Seminar Nasional Teknologi Komputer & Sains (SAINTEKS)
SAINTEKS 2020
ISBN: 978-602-52720-7-3
Februari 2020
Hal 429-437

Enkripsi File Teks Menerapkan Algoritma Skipjack

Bryando H, Nurul Hapifa, Rodiana

Program Studi Teknik Informatika, STMIK Budi Darma, Medan Indonesia
Email: [1]bryanexodia@gmail.com, [1] nuruhapifahpurba@gmail.com, [1]anarodiana4@gmail.com

Abstrak−Salah satu hal yang terpenting untuk menjamin kerahasiaan data atau informasi dengan mnelakukan proses enkripsi. Kerahasiaan data atau informasi merupakan hal yang sangat penting dan harus dijaga dengan saat aman. Dengan bertambah banyak nya arus lintas pengiriman data maka pengiriman data sangat rentan dari gangguan orang-orang iseng yamg ingin menegetahui secara paksa isi informasi dari orang lain. Gangguan tersebut berupa penyadapan atau manipulasi data atau text demi keuntungan mereka. Masalah ini dapat diatasi dengan melalukan proses enkripsi pada teks atau dokumen yang akan di kirim sehinnga bisa di baca oleh orang yang tidak punya kepentingan, penulis memakai enkripsi data dengan metode skipjack. Metode ini merupakan cara proses enkripsi pada file text pada suatu aplikasi internet dengan contoh email service dan mempunyai 32 tahap dan disertakan rumus permutasi yang merupakan bagian dari metode skipjack, dengan memakai metode skipjack ini data atau teks yang akan dikirim melalui internet sudah dienkripsi oleh pengirim, kemudian sampai kepada penerima yang didekripsikan kembali menjadi data semula atau plaintext oleh penerima.

Kata Kunci: Kriptografi,Enkripsi,Skipjack,Teks,Informasi

1. PENDAHULUAN

Pada saat ini informasi telah menjelma menjadi hal yang sangat penting dalam aktivitas manusia, tentu saja hal ini sangat memerlukan keamanan informasi yang sangat ketat agar terhindar dari penyadapan yang tidak disukai[1].

Dalam menjaga keamanan informasi,tentunya diperlukan sistem yang menggontrol agar informasi itu terjamin kerahasiaanya.Informasi tersebut harus tetap dirahasiakan selama pengiriman dan harus tetap dirahasiakan sampai ke tempat tujuan. Maka sangat dibutuhkan keamanan agar informasi yang dikirimkan tidak bisa dibuka oleh orang yang tidak bertanggungjawab.Untuk memenuhi hal itu,maka dilakukanlah suatu proses penyandian(ekripsi) terhadap informasi yang akan dikirimkan kepada penerima sehingga hanya penerima yang memiliki kunci rahasia yang bisa membuka enkripsi tersebut[2].

Salah satu algoritma kriptografi yang akan dipakai penulis ialah Algoritma Skipjack. Skipjack adalah suatu algoritma yang telah dikembangkan pada tahun 1987. Skipjack yang disebut juga sebagai representasi dari *family of encryption algorithms* yang dikembangkan oleh Badan Keamanan Nasional Amerika pada tahun 1980[3]. Skipjack memiliki banyak sekali kegunaan dan digunakan oleh banyak lembaga dan insitusi. Dari beberapa hal tersebut Skipjack juga mempunyain kelemahan yaitu: sering terjadinya informasi yang tidak dienkripsi dan dapat dengan mudah di bobol kerhasiaan dari informasi tersebut[1]. Walaupun terdapat kelemahan, Skipjack menyajikan keamanan yang super kuat dan dapat bertahan selama bertahun-tahun sebelum algoritma skipjack ini di rusak oleh *blueforce atack*[3].

Hal yang membuat penulis berkeinginan membuat artikel tentang Algoritma Skipjack ialah karna sampai saat ini masih banyak lembaga atau instansi khusunya di indonesia yang belum mengerti tentang Algortima Skipjack ini.

