Algoritma Cipher Block EZPZ

easy to code hard to break

Muhammad Visat Sutarno (13513037)
Program Studi Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
Jl. Ganesha 10 Bandung 40132, Indonesia
mvisat@gmail.com

Muhammad Aodyra Khaidir (13513063)
Program Studi Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
Jl. Ganesha 10 Bandung 40132, Indonesia
aodyra@gmail.com

Abstrak*—Saat ini informasi bergerak secara cepat dari satu tempat ke tempat lain. Informasi tersebut dikirim dalam data melalui jaringan publik. Orang-orang dapat dengan mudah mengakses dan menggunakan data yang dikirim. Informasi yang dikirim harus dijaga kerahasiannya. Makalah ini mengajukan algoritma kriptografi block cipher untuk menyembunyikan isi informasi. Algoritma ini Ukuran block yang digunakan adalah 128 bit. Jaringan feistel digunakan dalam proses enkripsi dan dekripsi. Panjang kunci yang digunakan adalah 128 bit. Proses enkripsi pada jaringan feistel menggunakan operasi XOR, substitusi, pergeseran sikuler, dan transposisi. Proses ini dilakukan sebanyak 16 kali putaran dalam enkripsi satu blok pesan. Setiap putaran enkripsi menggunakan kunci internal yang dihasilkan dari kunci masukan pengguna.

Keywords—block cipher; enkripsi; dekripsi; substitusi; transposisi, jaringan feistel, operasi XOR.

I. Pendahuluan

Informasi dan komunikasi dibutuhkan oleh semua orang untuk beraktivitas dalam kehidupan sehari-hari. Komunikasi memungkinkan adanya pertukaran informasi dari satu tempat ke tempat lain. Komunikasi yang banyak digunakan saat ini adalah komunikasi yang melalui jaringan publik seperti internet dan jaringan seluler. Komunikasi yang dilakukan sering membawa informasi yang penting. Informasi yang penting menjadi berbahaya apabila orang yang tidak mempunyai hak mengetahui isi informasi. Keamanan informasi menjadi hal penting sehingga isi yang dikandung tidak diketahui oleh sembarang orang.

Kriptografi adalah ilmu yang mempelajari bagaimana menyembunyikan isi sebuah pesan. Penyembunyian isi sebuah pesan ini bertujuan untuk menjaga kerahasiaan isi pesan dari orang-orang yang tidak berhak dalam membaca pesan. Kriptografi banyak digunakan dalam berbagai bidang. Beberapa bidang tersebut adalah pengiriman surat elektronik, pengiriman data, saluran telefon, dst. Kriptografi menggunakan kunci untuk enkripsi dan dekripsi sebuah pesan. Kriptografi dibagi ke dalam dua kelompok berdasarkan kunci yang digunakan untuk enkripsi dan dekripsi. Kelompok tersebut adalah kriptografi kunci simetri dan kunci asimetri.

Block cipher adalah salah satu teknik dalam kriptografi modern. Teknik ini mengelompokkan pesan ke dalam blok-blok dengan ukuran yang sama. Setiap blok pesan dilakukan enkripsi dengan kunci masukan pengguna.

Makalah ini memaparkan algoritma *chiper block* baru dengan nama EZPZ. Algoritma ini dibangung dengan prinsip kemudahan implementasi, tetapi sulit untuk dipecahkan. Algoritma ini menggunakan operasi XOR, substitusi, transposisi, dan pergeseran sikuler. Ukuran blok pesan algoritma ini adalah 128 bit. Algoritma ini menerima kunci masukan pengguna sebesar 128 bit.

II. Dasar Teori

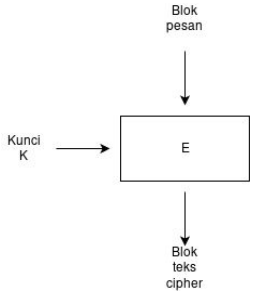
A. Cipher Blok

Algoritma *block cipher* adalah salah satu jenis kriptografi modern kunci simetris. Proses enkripsi dan dekripsi teknik ini menggunakan kunci yang sama. *Block cipher* membagi pesan ke dalam blok-blok dengan ukuran yang ditentukan. Setiap blok dilakukan enkripsi dan dekripsi dengan kunci masukan pengguna. Algoritma ini banyak digunakan dalam proses pengiriman data melalui jaringan internet, karena ukuran blok-blok yang dapat ditentukan.

