Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

Penerapan Algoritma Kriptografi Word Auto Key Encryption Dalam Untuk Pengamanan Soal Ujian Berbasis Komputer

Salma Dola Silalahi, Guidio Leonarde Ginting

Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia
Email: [1]salmasilalahi33@gmail.com
Submitted: **24/09/2020**; Accepted: **08/11/2020**; Published: **27/11/2020**

Abstrak−Ujian Akhir Semester (UAS) berbasis komputer yang dilaksanakan di SMK Swasta Dwiwarna Medan dilakukan dengan bantuan aplikasi berbasis web. Tampilan aplikasi dibagi menjadi dua bagian yaitu web client yang diakses oleh siswa dan web admin yang diakses oleh admin yang memiliki otoritas. Soal ujian adalah salah satu aspek penting untuk menilai kemampuan siswa. Masalah yang terjadi adalah record soal ujian yang disimpan kedalam database ujian oleh admin sangat rentan terhadap pengaksesan dan dapat dibaca oleh siswa di menu admin ketika penginputan soal. Hal ini terjadi karena tidak adanya pengaman record database soal didalam aplikasi web ujian tersebut. Oleh sebab dibutuhkan sebuah teknik pengamanan kriptografi. Teknik kriptografi dapat mengamankan record soal dengan mengacak nilai dari karakter soal yang masuk menjadi nilai yang tidak dimengerti. Penelitian ini akan mengenkripsi record soal database ujian komputerisasi dengan algortima Word Auto Key Encryption (WAKE) pada saat admin menginputkan soal di web admin. Sedangkan proses dekripsi menggunakan algortima WAKE dilakukan sistem aplikasi ketika web client diakses oleh siswa-siswi dengan login dan menekan tombol mulai ujian.

Kata Kunci: Kriptografi, Pengamanan, Soal, Ujian, WAKE

Abstract−The computer-based Final Semester Examination (UAS) held at the Dwiwarna Private Vocational School in Medan was carried out with the help of a web-based application. The application display is divided into two parts, namely the web client accessed by students and the web admin accessed by administrators who have authority. Exam questions are an important aspect of assessing student ability. The problem that occurs is that the record of exam questions stored in the exam database by the admin is very vulnerable to access and can be read by students in the admin menu when inputting questions. This happens because there is no safety record database of questions in the exam web application. Therefore we need a cryptographic security technique. Cryptographic techniques can secure the record of questions by randomizing the value of the incoming question character into a value that is not understood. This research will encrypt the computerized exam database question records with the Word Auto Key Encryption (WAKE) algorithm when the admin enters the questions in the admin web. While the decryption process using the WAKE algorithm is carried out by the application system when the web client is accessed by students by logging in and pressing the start test button.

Keywords: Cryptography, Security, Questions, Exam, WAKE

1. PENDAHULUAN

SMK adalah singkatan dari Sekolah Menengah Kejuruan. SMK memiliki beberapa jurusan terutama dalam bidang IT yaitu Teknik Komputer dan Jaringan serta Rekaya Perangkat Lunak. SMK Swasta Dwiwarna Medan adalah salah satu sekolah yang memiliki kedua jurusan tersebut. SMK Swasta Dwiwarna Medan saat ini telah memasuki pada era teknologi, dimana hampir segala sesuatunya dilakukan dengan komputerisasi. Buktinya ujian nasional yang telah diselenggarakan menggunakan komputer atau lebih dikenal dengan Ujian Nasional Berbasis Komputer (UNBK) yang dimulai tahun 2015. Tidak hanya Ujian Nasional (UN), ujian akhir semester pada jurusan produktif Teknik Komputer dan Jaringan serta Rekayasa Perangkat Lunak juga dilaksanakan menggunakan komputer sebagai media ujianya yang dimulai pada tahun 2017.

Ujian Akhir Semester (UAS) berbasis komputer yang dilaksanakan di SMK Swasta Dwiwarna Medan untuk saat ini hanya soal ujian jurusan Teknik Komputer dan Jaringan dan Rekayasa Perangkat Lunak dengan mode ujian pilihan berganda. Aplikasi yang digunakan untuk menampilkan soal ujian berbasis *web* dan soal ujian disimpan ke dalam *database*. Ujian online berbasis web dimulai dengan peserta masuk/*login* ke aplikasi ujian *online* kemudian mengerjakan semua soal yang diujiankan dan pengumpulan jawaban ujian dilakukan dengan peserta mensubmit semua jawaban yang telah dipilih ke dalam *server*. Kemudian *server* akan memeriksa hasil ujian siswa dan mendapatkan *score* berupa hasil jawaban yang benar atau jumlah jawaban yang salah.

Masalah yang pernah terjadi dengan ujian berbasis *web* adalah ketika proses penyimpanan soal ke dalam *database* yang dilakukan oleh *admin*. Soal yang sudah disimpan dapat dibaca dengan mudah oleh siswa di *interface web admin*, hal tersebut terjadi karena sistem *web* belum mempunyai keamanan enkripsi *database* soal yang mengakibatkan soal dapat dibaca dengan jelas. Masalah lain yang terjadi adalah sistem ujian berbasis *web* pada SMK Swasta Dwiwarna hanya mengandalkan *server* tanpa pengamanan lain, dimana pada dasarnya komputer *server* tidak memeliki konfigurasi keamanan *database*. *Database* soal ujian yang terletak di dalam komputer dapat dengan mudah diakses oleh pihak yang tidak bertanggung jawab jika hanya mengandalkan keamanan dasar dari komputer itu sendiri. Sehingga dibutuhkan sebuah teknik keamanan kriptografi yang dapat meminalisir penyadapan *database* yang dapat dibaca oleh manusia.

Metode kriptografi melakukan pengacakan terhadap data yang diamankan sehingga data tersebut sulit dibaca dan tidak menutup kemungkinan tidak bisa dibaca sama sekali. Salah satu teknik kriptografi yang dapat

digunakan untuk mengamankan *database* yaitu algoritma kriptografi *Word Auto Key Encryption* (WAKE). Metode WAKE menggunakan kunci 128 bit, dan sebuah tabel 256 x 32 bit. Proses algoritma metode ini menggunakan operasi XOR, AND, OR, dan *shiftRigth.*

Beberapa peneliti sudah melakukan penelitian menggunakan algoritma WAKE dan menghasilkan kesimpulan bahwa algoritma WAKE cocok dalam mengenkripsi semua teks karakter. Penelitian yang dilakukan Halasson Gultom (2013), berjudul "Penyandian Email Menggunakan Algoritma Kriptografi WAKE (Word Auto Key Encryption)", bisa didapatkan kesimpulan bahwa algoritma kriptografi WAKE berhasil mengenkripsi sebuah *email* yang berisi karakter teks huruf dan angka. Sedangkan penelitian Eddy (2014) yang berjudul "Pembelajaran Enkripsi Metode Word Auto Key Encryption", memberikan kesimpulan bahwa proses penyelesaian metode WAKE memiliki tingkat keamanan data yang terjamin karena proses metode yang cukup rumit dan sulit dalam hitungan manualnya.