2. METODOLOGI PENELITIAN

2.1 Kriptografi

Kriptografi (*Cryptogrpahy*) adalah suatu bidang yang mempelajari cara menjaga agar informasi atau data tetap terjamin kerahasiaanya saat dikirim oleh pengirim ke penerima tanpa mengalami gangguan dari orang yang tidak bertanggung jawab[4]. Dalam perkembanganya kriptografi sering dipakai untuk identifikasi pengirim data dengan menggunakan tanda tangan digital atau sidik jari digital*(Fingerprint).* Salah satu tugas dari kriptografi adalah menjaga agar *plaintext* tetap terjamin kerahasiaanya dari orang yang tidak bertanggung jawab atau penyadap*(cryptanalysis*[2]*).*

Plaintext adalah suatu informasi yang bisa dibaca dan dipahami maknanya.*Plaintext* bisa juga berupa data yang dikirim melalui cara yang manual(melalui kantor pos), atau bisa juga melalui cara yang sudah terkomputerisasi(E.mail). pesan/informasi yang tersimpan dapat juga berbentuk teks,audio dan video dan lain-lain[1]. *Chippertext* adalah suatu pesan/informasi tidak bisa dipahami maknanya oleh orang yang tidak bertanggung jawab, maka pesan/informasi perlu dirubah/diacak ke bentuk yang lebih aman dan tidak dapat dipahami oleh orang asing. *Cipherteks* harus dapat dirubah menjadi *plainteks* agar informasi yang diterima oleh si penerima bisa terbaca[1].

Enkripsi*(Encryption)* adalah sebuah teknik yang bisa menjadikan pesan yang bisa dibaca(*Plaintext)* menjadi pesan yang acak dan tidak akan bisa dibaca jika tidak memiliki kunci untuk membuka data atau informasi tersebut[2]. sedangkan Deskripsi*(Decryption)* merupakan kebalikan dari enkripsi yang dimana proses deskripsi bertujuan mengubah bentuk *ciphertext* ke bentuk *plaintext* dengan mengunakan pembalik atau key yang sama. Enkripsi dan Deskripsi harus bisa diterapkan pada pesan/informasi yang akan dikirimkan atau pesan yang akan disimpan[1].

2.1. Algoritma Kriptografi Skipjack

Alogoritma skipjack adalah salah satu algoritma yang dimana Algoritma skipjack ini memakai kunci yang sama untuk melakukan proses enkripsi dan deskripsinya[2][5]. Algoritma skipjack dikembangkan pada tahun 1987 dan dipublikasikan pada tahun 1998 oleh badan keamanan amerika serikat[5][6]. Algoritma skipjack memiliki masukan*(Plaintext)* yang

ISBN: 978-602-52720-7-3
Februari 2020
Hal 429-437

berukuran 64 bit yang kemudian data tersebut dirubah ke bentuk kumpulan blok-blok data berukuran 64 bit yang selanjutnya diproses dengan menggunakan kunci yang sama untuk menghasilkan keluaran*(Chipertext)*(Supriatno,2007)[2]. Algoritma skipjack merupakan *iterated block chiper* yang dalam proses enkripsi maupun deskripsinya memiliki 32 putaran artinya algoritma skipjack ini diputar 32 kali untuk menghasilkan data keluaran*(Chipertext).* Kombinasi tersebutlah yang membuat algoritma skipjack ini mempunyai tingkat keamanan yang sangat tinggi.operasi matematik yang digunakan dalam algoritma skipjack ini adalah XOR dan permutasi[2][5].

3. HASIL DAN PEMBAHASAN

Permasalahan-permasalahan yang dapat ditemukan untuk perancangan proses enkripsi file teks adalah:
1. Langkah-langkah melakukan proses enkripsi terhadap file teks
2. Algortima yang dibutuhkan harus di sesuaikan untuk kebutuhan dalam proses enkripsi file teks

Pada metode skipjack, plaintext yang hendak dilakukan proses enkripsi harus dikonversikan terlebih dahulu ke bilangan hexadesimal. Bilangan hexadesimal merupakan nilai ASCII (*American Standart Code for Information Interchange)* Dari masing-masing karakter dari palintext tersebut.