Block cipher dibagi ke dalam beberapa kelompok. Pengelompokan mode *block cipher* berdasarkan cara operasi blok. Kelompok-kelompok tersebut adalah *Electronic Code Book* (ECB), *Cipher Block Chaining* (CBC), *Cipher Feedback* (CFB), dan *Output Feedback* (OFB).

Electronic Code Book (ECB) adalah algoritma kriptografi *cipher block* yang melakukan enkripsi blok pesan secara independen. Proses enkripsi blok pesan menghasilkan teks *cipher* yang sama setiap kali dienkripsi dengan suatu kunci.

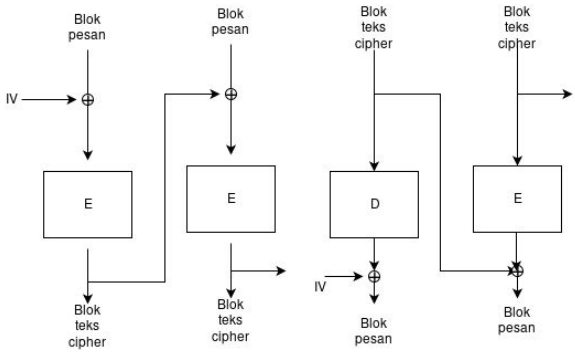
Secara umum proses enkripsi dapat digambarkan dalam skema berikut.



Gambar 1. Skema enkripsi ECB

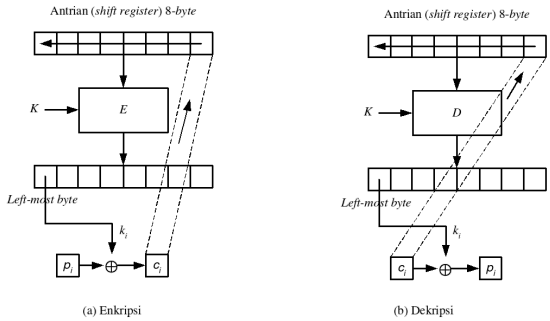
Electronic Code Book memiliki beberapa kelebihan dan kekurangan. proses enkripsi dapat dilakukan secara tidak berurutan. Salah satu aplikasi yang sesuai dengan metode ini adalah enkripsi basis data. Kesalahan beberapa bit pada proses enkripsi blok pesan mempengaruhi blok teks *cipher* yang bersesuaian. Namun, Manipulasi teks *cipher* mudah dilakukan jika lawan mengetahui posisi pesan yang bersesuaian. Jadi kesalahan proses beberapa bit pada proses enkripsi atau dekripsi tetap menghasilkan pesan yang bermakna.

Cipher Block Chaining (CBC) adalah algoritma kriptografi *cipher block* yang memiliki ketergantungan dengan blok pesan sebelumnya. Proses enkripsi blok pesan menggunakan hasil enkripsi proses sebelumnya. Secara umum proses enkripsi dapat digambarkan dalam skema berikut.



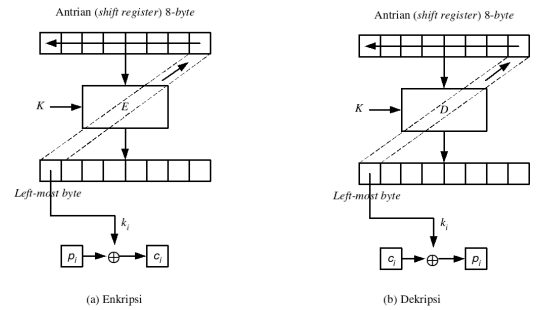
Gambar 2. Skema enkripsi dan dekripsi CBC

Cipher-Feedback (CFB) adalah algoritma kriptografi *cipher block* yang melakukan enkripsi pada unit yang lebih kecil dari ukuran blok. Algoritma ini dapat melakukan enkripsi bit per bit, 2 bit, 3 bit, dst. algoritma ini menggunakan sebuah antrian dengan ukuran sama dengan blok pesan. Salah satu aplikasi yang diterapkan dengan mode CFB adalah komunikasi data. Komunikasi data melakukan proses enkripsi dengan blok yang belum lengkap seperti *stream cipher*.



