Kode *ASCII* dari 'I' = 49
Kode *ASCII* dari 'D' = 44
Kode *ASCII* dari 'A' = 41
Kode *ASCII* dari 'R' = 52
Kode *ASCII* dari 'M' = 4D
Kode *ASCII* dari 'A' = 41
Kode *ASCII* dari ',' = 2C
Kode *ASCII* dari 'T' = 54
Kode *ASCII* dari 'E' = 45
Kode *ASCII* dari 'S' = 53
Kode *ASCII* dari 'T' = 54
Kode *ASCII* dari 'I' = 49
Kode *ASCII* dari 'N' = 4E
Kode *ASCII* dari 'G' = 47
Kode *ASCII* dari ' ' = 20
Kode *ASCII* dari 'A' = 41
Kode *ASCII* dari 'J' = 4A
Kode *ASCII* dari 'A' = 41
Kode *ASCII* dari '.' = 2E
Kode *ASCII* dari '.' = 2E
Kode *ASCII* dari '.' = 2E
Kode *ASCII* dari ''' = 27
Plain Text (dalam heksa) =
2753544D494B20425544494441524D412C54455354494E4720414A412E2E2E27
Kunci dari proses pembentukan kunci = AC33A80D
Cipher Text = Plain Text XOR Key
27 XOR AC = 8B = '‹'
53 XOR 33 = 60 = '`'
54 XOR A8 = FC = 'ü'
4D XOR 0D = 40 = '@'
49 XOR AC = E5 = 'å'
4B XOR 33 = 78 = 'x'
20 XOR A8 = 88 = 'ˆ'
42 XOR 0D = 4F = 'O'

55 XOR AC = F9 = 'ù'
44 XOR 33 = 77 = 'w'
49 XOR A8 = E1 = 'á'
44 XOR 0D = 49 = 'I'
41 XOR AC = ED = 'í'
52 XOR 33 = 61 = 'a'
4D XOR A8 = E5 = 'å'
41 XOR 0D = 4C = 'L'
2C XOR AC = 80 = '€'
54 XOR 33 = 67 = 'g'
45 XOR A8 = ED = 'í'
53 XOR 0D = 5E = '^'
54 XOR AC = F8 = 'ø'
49 XOR 33 = 7A = 'z'
4E XOR A8 = E6 = 'æ'
47 XOR 0D = 4A = 'J'
20 XOR AC = 8C = 'Œ'
41 XOR 33 = 72 = 'r'
4A XOR A8 = E2 = 'â'
41 XOR 0D = 4C = 'L'
2E XOR AC = 82 = ','
2E XOR 33 = 1D = '_'
2E XOR A8 = 86 = '†'
27 XOR 0D = 2A = '*'
Hasil proses enkripsi = ‹`ü@åxˆOùwáIíaåL€gí^øzæJŒrâL,_†*

3.4. Proses Dekripsi

Proses dekripsi dari metode WAKE untuk menghasilkan *plaintext* adalah berupa hasil operasi XOR dari *ciphertext* dan 32 bit kunci yang dihasilkan dari proses pembentukan kunci.
Contoh :
Cipher Text : ‹`ü@åxˆOùwáIíaåL€gí^øzæJŒrâL,_†*
Kunci : AGUS MARISAK..AJA

Cipher Text : '‹`ü@åxˆOùwáIíaåL€gí^øzæJŒrâL,_†*'
Kode *ASCII* dari '‹' = 8B
Kode *ASCII* dari '`' = 60
Kode *ASCII* dari 'ü' = FC
Kode *ASCII* dari '@' = 40
Kode *ASCII* dari 'å' = E5
Kode *ASCII* dari 'x' = 78
Kode *ASCII* dari 'ˆ' = 88
Kode *ASCII* dari 'O' = 4F
Kode *ASCII* dari 'ù' = F9
Kode *ASCII* dari 'w' = 77
Kode *ASCII* dari 'á' = E1
Kode *ASCII* dari 'I' = 49
Kode *ASCII* dari 'í' = ED
Kode *ASCII* dari 'a' = 61
Kode *ASCII* dari 'å' = E5
Kode *ASCII* dari 'L' = 4C
Kode *ASCII* dari '€' = 80
Kode *ASCII* dari 'g' = 67