Pada algoritma skipjack menggunakan kunci simetri. Proses pemyusunan kunci algoritma skipjack adalah proses yang sangat penting untuk dilakukan sebelum melakukan penengkripsian. Berikut adalah proses pelaksanaan pengoloahan kunci untuk menghasilkan 10 subkunci*(cryptovariable).*
Plaintext: Wisudaku
Kunci: Kriptologi
Proses pengubahan bentuk kunci ke dalam bentuk hexadesimal
Hexadesimal: 4B,5249,50,54,4F,4C,4F,47,49
Bagi kunci (*key)* menjadi 10 subkunci *(cryptovariable)* masing-masing memiliki ukuran 8bit sebagai berikut :
Cv[0] = 4B
Cv[1] = 52
Cv[2] = 49
Cv[3] = 50
Cv[4] = 54
Cv[5] = 4F
Cv[6] = 4C
Cv[7] = 4F
Cv[8] = 47
Cv[9] = 49

3.1 Proses Enkripsi File Teks

Proses enkripsi file teks menggunakan algoritma skipjack dengan memiliki 32 putaran dengan memakai 10 buah sub kunci yang merupakan hasil pembagian dari kunci rahasia. Berikut adalah contoh dari pelaksanaan proses perhitungan manual dari Proses enkripsi file teks mengunakan algoritma skipjack:
Plaintext: WISuDAKu
Kunci: Kriptografi

Pengolahan Kunci
Ubah bentuk kunci ke dalam bentuk hexadesimal
Hexadesimal: 4B,5249,50,54,4F,4C,4F,47,49
Bagi kunci (*key)* menjadi 10 subkunci *(cryptovariable)* masing-masing memiliki ukuran 8bit sebagai berikut :
Cv[0] = 4B
Cv[1] = 52
Cv[2] = 49
Cv[3] = 50
Cv[4] = 54
Cv[5] = 4F
Cv[6] = 4C
Cv[7] = 4F
Cv[8] = 47
Cv[9] = 49

Proses Enkripsi
Ubah bentuk *Plaintext* ke dalam bentuk hexadesimal
Hexadesimal: 57,49,53,55,44,41,4B,55
Selanjutnya bagi *Plaintext* menjadi empat bagian (w1, w2, w3, w4) sebagai berikut:
W1(0) = 5749
W2(0) = 5355
W3(0) = 4441
W4(0) = 4B55

ISBN: 978-602-52720-7-3
Februari 2020
Hal 429-437

Putaran ke-1 (Rule A, K = 0, Counter = 1)
G(W1(0)) = G(5749)
g1 = 57
g2 = 49
g3 = F(g2 ⊕ cv[(4*k) mod 10]) ⊕ g1
cv[(4*0)mod 10] = cv[0] = 4B
g2 = 49: 01001001
cv[0] = 4B : 01001001 ⊕
0000 0010
F(0000 0010) = F(02) = 09
F(02) = 09 = 0000 1001
g1 = 57 = 01010111 ⊕
= 01011110 (5E)
g3 = 01011110(5E)
g4 = F(g3 ⊕ cv[(4*k)+1 mod 10]) ⊕ g2
cv[(4*0)+1mod 10] = cv[1] = 52
g3 = 5E = 01011110
cv[1] = 52 = 01010010 ⊕
0000 1100
F(0000 1100) = F(0C) = 99
F(0C) = 99 = 10011001
G2 = 49 = 01001001 ⊕
= 11010000(D0)
G4 = 11010000(D0)
g5 = F(g4 ⊕ cv[(4*k)+2 mod 10]) ⊕ g3
cv[(4*0)+2mod 10] = cv[2] = 49
g4 = D0 = 11010000
cv[2] = 49 = 01001001 ⊕
10011001
F(11011101) = F(99) = 74
F(99) = 74 = 01110100
G3 = 5E = 01011110 ⊕
= 00101010(2A)
G5 = 00101010(2A)
G6 = F(g5 ⊕ cv[(4*k)+3 mod 10]) ⊕ g4
cv[(4*0)+3mod 10] = cv[3] = 50
g5 = 2A = 00101010
cv[3] = 50 = 01001001 ⊕
0111010
F(01111010) = F(7A) = D6
F(7A) = D6 = 01110100
G4 = D0 = 11010000 ⊕
= 11010000(D0)
G6 = 00000110(06)
G5+G6 = 2A06
W1(1) = g(w1(0) ⊕ w4(0) ⊕ counter
G(w1(0) = 2A06 = 0010101000000110
W4(0) = 4B55 = 0100100001010101 ⊕
0110000101010011
Counter = 1 0000000000000001 ⊕
= 0110000101010010
W1(1) = [6152]
W2(1) = (w1(0) = [2A06]
W3(1) = w2(0) = [5355]
W4(1) = w3(0) = [4441]