Gamber 3. Skema enkripsi dan dekripsi CFB [*Algoritma Kripto Moden oleh M, Rinaldi. 2015*]

Output-Feedback (OFB) adalah algoritma kriptografi *cipher block* yang memiliki proses mirip dengan CFB. Beberapa bit hasil enkripsi antrian digunakan kembali menjadi elemen posisi paling kanan di antrian. Kesalahan n-bit pada blok pesan akan mempengaruhi blok teks *cipher* yang bersesuaian saja.



Gambar 4. Skema enkripsi dan dekripsi OFB [*Algoritma Kripto Moden oleh M, Rinaldi. 2015*]

B. Prinsip Konfusi dan Difusi

Prinsip konfusi dalam kriptografi adalah menyembunyikan hubungan antara teks pesan, teks *cipher*, dan kunci masukan. Prinsip bertujuan agar distribusi statistik pesan dalam suatu bahasa sulit dianalisis. Salah satu metode yang digunakan untuk menerapkan prinsip ini adalah penggunaan operasi substitusi pada proses enkripsi.

Prinsip difusi dalam kriptografi adalah menyebarkan pengaruh satu bit pesan atau kunci ke sebanyak mungkin teks *cipher* selanjutnya. Perubahan beberapa bit pada pesan akan menghasilkan perubahan pada teks *cipher* yang tidak dapat diprediksi. Beberapa mode algoritma kriptografi *block cipher* yang menggunakan prinsip ini adalah CBC dan CFB.

C. Jaringan Feistel

Jaringan feistel adalah struktur proses enkripsi dan dekripsi yang digunakan dalam *block cipher*. Struktur proses ini membagi pesan ke dalam dua bagian. Jaringan feistel bersifat *reversible* dalam proses enkripsi dan dekripsi. Jaringan ini memungkinkan kita untuk melakukan proses dekripsi yang serupa dengan proses enkripsi.

Setiap bagian dari pesan dilakukan proses yang berbeda. Bagian kanan pesan menjadi bagian kiri pada tahapan selanjutnya. Bagian kanan tahapan selanjutnya dihasilkan dengan memasukkan bagian kanan pesan tahap sebelumnya ke dalam fungsi enkripsi F dengan suatu kunci. Lalu, hasil enkripsi fungsi F di-XOR kan dengan bagian kiri tahap sebelumnya.

Jaringan feistel biasanya dilakukan bersamaan *cipher* berulang. Proses pertukaran bagian kanan dan kiri dilakukan beberapa kali dengan kunci internal pada fungsi f yang dihasilkan dari satu kunci masukan. Kunci internal yang dihasilkan menggunakan algoritma tertentu sehingga kunci masukan menghasilkan beberapa kunci internal yang berbeda.

III. Rancangan Algoritma

Algoritma EZPZ menggunakan beberapa tahap dalam proses enkripsi blok pesan. algoritma ini menggunakan jaringan feistel sebagai struktur proses enkripsi dan dekripsi. Ukuran blok pesan yang digunakan adalah 128 bit. Blok pesan ini dibagi ke dalam dua bagian yang sama besar. Proses enkripsi pada jaringan feistel dilakukan sebanyak 16 kali. Fungsi enkripsi F pada jaringan feistel menggunakan beberapa operasi seperti operasi XOR, pergeseran sikuler, transposisi bit-bit, dan subsitutsi. Ukuran kunci yang digunakan adalah 128 bit. Kunci masukan akan menghasilkan 16 kunci internal untuk setiap proses putaran di jaringan feistel. Kunci internal digunakan pada tahap operasi XOR fungsi F di jaringan feistel.

A. Jaringan Feistel

Algoritma ini menggunakan jaringan feistel umum dalam proses enkripsi dan dekripsi. Blok pesan sebesar 128 bit dibagi ke dalam dua bagian. Bagian 64 bit pertama menjadi bagian kiri pada jaringan feistel sedangkan bagian 64 bit selanjutnya menjadi bagian kanan pada jaringan feistel. Bagian kanan akan menjadi bagian kiri pada iterasi selanjutnya. Bagian kiri dilakukan operasi xor dengan hasil fungsi F bagian kanan. Iterasi ini dilakukan sebanyak enam belas kali. Gambar jaringan feistel yang digunakan sebagai berikut.

