Algoritma Block Cipher RF1

Micky Yudi Utama
Program Studi Teknik Informatika
Institut Teknologi Bandung
Jl. Ganesha 10 Bandung 40132, Indonesia
mickyyu96@gmail.com

Varian Caesar
Program Studi Teknik Informatika
Institut Teknologi Bandung
Jl. Ganesha 10 Bandung 40132, Indonesia
variancaesar@gmail.com

***Abstrak*—Kebutuhan akan keamanan dalam berkomunikasi telah melahirkan banyak jenis algoritma kriptografi. Block Cipher adalah teknik pengkodean pesan pada kriptografi modern yang bekerja pada sekumpulan blok pesan. Sampai saat ini telah banyak algoritma Block Cipher yang berhasil dibuat, seperti AES, DES, RC5, dll. Pada makalah ini kami mengusulkan algoritma Block Cipher baru yang diberi nama RF1. Algoritma ini cukup sederhana namun sukar dipecahkan. RF1 menggunakan jaringan Feistel di dalamnya dan menerapkan prinsip *confusion* dan *diffusion* dari Shannon. Pada proses enkripsi, dilakukan iterasi sebanyak 16 putaran. Setiap putaran akan dibangkitkan kunci internal yang unik yang diturunkan dari kunci eksternal yang dimasukkan pengguna. Kunci internal ini akan digunakan sebagai masukan dalam *round function* yang akan dijelaskan kemudian. Makalah ini akan membahas secara detil mengenai algoritma RF1.**

Kata kunci—block cipher, jaringan feistel, kriptografi, round function.

I. PENDAHULUAN

Dalam melakukan komunikasi, terkadang pesan yang ingin disampaikan dapat saja merupakan pesan rahasia. Sejak zaman Julius Caesar, sudah dikenal berbagai cara untuk mengamankan pesan melalui modifikasi terhadap data yang akan dikirim. Modifikasi ini bertujuan agar pesan yang akan dikirim hanya diketahui oleh pengirim dan penerima pesan saja. Ilmu atau seni yang mempelajari mengenai penyimpanan kerahasiaan pesan melalui modifikasi data menjadi bentuk yang tidak dapat dimengerti lagi disebut sebagai kriptografi.

Sejarah penggunaan penyimpanan pesan rahasia tertua dapat ditemukan pada peradaban Mesir Kuno, tahun 3000 SM. Pada saat itu, Bangsa Mesir menggunakan ukiran rahasia yang disebut sebagai *hieroglyphics* untuk menyampaikan pesan pada orang tertentu. Sejak itu, telah banyak teknik penyimpanan pesan yang diciptakan. Teknik yang paling umum digunakan adalah metode substitusi dan transposisi. Seiring dengan perkembangan teknologi, perkembangan teknik kriptografi pun berkembang pesat dan muncullah istilah kriptografi modern. Kriptografi modern yang menggunakan komputer dalam pengoperasiannya merupakan teknik yang memiliki algoritma yang kompleks dan kekuatan kriptografinya ada pada kuncinya. Pada kriptografi modern, perubahan data pesan dilakukan pada skala bit atau biner sehingga dibutuhkan pengetahuan mengenai matematika untuk memahaminya.

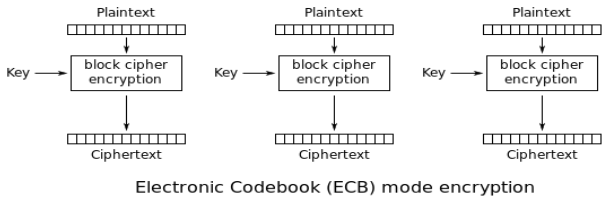
Salah satu teknik kriptografi modern yang sering digunakan adalah Block Cipher. Block Cipher merupakan algoritma kriptografi simetrik yang membagi pesan menjadi blok-blok dengan ukuran tertentu. Setiap blok akan dimodifikasi dengan operasi tertentu sehingga menghasilkan blok dengan ukuran yang sama. Pengembangan pada Block Cipher telah banyak dilakukan. Beberapa jenis algoritma yang sering digunakan seperti DES, GOST, RC5, Rijndael, dll. Pada makalah ini, kami mengajukan algoritma Block Cipher baru yakni RF1. Algoritma RF1 memanfaatkan Jaringan Feistel. Pada *round function*-nya dilakukan transposisi matriks, perkalian matriks, dan substitusi dengan memanfaatkan S-Box. Pengembangan algoritma ini diharapkan dapat memberikan kontribusi pada ilmu kriptografi.