Putaran ke-2 (Rule A, K = 1, Counter = 2)
Ciphertext = W1(1) + W2(1) + W3(1) + W4(1)
= 6152 2A06 5355 4441
G(W1(1)) = G(6152)
g1 = 61
g2 = 52
g3 = F(g2 ⊕ cv[(4*k) mod 10]) ⊕ g1

ISBN: 978-602-52720-7-3
Februari 2020
Hal 429-437

cv[(4*1)mod 10] = cv[4] = 54
g2 = 52: 01010010
cv[4] = 54 : 01010100 ⊕
00000110
F(0000 0110 = F(06) = F6
F(06) = F6 = 11110110
g1 = 61 = 01100001 ⊕
= 10010111 (97)
g3 = 10010111(97)
g4 = F(g3 ⊕ cv[(4*k)+1 mod 10]) ⊕ g2
cv[(4*1)+1mod 10] = cv[5] = 4F
g3 = 97 = 10010111
cv[5] = 4F = 01001111 ⊕
11011000
F(11011000) = F(D8) = EC
F(D8) = EC = 11101100
G2 = 52 = 01010010 ⊕
= 11010000(D0)
G4 = 10111110(BE)
g5 = F(g4 ⊕ cv[(4*k)+2 mod 10]) ⊕ g3
cv[(4*1)+2mod 10] = cv[6] = 4c
g4 = BE = 10111110
cv[6] = 4C = 01001100 ⊕
11110010
F(11110010) = F(F2) = A9
F(F2) = A9 = 10101001
G3 = 97 = 00111110 ⊕
= 00111110(3E)
G5 = 00111110(3E)
G6 = F(g5 ⊕ cv[(4*k)+3 mod 10]) ⊕ g4
cv[(4*1)+3mod 10] = cv[7] = 4F
g5 = 3E = 00111110
cv[7] = 4F = 01001111 ⊕
01110001
F(01110001) = F(71) = FC
F(71) = FC = 11111100
G4 = BE = 10111110 ⊕
= 01000010(42)
G6 = 01000010(42)
G5+G6 = 3E42
W1(2) = g(w1(1) ⊕ w4(1) ⊕ counter
G(w1(1) = 3E42 = 0011111001000010
W4(0) = 4441 = 0100010001000001 ⊕
011110100000011
Counter = 2 0000000000000010 ⊕
= 0111101000000001
W1(2) = [7A01]
W2(2) = (w1(1) = [3E42]
W3(2) = w2(1) = [2A06]
W4(2) = w3(1) = [5355]

Putaran ke-3 (Rule A, K = 2, Counter = 3)
Ciphertext = W1(2) + W2(2) + W3(2) + W4(2)
= 7A01 3E42 2A06 5355
G(W1(2)) = G(7A01)
g1 = 7A
g2 = 01
g3 = F(g2 ⊕ cv[(4*k) mod 10]) ⊕ g1
cv[(4*2)mod 10] = cv[8] = 47
g2 = 01 : 00000001
cv[8] = 47 : 01000111 ⊕
01000110
F(01000110) = F(46) = 81