B. Pembangkitan kunci internal

Kunci internal dibangkitkan oleh kunci masukan pengguna. Ukuran kunci yang diterima adalah 128 bit. Kunci 128 bit dibagi ke dalam enam belas kelompok berukuran 8 bit. Kelompok 8 bit ini akan disubstitusi dengan kotak substitusi menghasilkan kelompok-kelompok 4 bit. Kotak substitusi yang digunakan untuk pembangkitan kunci berisi empat baris dengan angka 0 - 3 dan enam belas kolom dengan dengan angka 0 - 15. Jadi setiap baris pada kotak substitusi berisi angka 0 sampai 15.

0	1	3	10	2	14	7	9	8	5	11	12	6	4	15	13
3	10	11	2	15	12	4	14	1	8	5	0	9	6	13	7
3	8	2	4	0	6	7	5	1	9	14	11	13	10	12	15
10	2	14	3	15	11	12	9	0	5	1	13	7	4	8	6

Tabel 1. Kotak substitusi pembangkita kunci internal

Kotak substitusi yang digunakan dihasilkan dengan urutan sesuai keinginan pembuat algoritma. Dari setiap kelompok 8 bit, Dua bit pertama dan dua bit terakhir dilakukan operasi XOR untuk menentukan baris. Bit 3 sampai bit 6 menunjukkan kolom pada kotak substitusi. Tabel di bawah menunjukkan kelompok 8 bit yang menghasilkan elemen tabel di baris 3 dan kolom 6.

1	**2**	3	4	5	6	**7**	**8**
1	**0**	0	1	1	0	**0**	**1**

Tabel 2. Penentuan baris dan kolom kotak substitusi dari kelompok 8 bit

Hasil substitusi menghasilkan 64 bit kunci internal. Kunci internal selanjutnya dihasilkan dengan menggeser kunci eksternal sebelas bit ke kiri secara sikuler dan subsitusi kembali. Proses ini dilakukan sebanyak jumlah putaran di jaringan feistel.

C. Round Function

Kunci internal yang dihasilkan digunakan pada setiap putaran fungsi enkripsi F untuk bagian kanan di jaringan feistel. Prosedur enkripsi fungsi F di jaringan feistel adalah:

1. Kunci internal ke-i di-*XOR*-kan dengan bagian kanan
2. Geser hasil operasi *XOR* 11 bit ke kiri secara sikuler
3. Bentuk kelompok 8 bit dengan aturan tertentu
4. Masukan kelompok 8 bit kedalam matriks berukuran 8x8 dari kiri atas sampai kanan bawah
5. Tranposisi kolom matriks dengan aturan tertentu
6. Bentuk kembali kelompok 8 bit berdasarkan urutan baris
7. substitusi kelompok 8 bit ke dalam kotak substitusi

Prosedur ini dilakukan setiap putaran di jaringan feistel.

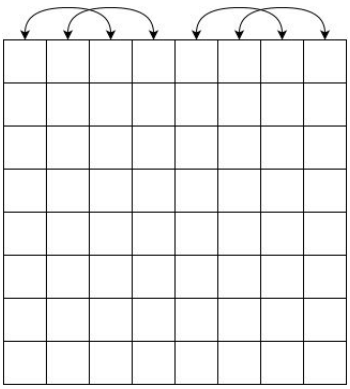
Pada tahap 3, pembentukan kelompok 8 bit dihasilkan dari pergeseran 64 bit secara sikuler dan aturan pembentukan kelompok. Aturan yang digunakan adalah 64 bit hasil pergesera di potong ke dalam kelompok 8 bit. Setelah perpotongan, kelompok 1 dihasilkan dari bit pertama setiap kelompok perpotongan. Pengelompokan berdasarkan posisi bit pada kelompok perpotongan menghasilkan delapan kelompok. Tabel di bawah menggambarkan pembentukan kelompok 8 bit.

Kelompok	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
8-bit kelompok pertama perpotongan	1	1	0	1	1	0	0	1

Tabel 3. Pembentukan kelompok 8 bit berdasarkan posisi bit setiap kelompok perpotongan

Pada tahap 4, kelompok 8 bit yang dihasilkan dimasukkan ke dalam matriks 8x8. Aturan pengisian sel matriks dilakukan dari ujung kiri atas sampai kanan bawah. Baris pertama matriks diisi dengan kelompok satu, kemudian baris berikutnya diisi dengan kelompok dua dan seterusnya. Setiap kolom pada baris diisi oleh bit-bit dalam kelompok 8 bit secara berurutan.