Kode *ASCII* dari 'í' = ED
Kode *ASCII* dari '^' = 5E
Kode *ASCII* dari 'ø' = F8
Kode *ASCII* dari 'z' = 7A
Kode *ASCII* dari 'æ' = E6
Kode *ASCII* dari 'J' = 4A
Kode *ASCII* dari 'Œ' = 8C
Kode *ASCII* dari 'r' = 72
Kode *ASCII* dari 'â' = E2
Kode *ASCII* dari 'L' = 4C
Kode *ASCII* dari ',' = 82
Kode *ASCII* dari '_' = 1D
Kode *ASCII* dari '†' = 86
Kode *ASCII* dari '*' = 2A
Cipher Text (dalam heksa) =
8B60FC40E578884FF977E149ED61E54C8067ED5EF87AE64A8C72E24C821D862A
Kunci dari proses pembentukan kunci = AC33A80D
Plain Text = Cipher Text XOR Key
8B XOR AC = 27 = '''
60 XOR 33 = 53 = 'S'
FC XOR A8 = 54 = 'T'
40 XOR 0D = 4D = 'M'
E5 XOR AC = 49 = 'I'
78 XOR 33 = 4B = 'K'
88 XOR A8 = 20 = ' '
4F XOR 0D = 42 = 'B'
F9 XOR AC = 55 = 'U'
77 XOR 33 = 44 = 'D'
E1 XOR A8 = 49 = 'I'
49 XOR 0D = 44 = 'D'
ED XOR AC = 41 = 'A'
61 XOR 33 = 52 = 'R'
E5 XOR A8 = 4D = 'M'
4C XOR 0D = 41 = 'A'
80 XOR AC = 2C = ','
67 XOR 33 = 54 = 'T'
ED XOR A8 = 45 = 'E'
5E XOR 0D = 53 = 'S'
F8 XOR AC = 54 = 'T'
7A XOR 33 = 49 = 'I'
E6 XOR A8 = 4E = 'N'
4A XOR 0D = 47 = 'G'
8C XOR AC = 20 = ' '
72 XOR 33 = 41 = 'A'
E2 XOR A8 = 4A = 'J'
4C XOR 0D = 41 = 'A'
82 XOR AC = 2E = '.'
1D XOR 33 = 2E = '.'
86 XOR A8 = 2E = '.'
2A XOR 0D = 27 = '''
Hasil proses dekripsi = 'STMIK BUDIDARMA,TESTING AJA...'

4. KESIMPULAN

Pada bagian terakhir ini akan ditemukan kesimpulan yang dapat diperoleh dari pembahasan sebelumnya :

a. Metode Word Auto Key Encryption merupakan salah satu Metode yang telah digunakan secara Komersial.
b. Inti dari WAKE terletak pada proses pembentukan table S-Box dan pembentukan kunci.
c. Proses pembentukan kunci memerlukan input kunci dengan panjang 128 bit biner atau 16 karakter ASCII.
d. Perangkat lunak kriptografi Metode Word Auto Key Encryption (WAKE) dirancang dengan menggunakan bahasa pemograman Microsoft Visual Basic 6.0.
e. Proses Enkripsi pada Metode WAKE adalah Melakukan Operasi Xor dari Plaintext (32 bit) dan kunci (32 bit) untuk menghasilkan Chipertext (32 bit).
f. Implentasi sistem dalam perangkat lunak pembelajaran ini mencakup spesifikasi kebutuhan hardware dan software.

DAFTAR PUSTAKA

[1] Ariyus,Dony., 2005., Computer Security ., Andi Offset, Yogyakarta.
[2] Murni, Aniati, 1992., Pengantar Pengolahan Citra, Elexmedia Komputindo., Jakarta.
[3] Munir, Rinaldi., 2004., Buku Teks Ilmu Komputer Matematika Diskrit Edisi Ketiga, Informatika., Bandung.
[4] Munir, Rinaldi., 2006., Pengolahan Citra Digital dengan Pendekatan Algoritmik., Informatika. Bandung.
[5] Sutoyo. T., 2009., Teori Pengolahan Citra Digital., Andi., Yogyakarta.