Berdasarkan paparan masalah yang sudah dijelaskan, penelitian ini akan mengamankan *record database* soal ujian berbasis *web* menggunakan algoritma WAKE 128 bit, karena memiliki tingkat keamanan yang komplek dan mode operasi yang cukup banyak. Enkripsi dan dekripsi dilakukan menggunakan bahasa HTML dan PHP. Hasil enkripsi adalah berupa data *ciphertext database* soal yang sudah tidak dapat dibaca oleh penyadapan jika terjadi penyadapan *database*.

2. METODOLOGI PENELITIAN

2.1 Kriptografi

Kriptografi berasal dari bahasa Yunani, *crypto* dan *graphia*.*Crypto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan).Kriptografi adalah sebuah teknik penyandian pesan yang dilakukan agar pesan dapat dikirim dan diterima dengan aman.Kriptografi bertujuan untuk menjaga kerahasian data dan informasi agar tidak disalah gunakan oleh pihak yang tidak sah[1].

2.2 Algoritma Word Auto Key Encryption (WAKE)

WAKE merupakan singkatan dari *Word Auto Key Encryption* ditemukan pertama kali oleh David J. Wheeler pada tahun 1993. Tujuannya adalah untuk mendesain sebuah system enkripsi yang berkeamanan tinggi. Inti dari algoritma WAKE adalah di dalam proses pembentukan S-box dan pembentukan kunci. Metode WAKE menggunakan kunci 128 bit dan sebuah tabel 256 x 32 bit. Dalam algoritmanya, metode WAKEmenggunakan operasi XOR, AND, OR dan Shift Right. Inti dari metode WAKE terletak pada proses pembentukan tabel S-Box dan proses pembentukan kunci [1]. Secara keseluruhan proses dalam algoritma WAKE adalah sebagai berikut [1].

1. Proses pembentukan tabel S-Box
2. Pembentukan kunci
3. Proses enkripsi dan dekripsi

Berikut dibawah ini langkah-langkah di dalam proses pembentukan table S-box [2]

1. Inisialisasi nilai TT[0] … TT[7] :
   TT[0] : 726A8F3B (dalam heksadesimal)
   TT[1] : E69A3B5C (dalam heksadesimal)
   TT[2] : D3C71FE5 (dalam heksadesimal)
   TT[3] : AB3C73D2 (dalam heksadesimal)
   TT[4] : 4D3A8EB3 (dalam heksadesimal)
   TT[5] : 0396D6E8 (dalam heksadesimal)
   TT[6] : 3D4C2F7A (dalam heksadesimal)
   TT[7] : 9EE27CF3 (dalam heksadesimal)
2. Inisialisasi untuk nilai awal T[0]…T[3] :
   T[0] = K[0]
   T[1] = K[1]
   T[2] = K[2]
   T[3] = K[3]
   K[0], K[1], K[2], K[3] merupakan inputan kunci yang dipecah menjadi empat bagian.
3. Untuk T[4] sampai T[255], lakukan proses berikut :
   X = T[n-4] + T[n-1]
   T[n] = X >> 3 XOR TT(X AND 7)
4. Untuk T[0] sampai T[22], lakukan proses berikut :
   T[n] = T[n] + T[n+89]
5. Set nilai untuk beberapa variabel di bawah ini :
   X = T[33]
   Z = T[59] OR (01000001h)
   Z = Z AND (FF7FFFFFh)

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

X = (X AND FF7FFFFFh) + Z
6. Untuk T[0] … T[255], lakukan proses berikut :
   X = (X AND FF7FFFFFh) + Z
   T[n] = T[n] AND 00FFFFFFh XOR X
7. Inisialisasi nilai untuk beberapa variabel berikut ini :
   T[256] = T[0]
   X = X AND 255
8. Untuk T[0] … T[255], lakukan proses berikut :
   Temp = (T[n XOR X] XOR X) AND 255
   T[n] = T[Temp]
   T[X] = T[n+1]

Proses pembentukan kunci dari metode wake dapat ditentukan sendiri yaitu sebanyak n putaran. Semakain banyak putaran dari proses pembentukan kunci, maka keamanan datanya akan semakin terjamin. Fungsi yng digunakan dalam proses pembentukan kunci adalah M(X,Y) = (X+Y) >> 8 XOR T[(X + Y) AND 225]. Pertama-tama kunci yang di input akan dipecah menjadi 4 bagian dan di set sebagian awal dari variabel A_0, B_0, C_0, Dan D_0. Nilai dari variabel ini akan di proses dengan melalui langkah-langkah berikut [1]

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})

Nili dari D_i merupakan milai dari kunci K_i.

Inti dari metode WAKE tidak terletak pada proses enkripsi dan dekripsinya, karena proses enkripsi dan dekripsinya hanya berupa XOR dari *plaintext* dan kunci untuk menghasilkan *chipertext* atau operasi XOR *chipertext* dan kunci untuk menghasilkan *plaintext* [1]. Adapun rumus untuk mendapat hasil dari enkripsi dan dekripsi adalah sebagai berikut:

$P = C \oplus K$
$C = P \oplus K$

Dimana :
P = *Plaintext*
K = Kunci
C = *Chipertext*

2.3 Keamanan

Masalah keamanan merupakan salah satu aspek terpenting pada sebuah sistem informasi. Masalah keamanan sering kali kurang mendapatkan perhatian dari para perancang dan pengolahan sistem informasi serta berada di urutan setelah tampilan, atau bahkan diurutan terakhir dalam daftar yang dianggap penting.Apabila menggagu peforma sistem, sering kali masalah keamanan tidak begitu dipedulikan bahkan ditiadakan [3].

Keamanan data adalah kebebasan dari bahaya atau sebagai kondisi keselamatan.Keamanan secara rinci adalah perlindungan data di dalam sesuatu sistem melawan terhadap otorisasi tidak sah, modifikasi, atau perusakan dan perlindungan sistem komputer terhadap pengguna yang tidak bertanggung jawab [4].

3. HASIL DAN PEMBAHASAN

3.1 Analisa Masalah

Database yang memiliki sifat rahasia sangat rentan terhadap pencurian dan penyadapan oleh pihak yang tidak bertanggung jawab untuk mendapatkan keuntungan kelompok atau pribadi tertentu. Salah satunya adalah sekolah SMK Swasta Dwiwarna Medan yang melakukan ujian akhir sekolah menggunakan komputer untuk jurusan TKJ dan RPL dengan penyimpanan soal di dalam *database*. Siswa yang memiliki kemampuan di bidang IT mungkin dapat menyadap *database* tersebut untuk keuntungan pribadi. Hal tersebut dapat diminalisir dengan teknik pengaman kriptografi.

Berdasarkan rumusan masalah pada bab sebelumnya dan paparan di atas, masalah yang terjadi adalah bagaimana sebuah *database* dengan *record* berisi soal berupa karakter huruf dan angka yang belum disandikan dapat diamankan dengan teknik kriptografi berbasis *web*. Teknik kriptografi akan mengacak data soal menjadi data yang tidak dapat dipahami ketika di baca. Metode yang digunakan dalam pembahasan ini adalah sebuah algoritma kriptografi simetri yaitu algoritma *Word Auto Key Encryption* (WAKE). Pengamanan *database* soal menggunakan algoritma WAKE dilakukan dengan 2 proses, yaitu proses enkripsi untuk mengamankan *database* dengan isi *record* soal didalam menu *admin web* dan proses dekripsi *database* soal didalam menu *user* ketika memulai ujian.