II. DASAR TEORI

A. Block Cipher

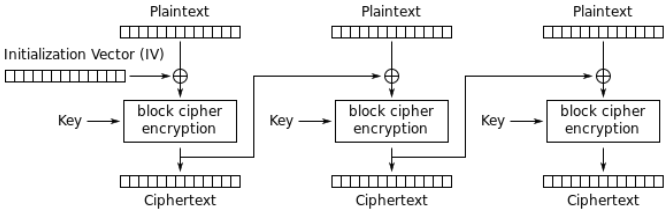
Block Cipher adalah algoritma kriptografi yang memproses sekumpulan bit tertentu pada pesan yang disebut sebagai blok. Pada algoritma ini, pesan dibagi menjadi beberapa blok. Setiap blok akan dienkripsi dengan menggunakan kunci untuk menghasilkan cipherteks dengan panjang yang sama. Pada Block Cipher, dikenal lima mode operasi, yakni Electronic Code Book (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFB), Output Feedback (OFB), dan Mode Counter.

1. Electronic Code Book (ECB)
   Pada mode ECB, setiap blok pada plainteks dienkripsi secara individual dan independen menjadi blok cipherteks. Keuntungannya adalah proses enkripsi dapat dilakukan secara paralel dan kesalahan 1 bit atau lebih pada blok cipherteks tidak akan mempengaruhi blok lainnya. Kelemahannya adalah plainteks yang sama menghasilkan cipherteks yang sama sehingga dapat diserang secara statistik. Gambar 1 merupakan skema enkripsi dari mode ECB.



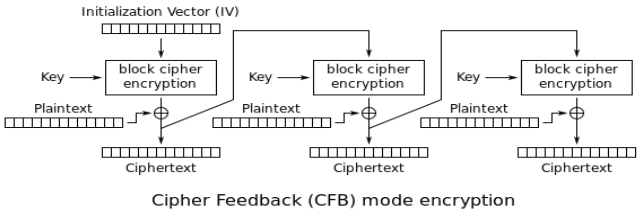
Gambar 1. Skema enkripsi pada mode ECB

2. Cipher Block Chaining (CBC)
   Mode CBC bertujuan untuk membuat ketergantungan antar blok. Pada mode ini, setiap blok plainteks akan di-XOR terlebih dahulu dengan blok cipherteks pada tahap sebelumnya sebelum dienkripsi. Enkripsi pada blok pertama menggunakan blok semu yang disebut sebagai *Initialization Vector* (IV). Blok semu ini dapat ditentukan oleh pengguna atau dibangkitkan secara acak oleh program. Keuntungan dari mode CBC adalah sulit dipecahkan oleh kriptanalis karena plainteks yang sama belum tentu menghasilkan cipherteks yang sama. Kelemahannya adalah kesalahan 1 bit pada sebuah blok plainteks dapat memberikan efek pada blok cipherteks yang berkoresponden dan semua blok cipherteks berikutnya. Gambar 2 merupakan skema enkripsi pada CBC.



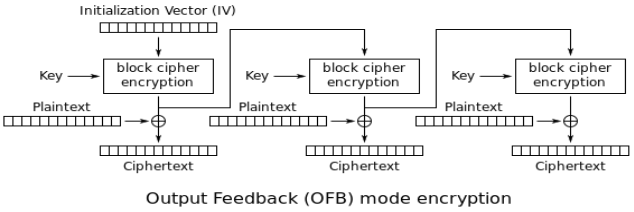
Gambar 2. Skema enkripsi pada mode CBC

3. Cipher Feedback (CFB)
   Mode CFB mirip dengan mode CBC. Perbedaannya adalah mode CFB melakukan enkripsi langsung pada IV. Hasil enkripsi kemudian di-XOR dengan plainteks untuk menghasilkan cipherteks. Cipherteks tersebut dijadikan IV untuk tahap selanjutnya. Sama seperti pada CBC, kesalahan 1 bit pada blok plainteks dapat merambat pada blok cipherteks yang berkoresponden dan semua blok cipherteks berikutnya. Gambar 3 merupakan skema enkripsi pada CFB.