ISBN: 978-602-52720-7-3
Februari 2020
Hal 429-437

F(46) = 81 = 10000001
g1 = 7A = 01111010 □
= 11111011 (FB)
g3 = 11111011(FB)
g4 = F(g3 ⊕ cv[(4*k)+1 mod 10]) ⊕ g2
cv[(4*2)+1mod 10] = cv[9] = 49
g3 = FB = 11111011
cv[9] = 49 = 01001001 ⊕
10110001
F(10110001) = F(B2) = 1C
F(B2) = 1C = 00011100
G2 = 01 = 00000001 ⊕
= 11010000(D0)
G4 = 00011101(1D)
g5 = F(g4 ⊕ cv[(4*k)+2 mod 10]) ⊕ g3
cv[(4*2)+2mod 10] = cv[0] = 4B
g4 = 1D = 00011101
cv[0] = 4B = 11011011 ⊕
01010110
F(01010110) = F(56) = BF
F(56) = BF = 10111111
G3 = FB = 11111011 ⊕
= 00111110(3E)
G5 = 01000100(44)
G6 = F(g5 ⊕ cv[(4*k)+3 mod 10]) ⊕ g4
cv[(4*2)+3mod 10] = cv[1] = 52
g5 = 44 = 01000100
cv[1] = 52 = 01010010 ⊕
00010110
F(00010110) = F(16) = CA
F(16) = CA = 11001010
G4 = 1D = 00011101 ⊕
= 11010111(D7)
G6 = 11010111(D7)
G5+G6 = 44D7
W1(3) = g(w1(2) ⊕ w4(2) ⊕ counter
G(w1(2) = 44D7 = 0100010011010111
W4(2) = 5355 = 0101001101010101 ⊕
0001011110000010(1782)
Counter = 3 0000000000000011 ⊕
= 0001011110000001
W1(3) = [1781]
W2(3) = (w1(2) = [44D7]
W3(3) = w2(2) = [3E42]
W4(3) = w3(2) = [2A06]

Putaran ke-4 (Rule A, K = 3, Counter = 4)
Ciphertext = W1(2) + W2(2) + W3(2) + W4(2)
= 1781 44D7 3E42 2A06
G(W1(3)) = G(1781)
g1 = 17
g2 = 81
g3 = F(g2 ⊕ cv[(4*k) mod 10]) ⊕ g1
cv[(4*2)mod 10] = cv[2] = 49
g2 = 81: 10000001
cv[8] = 49 : 01001001 ⊕
11001000
F(11001000) = F(C8) = 29
F(C8) = 29 = 00101001
g1 = 17 = 00010111 ⊕
= 00111110 (3E)
g3 = 00111110(3E)
g4 = F(g3 ⊕ cv[(4*k)+1 mod 10]) ⊕ g2

ISBN: 978-602-52720-7-3
Februari 2020
Hal 429-437

cv[(4*3)+1mod 10] = cv[3] = 50
g3 = 3E = 00111110
cv[3] = 50 =01010000 ⊕
01101110
F(01101110) = F(6E) = 9B
F(6E) = 9B = 10011011
G2 = 01 = 10000001 ⊕
= 00011010(1A)
G4 = 00011101(1D)
g5 = F(g4 ⊕ cv[(4*k)+2 mod 10]) ⊕ g3
cv[(4*3)+2mod 10] = cv[4] = 54
g4 = 1A = 00011010
cv[0] = 54 = 01010100 ⊕
01001110
F(01001110) = F(4E) = 2F
F(4E) = 2F = 00101111
G3 = 3E = 00111110 ⊕
= 00010001(11)
G5 = 01000100(11)
G6 = F(g5 ⊕ cv[(4*k)+3 mod 10]) ⊕ g4
cv[(4*3)+3mod 10] = cv[5] = 4F
g5 = 11 = 00010001
cv[5] = 4F = 01001111 ⊕
01011110
F(01011110) = F(5E) = D8
F(5E) = D8 = 11011000
G4 = 1A = 00011010 ⊕
= 11000010(C2)
G6 = 11000010 (C2)
G5+G6 = 1AC2
W1(4) = g(w1(3) ⊕ w4(3) ⊕ counter
G(w1(3) = 1AC2 = 0001101011000010
W4(3) = 2A06 = 0010101000000110 ⊕
0011000011000100(30C4)
Counter = 4 0000000000000100 ⊕
= 0011000011000000
W1(4) = [30C0]
W2(4) = (w1(3) = [1AC2]
W3(4) = w2(3) = [44D7]
W4(4) = w3(3) = [3E42]