Pada tahap 5, kolom-kolom matriks dilakukan tranposisi sehingga susunan matriks berubah. Aturan tranposisi yang digunakan adalah



Gambar 5. Proses tranposisi ko

Pertukaran kolom dilakukan dengan menukar kolom 1 dengan kolom 3, kolom 2 dengan kolom 4, kolom 5 dengan kolom 7, dan kolom 6 dengan kolom 8.

Pada tahap 7, kelompok 8 bit akan disubstitusi dengan kotak substitusi. Kotak substitusi yang digunakan adalah

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	0F	80	BF	65	75	47	51	BE	C4	60	67	48	C0	E1	7C	C5
1	2F	A0	E3	00	10	36	7F	E2	35	76	52	28	0E	15	A8	27
2	81	66	37	AA	50	74	05	18	A5	29	17	04	16	1C	4C	68
3	71	11	70	64	19	46	2A	1E	C1	61	A2	1D	26	49	7B	0D
4	3A	20	07	73	2B	7E	62	34	AC	77	53	6A	0A	4B	AD	69
5	1F	45	1A	C3	63	A1	2C	C6	7D	C2	2D	1B	78	54	79	4A
6	82	72	E3	3B	E5	C7	12	E6	09	90	C8	25	03	14	A7	55
7	3C	A9	83	08	21	A4	B7	22	56	8F	4E	B8	4D	24	AA	B9
8	F4	98	30	5D	4F	02	B6	33	2E	13	A3	5F	88	C9	D8	0C
9	3F	99	84	96	8D	32	06	8E	57	23	DA	6B	D9	B5	CA	B4
A	9B	01	31	AB	95	42	A6	5E	AF	DB	0B	B2	AE	D7	91	F1
B	3E	FF	6F	FE	D1	E7	BA	D0	94	FB	D6	EF	FA	F0	89	F2
C	3D	8C	BB	97	5B	43	E8	41	DC	86	58	6C	F3	CB	92	9A
D	9C	ED	39	FD	6E	EC	38	CF	FC	B0	EA	EE	F5	E0	8A	99
E	44	5C	9D	85	D2	E9	9F	6D	DD	CE	DE	87	F6	CC	D1	0A
F	BC	8B	9E	D3	8A	EB	D4	BD	59	40	D5	B1	F7	DF	E8	B3

Tabel 4. Kotak substitusi pada proses enkripsi

Kotak substitusi yang digunakan dihasilkan dengan urutan sesuai keinginan pembuat algoritma. Dari setiap kelom 8 bit, empat bit pertama menunjukkan baris dan empat bit terakhir menunjukkan kolom dalam matriks.

Hasil kelompok 8 bit disusun ke dalam 64 bit secara berurutan dari kelompok pertama. Susunan 64 bit di-*XOR*-kan dengan 64 bit bagian kiri jaringan feistel. Putaran ini dilakukan sebnyan 16 putaran.

IV. Eksprerimen dan Pembahasan Hasil

Berikut hasil eksperimen yang dilakukan terhadap pesan dengan kunci "TOO EZPZ EZPZ EZ". Proses enkripsi dan dekripsi dilakukan ke dalam Algoritma yang telah kami buat.

A. Mode ECB

Proses enkripsi dan dekripsi dilakukan dengan kunci yang ditetapkan. Proses ini digunakan pada pesan berukuran 32-byte. Pesan akan direpresentasikan dalam string dan hex. Hasil enkripsi dan dekripsi akan ditampilan dalam hex tiap byte.

Pesan (String)	Lorem ipsum dolor sit amet, cons
Pesan (Hex)	4c6f 7265 6d20 6970 7375 6d20 646f 6c6f 7220 7369 7420 616d 6574 2c20 636f 6e73

Teks Cipher (Hex)	6832 eb3a d194 32d1 dfee da3a 7751 8899 8633 970a 299a 3191 4ab1 8b03 8017 bb6f

Teks dekripsi (Hex)	4c6f 7265 6d20 6970 7375 6d20 646f 6c6f 7220 7369 7420 616d 6574 2c20 636f 6e73

Hasil enkripsi menunjukkan algoritma ini dapat melakukan enkripsi dan dekripsi dengan baik.