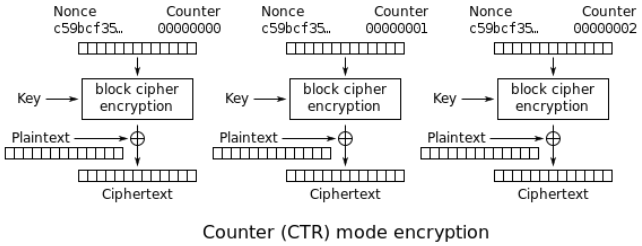
Gambar 3. Skema enkripsi pada mode CFB

4. Output Feedback (OFB)
   Mode OFB mirip dengan mode CFB. Perbedaannya adalah IV pada mode ini berasal dari hasil enkripsi IV pada tahap sebelumnya. Tujuan dari metode ini adalah memperbaiki kelemahan pada CBC dan CFB. Kesalahan 1 bit pada plainteks hanya mempengaruhi cipherteks yang berkoresponden saja. Selain itu proses enkripsi dapat dilakukan secara semi paralel. Ketika enkripsi IV selesai, proses enkripsi tahap selanjutnya sudah dapat dimulai tanpa menunggu proses XOR selesai. Gambar 4 merupakan skema enkripsi mode OFB.



Gambar 4. Skema enkripsi pada mode OFB

5. Counter
   Mode Counter menggunakan sebuah *counter* yang awalnya diinisialisasi suatu nilai tertentu. Untuk setiap putaran, nilai *counter* akan dinaikkan sebesar satu satuan. Nilai *counter* ini yang akan masuk ke dalam *round function*, dan hasilnya akan di-XOR dengan blok plainteks untuk menghasilkan blok cipherteks. Dengan mode Counter, kesalahan pada 1 bit pada blok plainteks tidak akan merambat pada blok-blok lainnya. Gambar 5 merupakan skema enkripsi dari mode Counter.



Gambar 5. Skema enkripsi pada mode Counter

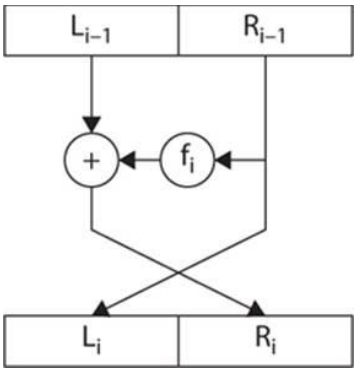
B. Properti *Confusion* dan *Diffusion*

Prinsip konfusi dan difusi (*confusion and diffusion*) pertama kali diperkenalkan oleh Claude Shannon pada tahun 1949 dalam publikasi yang berjudul *Communication Theory of Secret System*. Kedua prinsip ini digunakan untuk mempersulit pihak ketiga maupun kriptanalis dalam melakukan kriptanalisis.

Prinsip konfusi berarti memastikan bahwa cipherteks dan kunci memiliki hubungan yang rumit, semakin rumit akan semakin baik. Sedangkan, prinsip difusi berarti menyebarkan pengaruh satu bit pada plainteks dan kunci sebanyak mungkin pada bit cipherteks. Kedua prinsip ini dapat dipenuhi dengan menerapkan transformasi dan substitusi pada proses penyandian.

C. Jaringan Feistel

Jaringan Feistel merupakan struktur simetris yang digunakan pada kebanyakan konstruksi algoritma Block Cipher. Jaringan Feistel ditemukan oleh seorang fisikawan dari Jerman, Horst Feistel pada tahun 1970. Prinsip dari jaringan Feistel adalah dengan membagi pesan menjadi 2 bagian, yaitu L_0 dan R_0. Lalu definisikan sebuah *round function f* sedemikian sehingga L_n dan R_n, yaitu bagian kiri dan kanan pesan pada iterasi ke-n, merupakan hasil keluaran fungsi *f* untuk masukan L_{n-1} dan R_{n-1}. *Round function* yang digunakan pada setiap iterasi ditentukan oleh perancang algoritma dan sebaiknya mengandung transformasi dan substitusi agar menerapkan prinsip konfusi dan difusi. Struktur jaringan Feistel dapat dilihat pada Gambar 6.