Proses enkripsi untuk mendapatkan *ciphertext* algoritma WAKE adalah dengan membangkitan 255 nilai tabel S-Box yang terdiri dari 4 karakter (32 bit) bilangan hexadesimal melalui operasi AND, XOR dan OR. Tabel S-Box tersebut dikombinasikan dengan nilai hexadesimal kunci untuk mendapatkan *ciphertext* kunci yang

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

digunakan saat proses enkripsi *plaintext*. Sedangkan proses dekripsi dilakukan dengan cara yang sama untuk mendapatkan *plaintext*. Operasi dari enkripsi dan dekripsi algoritma WAKE dilakukan dalam aplikasi berbasis *web*.

3.2 Penerapan Penerapan Algoritma WAKE

Database tabel berisi soal ujian yang akan dienkripsi hanya berupa sampel untuk keperluan hitungan manual. Berikut diuraikan penerapan algoritma WAKE dalam mengenkripsi *database* tabel soal ujian yang berisi karakter huruf dan angka sebagai berikut:

Tabel 1. Sampel Tabel Soal Ujian

id_soal	Soal
1	Komputer adalah
2	2 jaringan

Berdasarkan pada tabel di atas didapati beberapa karakter huruf dan angka yang akan di enkripsi menggunakan algoritma WAKE. Kunci yang digunakan untuk proses enkripsi adalah SALMADOLSILALAHI dengan jumlah putaran sebanyak 2 kali. Jumlah putaran pada proses pembentukan kunci disesuiakan oleh pengguna. Proses enkripsi dibagi menjadi beberapa bagian yaitu, proses pembentukan S-Box dan proses pembentukan kunci.

1. Proses Pembentukan Tabel S-Box
   Sebelum melakukan proses enkripsi terlebih dahulu melakukan proses pembentukan S-Box dan dilanjutkan dengan pembentukan kunci enkripsi dan dekripsi. Adapun proses pembentukan tabel S-Box memiliki 8 tahap untuk mendapatkan nilai T.
   a. Inisialisasi nilai TT[0] … TT[7] :
      TT[0] : 726A8F3B (dalam heksadesimal)
      TT[1] : E69A3B5C (dalam heksadesimal)
      TT[2] : D3C71FE5 (dalam heksadesimal)
      TT[3] : AB3C73D2 (dalam heksadesimal)
      TT[4] : 4D3A8EB3 (dalam heksadesimal)
      TT[5] : 0396D6E8 (dalam heksadesimal)
      TT[6] : 3D4C2F7A (dalam heksadesimal)
      TT[7] : 9EE27CF3 (dalam heksadesimal)
      Nilai TT[0]..TT[7] adalah nilai tetap untuk proses pembentukan S-Box pada algoritma WAKE.
   b. Rubah kunci dalam bentuk Hexadesimal

Tabel 2. Kunci dalam bentuk Hexadesimal

No	Kunci	Kode Ascii
1	S	53
2	A	41
3	L	4C
4	M	4D
5	A	41
6	D	44
7	O	F4
8	L	C5
9	S	34
10	I	94
11	L	C4
12	A	41
13	L	C4
14	A	41
15	H	48
16	I	49

SALMADOLSILALAHI = 53414C4D41444F4C53494C414C414849
kemudian bagi menjadi 4 bagian seperti kententuan di bawah ini :
T[0] = K[0] = 53414C4D
T[1] = K[1] = 41444F4C
T[2] = K[2] = 53494C41
T[3] = K[3] = 4C414849

   c. Untuk mendapatkan nilai T[4] (n=4) hingga T[255] (n=255) maka dilakukan dengan langkah sebagai berikut :
      X = T[n-4] + T[n-1]

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

T[n] = X >> 3 XOR TT(X AND 7)
n = 4
➔ X = T[4-4] + T[4-1]
➔ X = T[0] + T[3] = 53414C4D + 4C414849 = 9F829496
➔ X >> 3 (*ShiftRight* 3 bit) = 9F829496 >> 3 = 13F05292
X AND 7 = 9F829496 AND 7 = 6
➔ T[4] = X >> 3 XOR TT[6] = 13F05292 XOR 3D4C2F7A = 2EBC7DE8

n = 5
➔ X = T[5-4] + T[5-1]
➔ X = T[1] + T[4] = 41444F4C + 2EBC7DE8= 7000CD34
➔ X >> 3 (*ShiftRight* 3 bit) = 7000CD34 >> 3 = 0E0019A6
X AND 7 = 7000CD34 AND 7 = 4
➔ T[5] = X >> 3 XOR TT[4] = 0E0019A6 XOR 4D3A8EB3 = 433A9715

n = 6
➔ X = T[6-4] + T[6-1]
➔ X = T[2] + T[5] = 53494C41+ 433A9715= 9683E356
➔ X >> 3 (*ShiftRight* 3 bit) = 9683E356 >> 3 = 12D07C6A
X AND 7 = 9683E356 AND 7 = 6
➔ T[6] = X >> 3 XOR TT[6] = 12D07C6A XOR 3D4C2F7A = 2F9C5310

n = 7
➔ X = T[7-4] + T[7-1]
➔ X = T[3] + T[6] = 4C414849+ 2F9C5310= 7BDD9B59
➔ X >> 3 (*ShiftRight* 3 bit) = 7BDD9B59>> 3 = 0F7BB36B
X AND 7 = 7BDD9B59 AND 7 = 1
➔ T[7] = X >> 3 XOR TT[1] = 0F7BB36B XOR E69A3B5C = E9E18837

n = 8
➔ X = T[8-4] + T[8-1]
➔ X = T[4] + T[7] = 2EBC7DE8+ E9E18837= 1189E061F
➔ X >> 3 (*ShiftRight* 3 bit) = 1189E061F >> 3 = 2313C0C3
X AND 7 = 1189E061F AND 7 = 7
➔ T[8] = X >> 3 XOR TT[7] = 2313C0C3 XOR 9EE27CF3 = BDF1BC30

n = 9
➔ X = T[9-4] + T[9-1]
➔ X = T[5] + T[8] = 433A9715 + BDF1BC30= 1012C5345
➔ X >> 3 (*ShiftRight* 3 bit) = 1012C5345 >> 3 = 20258A68
X AND 7 = 1012C5345 AND 7 = 5
➔ T[9] = X >> 3 XOR TT[5] = 20258A68 XOR 0396D6E8 = 23B35C80

n = 10
➔ X = T[10-4] + T[10-1]
➔ X = T[6] + T[9] = 2F9C5310 + 23B35C80= 534FAF90
➔ X >> 3 (*ShiftRight* 3 bit) = 534FAF90 >> 3 = 0A69F5F2
X AND 7 = 534FAF90 AND 7 = 0
➔ T[10] = X >> 3 XOR TT[0] = 0A69F5F2 XOR 726A8F3B = 78037AC9

Proses yang sama dilakukan hingga n =255 sebagai berikut :
n = 255
➔ X = T[255-4] + T[255-1]
➔ X = T[251] + T[254] = CD75EE68 + 9FB0B037= 16D269E9F
➔ X >> 3 (*ShiftRight* 3 bit) = 16D269E9F >> 3 = 2DA4D3D3
X AND 7 = 16D269E9F AND 7 = 7
➔ T[255]=X>> 3 XOR TT[7] = 2DA4D3D3 XOR 9EE27CF3 = B346AF20

d. Langkah ke 4 adalah mencari nilai T[0] (n=0) hingga T[22] (n=22) dengan kententuan sebagai berikut :
T[n] = T[n] + T[n+89]
n =0
T[0] = [0] + T[89] = 53414C4D + A09768B1 = F3D8B4FE
n =1