V. Analisis

A. Analisis terhadap sejumlah serangan

Metode *bruteforce* menemukan kunci yang sesuai membutuhkan waktu yang lama. Hal ini dikarenakan karena jumlah bit kunci yang cukup panjang. Metode bruteforce membutuhkan 2^{128} kemungkinan kunci. Jika setiap detik percobaan dilakukan sebanyak 1 juta kemungkinan, maka dibutuhkan ratusan ribu tahun untuk menemukan kunci yang sesuai. Sehingga algoritma ini cukup kuat terhadap serangan *bruteforce attack*.

B. Analisis Frekuensi

Metode analisis frekuensi menggunakan kemunculan suatu huruf dalam suatu bahasa. Algoritma ini menggunakan substitusi dan tranposisi sehingga huruf yang sama tidak menghasilkan teks *cipher* yang sama. Orang yang melakukan metode analisis frekuensi akan kesulitan untuk menentukan huruf yang sesuai dengan pesan.

VI. Kesimpulan

Algoritma yang kami kembang cukup baik dalam menyembunyikan sebuah pesan. Algoritma ini cocok untuk melakukan enkripsi pada pesan yang akan dikirimkan pada jaringan publik seperti internet. Ukuran kunci yang cukup panjang akan menyulitkan kriptanalis dalam membongkar pesan yang dikandung. Algoritma ini menerapkan prinsip konfusi dan difusi. Hubungan pesan dan teks *cipher* tersembunyi cukup baik.

Keberlanjutan algoritma ini sebaiknya melakukan analisis dan eksperimen lebih jauh. Kepastian tingkat keamanan dari algoritma ini sangat dibutuhkan agar bisa digunakan oleh banyak orang.

Refrensi

[1] Munir, Rinaldi. *Algoritma Kriptografi Modern*. Oktober 2015. Presentasi PowerPoint

[2] Hong, Deukjo, dkk. *HIGHT: A New Block Cipher Suitable for Low-Resource Device*.

[3] S, Jefferey, dkk. *MARC - A New Block Cipher Algorithm. TechTarget. Rijndael.* http://searchsecurity.techtarget.com/definition/Rijndael *diakses pada 28 Oktober 2016, 08.00 WIB.*

Pernyataan

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi

Bandung, 28 Oktober 2016

Muhammad Aodyra Khaidir (13513063)

Muhammad Visat Sutartno(13513037)

0	1	3	10	2	14	7	9	8	5	11	12	6	4	15	13
3	10	11	2	15	12	4	14	1	8	5	0	9	6	13	7
3	8	2	4	0	6	7	5	1	9	14	11	13	10	12	15
10	2	14	3	15	11	12	9	0	5	1	13	7	4	8	6

Tabel 1. Kotak substitusi pembangkita kunci internal

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	0F	80	BF	65	75	47	51	BE	C4	60	67	48	C0	E1	7C	C5
1	2F	A0	E3	00	10	36	7F	E2	35	76	52	28	0E	15	A8	
2	81	66	37	AA	50	74	05	18	A5	29	17	04	16	1C	4C	
3	71	11	70	64	19	46	2A	1E	C1	61	A2	1D	26	49	7B	
4	3A	20	07	73	2B	7E	62	34	AC	77	53	6A	0A	4B	AD	
5	1F	45	1A	C3	63	A1	2C	C6	7D	C2	2D	1B	78	54	79	
6	82	72	E3	3B	E5	C7	12	E6	09	90	C8	25	03	14	A7	
7	3C	A9	83	08	21	A4	B7	22	56	8F	4E	B8	4D	24	7A	
8	F4	98	30	5D	4F	02	B6	33	2E	13	A3	5F	88	C9	D8	
9	3F	99	84	96	8D	32	06	8E	57	23	DA	6B	D9	B5	CA	
A	9B	01	31	AB	95	42	A6	5E	AF	DB	0B	B2	AE	D7	91	
B	3E	FF	6F	FE	D1	E7	BA	D0	94	FB	D6	EF	FA	F0	89	
C	3D	8C	BB	97	5B	43	E8	41	DC	86	58	6C	F3	CB	92	
D	9C	ED	39	FD	6E	EC	38	CF	FC	B0	EA	EE	F5	E0	5A	
E	44	5C	9D	85	D2	E9	9F	6D	DD	CE	DE	87	F6	CC	F9	
F	BC	8B	9E	D3	8A	EB	D4	BD	59	40	D5	B1	F7	DF	F8	

Tabel 4. Kotak substitusi pada proses enkripsi