Hasil akhir proses enkripsi pada putaran ke 32 sebagai berikut : 17C2 B916 0D0A 5BC3

3.2 Proses Dekripsi File Teks

Proses dekripsi merupakan kebalikan dari proses enkripsi, yang bertujuan untuk mengembalikan *ciphertext* kebentuk *plaintext* (karakter awal). Berikut adalah proses dekripsi:

Ubah *ciphertext* kedalam bentuk *hexadesimal*: **17c2 B916 0d0a 5Bc3** kemudian bagi *ciphertext* menjadi 4 bagian (W1, W2, W3, W4) yaitu sebagai berikut:
W1(32) = 17c2 W3(32) = 0d0a
W2(32) = b916 W4(32) =5bc3

Putaran ke-1 (Rule B^{-1}, K = 32, Counter = 32)
G^{-1}(W2(32)) = G(B916)
G5 = B9
G6 = 16
G4 = F(g5 ⊕ cv[(4*(k-1)+3 mod 10]) ⊕ g6
cv[(4*(32-1)+3mod 10] = cv[7] = 4F
g5 = B9: 10111001
cv[7] = 4F : 01001111 ⊕
11110110
F(11110110) = F(F6) = B5
F(F6) = B5 = 10110101
G6 = 16 = 00010110 ⊕
= 10100011(A3)

ISBN: 978-602-52720-7-3
Februari 2020
Hal 429-437

G4 = 10100011(A3)
G3 = F(g4 ⊕ cv[(4*(k-1)+2 mod 10]) ⊕ g5
cv[(4*(32-1)+2mod 10] = cv[6] = 4C
g4 = A3: 10100011
cv[6] = 4C : 01001100 ⊕
11101111
F(11101111) = F(EF) = AC
F(EF) = AC = 10101100
G5 = B9 = 10111001 ⊕
= 00010101(15)
G3 = 00010101(15)
G2 = F(g3 ⊕ cv[(4*(k-1)+1 mod 10]) ⊕ g4
cv[(4*(32-1)+1mod 10] = cv[5] = 4F
g1 = 15: 00010101
cv[5] = 4F : 01001111 ⊕
01011010
F(01011010) = F(5A) = 80
F(5A) = 80 = 10000000
G4 = A3 = 10100011 ⊕
= 00100011(23)
G2 = 00100011(23)
G1 = F(g2 ⊕ cv[(4*(k-1) mod 10]) ⊕ g3
cv[(4*(32-1)mod 10] = cv[4] = 54
g2 = 23: 00100011
cv[4] = 54 : 01010100 ⊕
01110111
F(01110111) = F(77) = 20
F(77) = 20 = 01110111
G3 = 15 = 00010101 ⊕
= 00110101(35)
G1 = 00110101(35)
W1(31) = G^{-1}((w2(32)) = 3523
W2(31) = (w2(32) ⊕ w3(32) ⊕ counter
G^{-1}(w2(32) = 3523 = 0011010100100011
W3(32) = 0D0A = 0000110100001010 ⊕
0011100000101001(3829)
Counter = 32 0000000000110010 ⊕
= 0011100000011011(381B)
W3(31) = w4(32) = [5BC3]
W4(31) = w1(32) = [17C2]