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

T[1] = [1] + T[90] = 41444F4C + 51806C01 = 92C4BB4D
n =2
T[2] = [2] + T[91] = 53494C41 + 10B1A090 = 63FAECD1
n =3
T[3] = [3] + T[92] = 4C414849 + A0903411 = ECD17C5A
n =4
T[4] = [4] + T[93] = 2EBC7DE8 + 2C8A6770 = 5B46E558
Proses yang sama dilakukan hingga n=22 sebagai berikut :
n =22
T[22] = [22] + T[111] = 90AC89B1 + 13410833 = A3ED91E4

e. Langkah kelima melakukan set nilai bariabel sebagai berikut :
X = T[33] = 31A0B108
Z = T[59] OR 01000001 = 6577F190 OR 01000001 = 6477F191
Z = Z AND FF7FFFFF = 6477F191 AND FF7FFFFF = 6477F191
X = X AND FF7FFFFF = 81A0B108 AND FF7FFFFF = 3120B108

f. Langkah keenam kembali mencari nilai T[0] (n=0) hingga T[255] (0=255) dengan ketentuan sebagai berikut :
X = (X AND FF7FFFFF) + Z
T[n] = T[n] AND 00FFFFFF XOR X
n = 0
X = (3120B108 AND FF7FFFFF) + 6477F191 = 9598A299
T[0] = F3D8B4FE AND 00FFFFFF XOR 9598A299 = 95401667

n = 1
X = (9598A299 AND FF7FFFFF) + 6477F191 = F990942A
T[1] = 92C4BB4D AND 00FFFFFF XOR F990942A= F9542F67

n = 2
X = (F990942A AND FF7FFFFF) + 6477F191 = 5D8885BB
T[2] = 63FAECD1 AND 00FFFFFF XOR 5D8885BB = 5D72696A

n = 3
X = (5D8885BB AND FF7FFFFF) + 6477F191 = C180774C
T[3] = ECD17C5A AND 00FFFFFF XOR C180774C= C1510B16

n = 4
X = (C180774C AND FF7FFFFF) + 6477F191 = 257868DD
T[4] = 5B46E558 AND 00FFFFFF XOR 257868DD = 253E8D85
Lakukan langkah berikut dengan cara yang sama ini hingga nilai n=255 seperti di bawah ini :
n = 255
X = (656CBE01 AND FF7FFFFF) + 6477F191 = C9E4AF92
T[255] = B346AF20 AND 00FFFFFF XOR C9E4AF92 = C9A200B2

g. Langkah ketujuh melakukan set nilai variable dengan ketentuan sebagai berikut :
T[256] = T[0] = 95401667
X = X AND 255 = C9E4AF92 AND $255_{(10)}$ = 00000092 (desimal 146)

h. Langkah kedelapan, ini adalah langkah terakhir dalam pembentukan tabel S-Box algoritma WAKE untuk mendapatkan nilai T[0] (n=0) hingga T[255] (n=255) dengan kententuan sebagai berikut :
Temp = (T[n XOR X] XOR X) AND 255
T[n] = T[Temp]
T[X] = T[n+1]
n = 0
Temp = (T[0 XOR 00000092] XOR 00000092) AND $255_{(10)}$
Temp = T[146] XOR 00000092 AND AND $255_{(10)}$
Temp = 10A19BC6 XOR 00000092 AND $255_{(10)}$ = 00000054 (desimal 84)
T[0] = T[84] = 5C90117B
T[146] = T[1] = F9542F67

n = 1
Temp = (T[1 XOR 00000092] XOR 00000092) AND $255_{(10)}$
Temp = T[147] XOR 00000092 AND AND $255_{(10)}$
Temp = 8790A165 XOR 00000092 AND $255_{(10)}$ = 000000F7 (desimal 247)
T[1] = T[247] = B0790A10

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

T[146] = T[2] = 5D72696A

n = 2
Temp = (T[2 XOR 00000092] XOR 00000092) AND $255_{(10)}$
Temp = T[144] XOR 00000092 AND AND $255_{(10)}$
Temp = 65908810 XOR 00000092 AND $255_{(10)}$ = 00000082 (desimal 130)
T[2] = T[130] = D8819065
T[146] = T[3] = C1510B16

n = 3
Temp = (T[3 XOR 00000092] XOR 00000092) AND $255_{(10)}$
Temp = T[145] XOR 00000092 AND AND $255_{(10)}$
Temp = 35901180 XOR 00000092 AND $255_{(10)}$ = 00000012 (desimal 18)
T[3] = T[18] = 1AB07701
T[146] = T[4] = 253E8D85

Proses yang sama dilakukan hingga nilai n=255 sebagai berikut :
n = 255
Temp = (T[255 XOR 00000092] XOR 00000092) AND $255_{(10)}$
Temp = T[163] XOR 00000092 AND AND $255_{(10)}$
Temp = BC1019F1 XOR 00000092 AND $255_{(10)}$ = 00000063 (desimal 99)
T[255] = T[99] = E10865A1
T[146] = T[256] = 95401667
Sehingga didapat nilai T[0] hingga T[255] keseluruhan yang dapat dilihat di dalam tabel di bawah ini :

Tabel 3. Nilai Akhir Pembentukan S-Box

T[n]	Nilai	T[n]	Nilai	T[n]	Nilai
T[0]	5C90117B	T[85]	7793B24A	T[170]	5E213B26
T[1]	B0790A10	T[86]	B3518301	T[171]	78B311F1
T[2]	D8819065	T[87]	2878C341	T[172]	554E3112
T[3]	1AB07701	T[88]	F55790A7	T[173]	99B6665E
T[4]	05F2A1B5	T[89]	324B2761	T[174]	1B345E17
T[5]	1D08023C	T[90]	4714C23E	T[175]	B3561900
T[6]	5976F5AE	T[91]	79A4531B	T[176]	D45167E3
T[7]	4B32C376	T[92]	B128C538	T[177]	3F561A21
T[8]	B7E15163	T[93]	F6292C68	T[178]	E789B679
T[9]	5618CB1C	T[94]	347A14B1	T[179]	871C344B
T[10]	6D78935A	T[95]	428B688A	T[180]	F3516339
T[11]	47D498E4	T[96]	90B67C14	T[181]	80C251E2
T[12]	9B58ECF3	T[97]	C3178B55	T[182]	0C16C316
T[13]	1436D3D7	T[98]	0367C234	T[183]	7D155E20
T[14]	341D62D9	T[99]	78B5621A	T[184]	9B157F11
T[15]	708C564B	T[100]	83B54E61	T[185]	6E37903A
T[16]	BB3F7BF5	T[101]	D437902A	T[186]	D4317701
T[17]	95E5E920	T[102]	67D234A9	T[187]	B7823C18
T[18]	5611CB1C	T[103]	90C15891	T[188]	8C34A139
T[19]	F45044D5	T[104]	46B3537B	T[189]	B6608C43
T[20]	9286B200	T[105]	891B6136	T[190]	90A10B25
T[21]	47D498E5	T[106]	741C3250	T[191]	F64A3259
T[22]	227B060F	T[107]	12D436B1	T[192]	536D3418
T[23]	2B4C3474	T[108]	489B538F	T[193]	9D456E41
T[24]	42619B51	T[109]	90A25F34	T[194]	56B413A4
T[25]	708C564B	T[110]	8A435B73	T[195]	28A23C10
T[26]	1D01023C	T[111]	7F363901	T[196]	57B26E90
T[27]	341D62D9	T[112]	F56E4360	T[197]	6E4552F6
T[28]	0EC3D004	T[113]	1189B441	T[198]	B1F18965
T[29]	ACF4B98D	T[114]	565519C0	T[199]	9810AB11
T[30]	70BC486D	T[115]	67A2139B	T[200]	7B411E48
T[31]	4C5673A3	T[116]	8002A6E0	T[201]	67C31202
T[32]	80B3670C	T[117]	17E34A58	T[202]	C56A2310
T[33]	2A6C7759	T[118]	73B61990	T[203]	D4515C32
T[34]	145D678C	T[119]	B638121C	T[204]	B788C351
T[35]	A5473C90	T[120]	F378F67A	T[205]	5A443B16
T[36]	897B7A32	T[121]	31E63101	T[206]	A92E4156
T[37]	6A34C567	T[122]	37B61899	T[207]	D3561B71

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

T[n]	Nilai	T[n]	Nilai	T[n]	Nilai
T[38]	B37821A9	T[123]	B71735F5	T[208]	8C5E312F
T[39]	90F58021	T[124]	233C57E5	T[209]	65881075
T[40]	F549C671	T[125]	11E97780	T[210]	48C31B42
T[41]	125C78B8	T[126]	90C3773A	T[211]	800C43A1
T[42]	3E2090D2	T[127]	D458801A	T[212]	90B64E27
T[43]	D035A11B	T[128]	00A231C5	T[213]	45B78C16
T[44]	6A89B580	T[129]	75C32212	T[214]	F5590C12
T[45]	77C90B12	T[130]	456B443E	T[215]	876C312E
T[46]	48414C53	T[131]	065B3A24	T[216]	66B427C1
T[47]	1291B1B2	T[132]	8865B41A	T[217]	4D2D2651
T[48]	F10A1189	T[133]	60C42A54	T[218]	5B322C12
T[49]	13F052F2	T[134]	4567C321	T[219]	98F346EA
T[50]	27E0A5E1	T[135]	864E3102	T[220]	C35B4611
T[51]	444F9F82	T[136]	B785154A	T[221]	9810AB11
T[52]	9F829981	T[137]	D5778311	T[222]	1907D110
T[53]	F3C72EE9	T[138]	90F56326	T[223]	09D251F3
T[54]	A7F2C4D3	T[139]	9C642A62	T[224]	48B849F4
T[55]	8E6D6930	T[140]	F9640021	T[225]	3C36810A
T[56]	B3F8A627	T[141]	0534A129	T[226]	66C351A2
T[57]	509D567A	T[142]	086C6521	T[227]	F66A2910
T[58]	15D4907C	T[143]	88B541A5	T[228]	D88C1562
T[59]	30B789E5	T[144]	90C45E21	T[229]	48B378A1
T[60]	10D5F619	T[145]	57B431A9	T[230]	56C16379
T[61]	0D45C312	T[146]	95401667	T[231]	C3188E22
T[62]	90A4B671	T[147]	44C651A1	T[232]	957B235A
T[63]	00C11B3F	T[148]	B8431A54	T[233]	5A2177B2
T[64]	5F892231	T[149]	7C3100C1	T[234]	8B56100F
T[65]	89245C12	T[150]	B0701811	T[235]	34C128A1
T[66]	125D7863	T[151]	6C455001	T[236]	F63690A2
T[67]	0341A783	T[152]	33A12E58	T[237]	70B45588
T[68]	19547D52	T[153]	986C134A	T[238]	58B418F2
T[69]	2849C561	T[154]	5A1224C0	T[239]	08B471E1
T[70]	88371B74	T[155]	4532C318	T[240]	78B129C2
T[71	B6183061	T[156]	00B5431E	T[241]	37F3190C
T[72]	99316D13	T[157]	51C651A0	T[242]	F546C128
T[73]	B299E713	T[158]	84C5521A	T[243]	969B92F1
T[74]	53C31471	T[159]	74C41A40	T[244]	009A26B2
T[75]	A731978C	T[160]	996E3195	T[245]	8694B219
T[76]	29278B21	T[161]	537F4380	T[246]	D442600C
T[77]	C531768A	T[162]	11A466C4	T[247]	20A3C537
T[78]	789F6137	T[163]	08096B54	T[248]	89188F61
T[79]	7420E739	T[164]	800F6439	T[249]	9039B0A2
T[80]	E7802C46	T[165]	8432A41C	T[250]	7B41D533
T[81]	52D51892	T[166]	9976B6531	T[251]	204B7489
T[82]	981D4A13	T[167]	07B32A10	T[252]	4C313890
T[83]	B0528313	T[168]	97B5531A	T[253]	B47810A1
T[84]	998A2132	T[169]	996B4321	T[254]	129B2413
				T[255]	E10865A1

2. Pembentukan Kunci
   Proses selanjutnya adalah pembentukan kunci yang dilakukan sebnayak 2 kali putaran. Kunci yang digunakan dalam keperluan hitungan manual adalah SALMADOLSILALAHI yang dirubah kedalam bentuk hexadesimal 53414C4D41444F4C53494C414C414849. Pertama-tama kunci yang di input akan dipecah menjadi 4 bagian dan di set sebagian awal dari variabel A_0, B_0, C_0, Dan D_0.
   A0 = 53414C4D
   B0 = 41444F4C
   C0 = 53494C41
   D0 = 4C414849
   Nilai dari variabel ini akan di proses dengan melalui langkah-langkah berikut:
   $A_{i+1} = M(A_i, D_i)$
   $B_{i+1} = M(B_i, A_{i+1})$
   $C_{i+1} = M(C_i, B_{i+1})$
   $D_{i+1} = M(D_i, C_{i+1})$
   Fungsi yng digunakan dalam proses pembentukan kunci adalah M(X,Y) = (X+Y) >> 8 XOR T[(X + Y) AND 225].
   Adapun proses pembentukan kunci dimulai dari ronde pertama hingga ronde kedua.

a. Ronde 1
M(A0, D0) = M(53414C4D, 4C414849) = (53414C4D + 4C414849) >> 8 XOR T[(53414C4D + 4C414849) AND 255]
= 9F829496 >> 8 XOR T[9F829496 AND 255]
= 9F829496 >> 8 XOR T[150]
= 009F8294 XOR B0701811 = B0EF9A85
A[1] = B0EF9A85

M(B0, A1) = M(41444F4C, B0EF9A85) = (41444F4C + B0EF9A85) >> 8 XOR T[(41444F4C + B0EF9A85) AND 255]
= F233E9D1 >> 8 XOR T[F233E9D1 AND 255]
= F233E9D1 >> 8 XOR T[209]
= 00F233E9 XOR 65881075 = 657A239C
B[1] = 657A239C