Putaran ke-2 (Rule B^{-1}, K = 31, Counter = 31)
Plaintext: W1(31) + W2(31) + W3(31) + W4(31) = 3523,381B,5B73,17C2
G^{-1}(W2(31)) = G(281B)
G5 = 38
G6 = 1B
G4 = F(g5 ⊕ cv[(4*(k-1)+3 mod 10]) ⊕ g6
cv[(4*(31-1)+3mod 10] = cv[3] = 50
g5 = 38: 00111000
cv[3] = 50 : 01010000 ⊕
01101000
F(01101000) = F(68) = 45
F(68) = 45 = 01000101
G6 = 1B = 00011011 ⊕
= 01011110(5E)
G4 = 01011110(5E)
G3 = F(g4 ⊕ cv[(4*(k-1)+2 mod 10]) ⊕ g5
cv[(4*(31-1)+2mod 10] = cv[2] = 49
g4 = 5E: 01011110
cv[2] = 49 : 01001001 ⊕
00010111
F(00010111) = F(17) = 2E
F(17) = 2E = 00101110

ISBN: 978-602-52720-7-3
Februari 2020
Hal 429-437

G5 = 38 = 00111000 ⊕
= 00010110(16)
G3 = 00010110 (16)
G2 = F(g3 ⊕ cv[(4*(k-1)+1 mod 10]) ⊕ g4
cv[(4*(31-1)+1mod 10] = cv[1] = 52
g3 = 16: 00010110
cv[1] = 52 : 01010010 ⊕
01000100
F(01000100) = F(44) = C1
F(44) = C1 = 11000001
G4 = 5E = 01011110 ⊕
= 10011111(9F)
G2 = 10011111(9F)
G1 = F(g2 ⊕ cv[(4*(k-1) mod 10]) ⊕ g3
cv[(4*(31-1)mod 10] = cv[0] = 4B
g2 = 9F: 10011111
cv[0] = 4B : 01001011 ⊕
11010100
F(11010100) = F(D4) = 75
F(D4) = 75 = 01110101
G3 = 16 = 00010110 ⊕
= 01100011(63)
G1 = 01100011(63)
W1(30) = G^{-1}((w2(32)) = 759F
W2(30) = (w2(31) ⊕ w3(31) ⊕ counter
G^{-1}(w2(31) = 759F = 011101011001111
W3(31) = 5B73 = 0101101101110011 ⊕
0010111011101100(2EEC)
Counter = 31 0000000000110001 ⊕
= 0010111011011101(2EDD)
W3(30) = w4(31) = [17C2]
W4(30) = w1(31) = [3523]

Putaran ke-3 (Rule B^{-1}, K = 30, Counter = 30)
Plaintext: W1(30) + W2(30) + W3(30) + W4(30) = 759F,2EDD,17C2,3523
G^{-1}(W2(30)) = G(2EDD)
G5 = 2E
G6 = DD
G4 = F(g5 ⊕ cv[(4*(k-1)+3 mod 10]) ⊕ g6
cv[(4*(30-1)+3mod 10] = cv[9] = 49
g5 = 2E: 01100111
cv[9] = 49 : 01001001 ⊕
01100111
F(01100111) = F(67) = 69
F(67) = 69 = 01101001
G6 = DD = 11011101 ⊕
= 10110100(B4)
G4 = 10110100(B4)
G3 = F(g4 ⊕ cv[(4*(k-1)+2 mod 10]) ⊕ g5
cv[(4*(30-1)+2mod 10] = cv[8] = 47
g4 = B4: 10110100
cv[8] = 47 : 01000111 ⊕
11110011
F(11110011) = F(F3) = 13
F(F3) = 13 = 00010011
G5 = 2E = 00101110 ⊕
= 00111101(3D)
G3 = 00111101(3D)
G2 = F(g3 ⊕ cv[(4*(k-1)+1 mod 10]) ⊕ g4
cv[(4*(30-1)+1mod 10] = cv[7] = 4F
g3 = 3D: 00111101
cv[7] = 4F : 01001111 ⊕
01110010