M(C0, B1) = M(53494C41, 657A239C) = (53494C41+ 657A239C) >> 8 XOR T[(53494C41+ 657A239C) AND 255]
= B8C36FDD>> 8 XOR T[B8C36FDD AND 255]
= B8C36FDD>> 8 XOR T[221]
= 00B8C36F XOR 9810AB11 = 98A8687E
C[1] = 98A8687E

M(D0, C1) = M(4C414849, 98A8687E) = (53494C41+ 98A8687E) >> 8 XOR T[(53494C41+ 98A8687E) AND 255]
= 4C414849>> 8 XOR T[4C414849 AND 255]
= 4C414849>> 8 XOR T[199]
= 004C4148 XOR 9810AB11 = 985CEA59
D[1] = 985CEA59

b. Ronde 2
M(A1, D1) = M(B0EF9A85, 985CEA59) = (B0EF9A85+ 985CEA59) >> 8 XOR T[(B0EF9A85+ 985CEA59) AND 255]
= 494C84DE>> 8 XOR T[494C84DE AND 255]
= 494C84DE>> 8 XOR T[222]
= 00494C84 XOR 1907D110 = 194E9D94
A[2] = 194E9D94

M(B1, A2) = M(657A239C, 194E9D94) = (657A239C + 194E9D94) >> 8 XOR T[(657A239C + 194E9D94) AND 255]
= 7EC8C130 >> 8 XOR T[7EC8C130 AND 255]
= 7EC8C130 >> 8 XOR T[48]
= 007EC8C1 XOR F10A1189 = F174D948
B[2] = F174D948

M(C1, B2) = M(98A8687E, F174D948) = (98A8687E + F174D948) >> 8 XOR T[(98A8687E + F174D948) AND 255]
= 8A1D41C6 >> 8 XOR T[8A1D41C6 AND 255]
= 8A1D41C6 >> 8 XOR T[198]
= 008A1D41 XOR B1F18965 = B17B9424
C[2] = B17B9424

M(D1, C2) = M(985CEA59, B17B9424) = (985CEA59 + B17B9424) >> 8 XOR T[(985CEA59 + B17B9424) AND 255]
= 49D87E7D >> 8 XOR T[49D87E7D AND 255]
= 49D87E7D >> 8 XOR T[125]
= 0049D87E XOR 11E97780 = 11A0AFFE
D[2] = 11A0AFFE
Berdasarkan pada putaran kunci ronde terakhir (2) di dapatkan nilai D[2] yang akan menjadi kunci untuk enkripsi dengan *plaintext*. Sehingga kunci dari enkripsi *plaintext* untuk keperluan hitungan manual ini adalah:
Kunci = D[2] = 11A0AFFE

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

3. Enkripsi berdasarkan Algoritma WAKE
   Proses enkripsi hanya dapat dilakukan ketika proses pembentukan tabel S-Box dan pembentukan kunci sesuai ronde sudah dilakukan. Adapun *plaintext* soal yang akan dienkripsi menggunakan algoritma WAKE adalah :
   a. Komputer adalah
   b. 2 jaringan
   Kedua *plaintext* tersebut dirubah kedalam bentuk hexadesimal menggunakan kode ASCII. Adapun tabel ASCII *plaintext* 1 dapat dilihat seperti pada tabel di bawah ini :

Tabel 4. Kode ASCII Plaintext 1

No	Plaintext	Kode ASCII
1	K	4B
2	o	6F
3	m	6D
4	p	70
5	u	75
6	t	74
7	e	65
8	r	72
9	SPACE	20
10	a	61
11	d	64
12	a	61
13	l	6C
14	a	61
15	h	68

Berdasarkan pada tabel di atas didapati nilai hexadesimal dari *plaintext* yaitu, 4B, 6F, 6D, 70, 75, 74, 65, 72, 20, 61, 64, 61, 6C, 61, 68. Nilai hexadesimal *plaintext* tersebut akan dienkripsi dengna kunci algrotima WAKE yaitu 11A0AFFE. Proses enkripsi dilakukan dengan XOR nilai *plaintext* dengan nilai kunci secara berurutan sehingga didapati ciphertext dari soal pertama seperti pada tabel di bawah ini :

Tabel 5. Proses Enkripsi Plaintext 1

No	Proses Enkripsi Plaintext	Hexadesimal Ciphertext	Ciphertext
1	4B XOR 11	5A	Z
2	6F XOR A0	CF	Ï
3	6D XOR AF	C2	Â
4	70 XOR FE	8E	Ž
5	75 XOR 11	64	D
6	74 XOR A0	D0	Đ
7	65 XOR AF	CA	Ê
8	72 XOR FE	8C	OE
9	20 XOR 11	31	1
10	61 XOR A0	C1	Á
11	64 XOR AF	CB	Ë
12	61 XOR FE	9F	Ÿ
13	6C XOR 11	7D	}
14	61 XOR A0	C1	Á
15	68 XOR AF	C7	Ç

Berdasarkan pada tabel proses enkripsi di atas didapati sebuah *ciphertext* dari soal nomor 1 pada *plaintext*. *Ciphertext* dari hasil enkripsi soal nomor 1 sebagai berikut :
“ZÏÂŽdĐÊOE1ÁËŸ}ÁÇ”
Proses selanjutnya adalah enkripsi *plaintext* soal nomor kedua menggunakan algoritma WAKE. Adapun tabel ASCII *plaintext* 2 dapat dilihat seperti pada tabel di bawah ini :

Tabel 6. Kode ASCII Plaintext 2

No	Plaintext	Kode ASCII
1	2	02
2	SPACE	20
3	j	6A

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

No	Plaintext	Kode ASCII
4	a	61
5	r	72
6	i	69
7	n	6E
8	g	67
9	a	61
10	n	6E

Berdasarkan pada tabel di atas didapati nilai hexadesimal dari *plaintext* yaitu, 02, 20, 6A, 61, 72, 69, 6E, 67, 61, 6E. Nilai hexadesimal plaintext tersebut akan dienkripsi dengna kunci algrotima WAKE yaitu 11A0AFFE. Proses enkripsi dilakukan dengan XOR nilai *plaintext* dengan nilai kunci secara berurutan sehingga didapati *ciphertext* dari soal pertama seperti pada tabel di bawah ini :

Tabel 7. Proses Enkripsi Plaintext 2

No	Proses Enkripsi Plaintext	Hexadesimal Ciphertext	Ciphertext
1	02 XOR 11	13	DC3
2	20 XOR A0	80	€
3	6A XOR AF	C5	Å
4	61 XOR FE	9F	Ÿ
5	72 XOR 11	63	c
6	69 XOR A0	C9	É
7	6E XOR AF	C1	Á
8	67 XOR FE	99	™
9	61 XOR 11	70	P
10	6E XOR A0	CE	Î

Berdasarkan pada tabel proses enkripsi di atas didapati sebuah *ciphertext* dari soal nomor 2 pada plaintext. *Ciphertext* dari hasil enkripsi soal nomor 2 sebagai berikut "DC3€ÅŸcÉÁ™pÎ".
Adapun hasil enkripsi dari kedua sampel soal *database* dapat dilihat pada tabel di bawah ini :

Tabel 8. Ciphertext Soal Sampel

id_soal	Soal
1	ZÏÂŽdÐÊOE1ÁËŸ}ÁÇ
2	DC3€ÅŸcÉÁ™pÎ

Berdasarkan pada tabel di atas dapat diambil kesimpulan bahwa aproses enkripsi algoritma WAKE dapat merubah karakter soal yang dapat di baca menjadi tidak dapat dibaca bahkan dipahami oleh manusia. Hal ini dapat meminalisir kebocoran soal yang dapat di baca dikalangan siswa-siswi sekolah.

4. Proses Dekripsi Berdasarkan Algoritma WAKE
Proses dekripsi diperlukan untuk mengembalikan soal yang berbentuk *ciphertext* menjadi soal *plaintext* kembali. Adapun proses dekripsi menggunakan kunci yang sama saat proses enkripsi yaitu SALMADOLSILALAHI dengan jumlah putaran pembentukan kunci sebanyak 2 kali putaran. Proses pertama dekripsi adalah pembentukan tabel S-Box dan pembentukan kunci. Proses tersebut sama dengan proses dekripsi sehingga menghasilkan nilai S-Box dan kunci yang sama. Adapun *ciphertext* soal yang akan dienkripsi sebagai berikut :

Tabel 9. *Ciphertext* Soal Hasil Enkripsi

id_soal	Soal
1	ZÏÂŽdÐÊOE1ÁËŸ}ÁÇ
2	DC3€ÅŸcÉÁ™Pî

Berdasarkan pada tabel soal di atas proses dekripsi pertama terlebih dahulu dilakukan dengan *ciphertext* soal nomor 1 yaitu ZÏÂŽdÐÊOE1ÁËŸ}ÁÇ. *Ciphertext* dirubah kembali kedalam nilai hexadesimal. Kunci yang digunakan proses dekripsi adalah 11A0AFFE didapat dari proses pembangkitan kunci. Adapun proses dekripsi dari *ciphertext* soal nomor 1 dapat dilihat pada tabel di bawah ini :

Tabel 10. Proses Dekripsi Ciphertext Soal 1

No	Proses Dekripsi Ciphertext	Hexadesimal Palintext	Plaintext
1	5A XOR 11	4B	K
2	CF XOR A0	6F	o

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

No	Proses Dekripsi Ciphertext	Hexadesimal Palintext	Plaintext
3	C2 XOR AF	6D	m
4	8E XOR FE	70	p
5	64 XOR 11	75	u
6	D0 XOR A0	74	t
7	CA XOR AF	65	e
8	8C XOR FE	72	r
9	31 XOR 11	20	SPACE
10	C1 XOR A0	61	A
11	CB XOR AF	64	D
12	9F XOR FE	61	A
13	7D XOR 11	6C	L
14	C1 XOR A0	61	A
15	C7 XOR AF	68	H

Berdasarkan pada tabel dekripsi *ciphertext* soal 1 di atas, di dapati kembali *plaintext* soal yaitu "Komputer adalah". Adapun proses dekripsi untuk *ciphertext* soal nomor 2 dapat dilihat pada tabel di bawah ini :

Tabel 11. Proses Dekripsi *Ciphertext* Soal 2

No	Proses Dekripsi Ciphertext	Hexadesimal Plaintext	Plaintext
1	13 XOR 11	02	2
2	80 XOR A0	20	SPACE
3	C5 XOR AF	6A	J
4	9F XOR FE	61	A
5	63 XOR 11	72	R
6	C9 XOR A0	69	I
7	C1 XOR AF	6E	N
8	99 XOR FE	67	G
9	70 XOR 11	61	A
10	CE XOR A0	6E	N

Berdasarkan pada tabel dekripsi *ciphertext* soal 2 di atas, di dapati kembali *plaintext* soal nomor 2 yaitu "2 jaringan". Sehingga hasil dekripsi *ciphertext* soal keseluruhan dapat dilihat pada tabel di bawah ini :

Tabel 12. *Plaintext* Soal Hasil Dekripsi

id_soal	Soal
1	Komputer adalah
2	2 jaringan

Dekripsi menggunakan algoritma WAKE dapat mengembalikan soal hasil enkripsi menjadi *plaintext* berupa soal awal yang dapat di baca dan dipahami artinya.

3.3 Implementasi Program

Tampilan program aplikasi dibagi menjadi 2 bagian yaitu tampilan proses *input* dan tampilan proses *output*. Tampilan input adalah tampilan gambar yang ditampilkan untuk menyesusaikan aplikasi dengan rancangan aplikasi pada bab sebelumnya. Tampilan *output* program adalah proses enkripsi dan dekripsi soal serta tampilan hasil proses enkripsi dan dekripsi soal ujian. Tampilan *input* program terdiri dari tampilan *web admin* dan tampilan *web client* yang sesuai dengan rancangan *form* pada bab sebelumnya. Adapun tampilan program aplikasi berbasis *web* sebagai berikut :

Tampilan *web admin* adalah tampilan program aplikasi yang hanya dapat diakses oleh *admin* yang memiliki otoritas, Tampilan *web admin* terdiri dari beberapa menu yaitu, menu kelola soal, menu *add admin*, menu tes ujian, menu daftar *user*, menu hasil tes ujian, dan menu registrasi *user*. Sedangkan menu yang digunakan untuk proses enkripsi soal ujian komputerisasi menggunakan algoritma WAKE adalah menu kelola soal. Adapun untuk mengakses *web admin* diperlukanya *login admin* yang berhak.

Tampilan menu *login* adalah tampilan yang akan terbuka ketika *admin* pertama kali mengakses *web admin*. Adapun tampilan menu *login admin* sebagai berikut :

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

Gambar 1. Tampilan Login Admin

Tampilan menu kelola soal adalah tampilan pertama yang akan terbuka ketika admin yang memiliki otoritas *login* dengan *username* yang *valid*. Tampilan menu kelola soal adalah tempat dimana soal diinputkan dan dienkripsi menggunakan algoritma WAKE. Adapun tampilan menu kelola soal dapat dilihat pada gambar di bawah ini :

Gambar 2. Tampilan Menu Kelola Soal

Adapun tampilan *form* tambah soal serta proses enkripsi dapat dilihat pada gambar di bawah ini :

Gambar 3. Tampilan Menu Tambah Soal

Tampilan menu *add admin* adalah tampilan menu yang digunakan untuk menambahkan *admin web* yang memiliki wewenang dalam mengelola *web*. Adapun tampilan menu *add admin* dapat dilihat pada gambar di bawah ini :

Gambar 4. Tampilan Menu *Add Admin*

Adapun tampilan menu daftar *user* adalah menu yang menampilkan informasi dari *user* yang berhak melakukan ujian.

Gambar 5. Tampilan Menu Daftar *User*

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

Tampilan ini akan memberikan informasi dari *user* yang sudah selesai melakukan ujian komputerisasi.

Gambar 6. Tampilan Menu Hasil Ujian

Tampilan ini adalah menu yang digunakan oleh *admin* untuk menambah *user* ujian.