ISBN: 978-602-52720-7-3
Februari 2020
Hal 429-437

F(01110010) = F(72) = B2
F(72) = B2 = 10110010
G4 = B4 = 10110111 ⊕
= 00000110(06)
G2 = 00000110(06)
G1 = F(g2 ⊕ cv[(4*(k-1) mod 10]) ⊕ g3
cv[(4*(30-1)mod 10] = cv[6] = 4C
g2 = 06: 00000110
cv[6] = 4C : 01001100 ⊕
01001010
F(01001010) = F(4A) = 1A
F(4A) = 1A = 00011010
G3 = 3D = 00111101 ⊕
= 00100111(27)
G1 = 00100111(27)
$W1(29) = G^{-1}((w2(32)) = 2706$
W2(29) = (w2(30) ⊕ w3(30) ⊕ counter
$G^{-1}(w2(30) = 2706 = 0010011100000110$
W3(30) = 17C2 = 0001011111000010 ⊕
0011000011000100(30C4)
Counter = 30 0000000000101001 ⊕
= 0011000011101101(30ED)
W3(29) = w4(30) = [3523]
W4(29) = w1(30) = [759F]

Lakukan perhitungan dengan cara yang sama sampai putaran ke 32 sehingga akan mendapatkan hasil akhir sebagai berikut: 5749,5355, 4441,4B55

4. KESIMPULAN

Dengan menggunakan sistem keamanan data file teks yang akan di enkripsi akan sangat sulit dibuka/dipecahkan apabila tidak mempunyai kunci untuk melakukan deskrpisi pada file teks tersebut

Dengan mengunakan algoritma skipjack data yang akan di enkripsi akan sangat aman karena algoritma ini menggunakan 64 bit blok data dan menggunakan 80 bit kunci serta memakai 32 putaran sehingga membuat algoritma skipjack ini sangat susah untuk dibuka atau dipecahkan

Dari hasil pengujian pada waktu di proses maka dapat disimpulkan bahwa proses enkripsi pada algoritma skipjack dengan menggunakan kunci yang berbeda antara *Sender* dan *Receiver* akan menghasilkan *plaintext* yang berbeda namun *ciphertext* yang dihasilkan akan tetap sama

Dengan demikian tidak adanya resiko yang memumgkinkan bahwa algoritma skipjack ini dapat dipecahkan dalam kurun waktu 30-40 tahun yang akan datang . sehingga algoritma skipjack ini masih sangat aman untuk diterapkan.

REFERENCES

[1] Y. Asri, "PERANCANGAN APLIKASI ENKRIPSI DAN DEKRIPSI DATA DENGAN MENGGUNAKAN ALGORITMA SKIPJACK.pdf," vol. 2. p. 2, 2009.
[2] S. M. Indonesia, "ISSN 2442-4943 SISTEM PENGKODEAN DATA PADA FILE TEKS UNTUK KEAMANAN INFORMASI DENGAN MENGGUNAKAN METODE SKIPJACK INFORMASI DENGAN MENGGUNAKAN METODE SKIPJACK Abstrak," vol. 12, no. 1, pp. 59–72, 2018.
[3] M. A. H. Nim, "Studi Mengenai Algoritma Skipjack dan Penerapannya."
[4] P. Kriptografi and D. A. N. Pemakaianya, "2016," no. May, 2016.
[5] J. Muhammad, "Analisis dan Implementasi Kombinasi Algoritma SKIPJACK dan Algoritma MCALIECE pada Pengamanan File Teks," 2017.
[6] N. Sinaga, S. Aini, and B. Gulo, "Penerapan Algoritma Skipjack Untuk Menyandikan Short Message Service," *J-SAKTI (Jurnal Sains Komput. dan Inform.*, vol. 2, no. 1, p. 33, 2018, doi: 10.30645/j-sakti.v2i1.54.