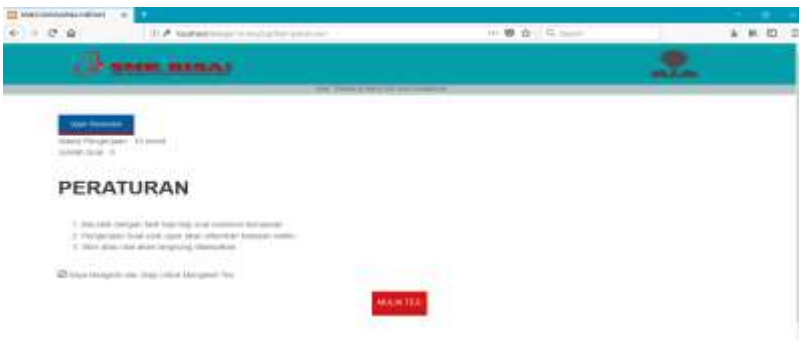
Gambar 7. Tampilan Menu Registrasi *User*

Tampilan *web client* adalah tampilan *web* yang digunakan user untuk memulai ujian. Pada *web client* adalah proses melakukan dekripsi soal menggunakan algoritma WAKE yang sudah dienkripsi sebelumnya oleh *admin*.

Tampilan menu *login* adalah tampilan pertama kali yang akan terbuka ketika *web* diakses oleh *user* yang hendak ujian. *User* memasukan *username* dan *password* yang sudah terdaftar di dalam *database* untuk memulai ujian. Adapun tampilan menu *login web client* adalah sebagai berikut :

Gambar 8. Tampilan Menu *Login Client*

Pada tampilan peraturan ujian, terdapat aturan yang harus disetujui oleh *user* yang hendak ujian. Proses dekripsi soal ujian menggunakan algoritma WAKE adalah ketika *user* memilih tombol Mulai Tes pada tampilan menu peraturan ujian. Adapun tampilan peraturan ujian dapat dilihat pada gambar di bawah ini :

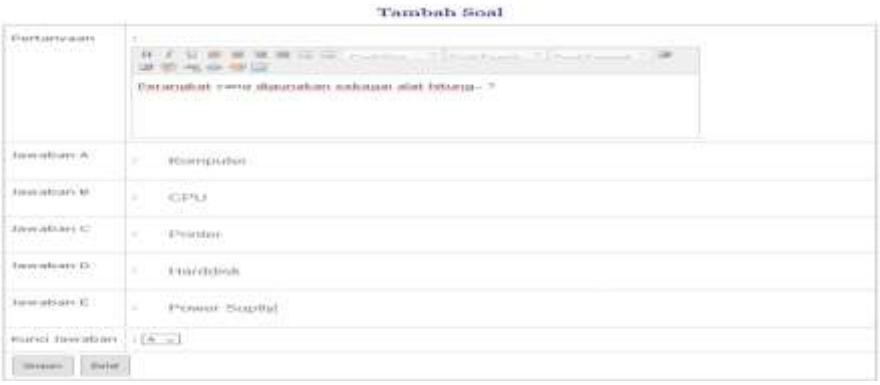


Gambar 9. Tampilan Menu Peraturan Ujian

Tampilan *output* terdiri dari tampilan proses enkripsi soal ujian menggunakan algoritma WAKE di *web admin* dan tampilan proses dekripsi soal ujian menggunakan algoritma WAKE di *web client*. Proses enkripsi soal ujian dilakukan pada menu *web admin* kelola soal. *Admin* yang akan menambahkan soal ujian terlebih dahulu menginput soal kedalam *form* yang sudah didesain sebelumnya. Enkripsi menggunakan algoritma WAKE berjalan

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)

ketika *user* memilih menyimpan soal dengan *button* "SIMPAN". Soal yang tersimpan akan langsung terenkripsi didalam *database* web ujian. Adapun proses enkripsi soal dapat dilihat pada gambar di bawah ini :



Gambar 10. Proses Penginputan Soal

Gambar 11. Soal Terenkripsi

Proses dekripsi soal dilakukan pada menu *web client*. Dekripsi dimulai ketika *user* berada pada menu peraturan ujian dan memilih menyetujui peraturan serta memulai ujian dengan menekan *button* "MULAI TES".

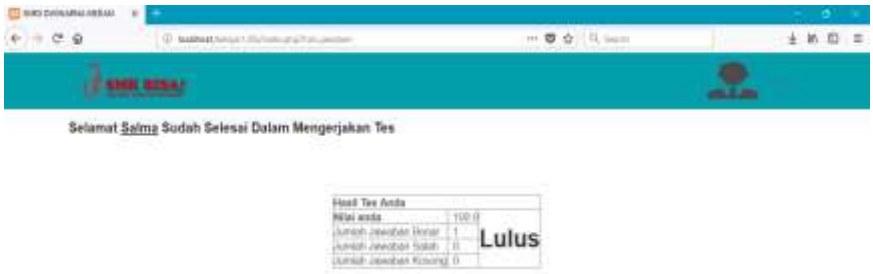
Gambar 12. Proses Dekripsi

Adapun ketika *user* memilih *button* MULAI TES, tampilan akan menuju pada menu soal ujian yang sudah didekripsi menggunakan algoritma WAKE seperti gambar di bawah ini :

Gambar 13. Soal Terdekripsi

Berdasarkan pada gambar 13. di atas, proses dekripsi soal berhasil dilakukan menggunakan algoritma WAKE. Soal yang didekripsi akan kembali menjadi soal yang dapat dibaca dan dipahami maknanya oleh *user* yang sedang ujian. *User* dapat menjawab soal dan memilih tombol "SELESAI" untuk mengakhiri ujian. Sistem akan langsung menampilkan hasil ujian seperti gambar di bawah ini :

Journal of Computer System and Informatics (JoSYC)
Volume 2, No. 1, November 2020, Page 91-106
ISSN 2714-8912 (media online)
ISSN 2714-7150 (media cetak)



Gambar 14. Hasil Ujian *User*

4. KESIMPULAN

Berdasarkan hasil dari implementasi sistem yang telah dilakukan dapat diambil kesimpulan bahwa soal yang terenkripsi menggunakan algoritma WAKE berbasis *web ujian* mengalami perubahan yang siginifikan. Soal tidak dapat dibaca dan dipahami maknanya sehingga meminilisi kebocoran soal. Waktu yang dibutuhkan untuk proses enkripsi dan dekripsi soal bervariasi. Hal ini dipengaruhi oleh panjangnya karakter soal yang diinputkan.

REFERENCES

[1] D. Arius, *Pengantar Ilmu Kriptografi dan Implemantasi*. ANDI Yogyakarta, 2008.
[2] H. Gultom, "Penyandian Email Menggunakan Algoritma Kriptografi WAKE(Word Auto Key Encryption)," *Pelita Inform. Budi Darma*, vol. 4, no. 1, pp. 107–111, 2013.
[3] Ariyus and Dony, *Kriptografi Keamanan Data dan Komunikasi*. Yogyakarta: Grahailmu, 2006.
[4] E. R. Agustina, A. Kurniati, L. S. Negara, P. Minggu, and J. Selatan, "Pemanfaatan Kriptografi Dalam Mewujudkan," *Semin. Nas. Inform. 2009 (semnasIF 2009)*, vol. 2009, no. semnasIF, pp. 22–28, 2009.
[5] F. A. Sinaga and Mesran, "IMPLEMENTASI ALGORITMA ROT13 DAN ALGORITMA CAESAR CHIPER DALAM PENYANDIAN TEKS," *Pelita Inform. Inf. dan Inform.*, vol. 16, no. 1, Jan. 2017.