Gambar 6. Jaringan Feistel

Salah satu alasan mengapa jaringan Feistel banyak digunakan pada Block Cipher karena fungsi yang digunakan untuk enkripsi akan sama dengan fungsi yang digunakan pada proses dekripsi. Sehingga tidak diperlukan tahapan yang berbeda antara proses enkripsi dan dekripsi. Hal ini bisa tercapai karena jaringan Feistel memiliki struktur yang *reversible* dan tidak terpengaruh dengan fungsi *f* yang digunakan. Fungsi *f* dapat dibuat serumit mungkin.

D. Kotak-S (S-Box)

Kotak-S (S-Box) adalah sebuah matriks yang berisi substitusi sederhana yang memetakan satu atau lebih bit dengan satu atau lebih bit yang lain. Kotak-S memetakan masukan berukuran m bit menjadi keluaran berukuran n bit, sehingga kotak-S tersebut dinamakan kotak *m x n* S-Box. Pada contoh Kotak-S berukuran 6 x 4, bit MSB dan LSB pada pesan berukuran 6 bit akan menjadi nomor baris dan 4 bit sisanya akan menjadi nomor kolom. Entri yang diperoleh pada Kotak-S akan diubah ke biner dan digunakan untuk mengganti pesan 6 bit tersebut, kini pesan tersebut memiliki panjang 4 bit.

III. RANCANGAN ALGORITMA

Algoritma yang diusulkan diberi nama RF1. Algoritma ini mengadaptasi algoritma Block Cipher dengan memanfaatkan Jaringan Feistel. Masukan dari algoritma ini adalah kunci sepanjang 128 bit dan blok plainteks sepanjang 128 bit. Jumlah iterasi dilakukan sebanyak 16 putaran. Setiap putaran menggunakan kunci internal yang dibangkitkan dari kunci eksternal.

A. Pembangkitan Kunci Internal

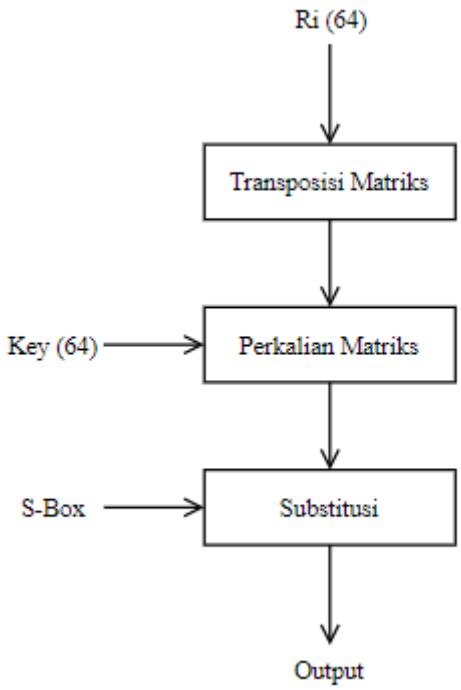
Tahap pembangkitan kunci internal dimulai dengan membagi kunci eksternal menjadi 2 bagian yang sama panjang, yakni 64 bit untuk masing-masing yang kemudian disebut L dan R. Pada putaran ke-n, L akan digeser sejauh 4 (n-1) satuan ke kiri secara sirkuler dan R akan digeser sejauh 4×(n-1) satuan ke kanan secara sirkuler. Hasil pergeseran dari L dan R pada putaran ke-n kemudian di XOR untuk menghasilkan kunci internal pada putaran tersebut. Berikut adalah contoh pembangkitan kunci internal pada putaran ke-3.

Kunci eksternal:
01101001 01101110 01101001 01100001 01100100
01100001 01101100 01100001 01101000 01101011
01110101 01101110 01100011 01101001 01101110
01111001

L:
01101001 01101110 01101001 01100001 01100100
01100001 01101100 01100001
R:
01101000 01101011 01110101 01101110 01100011
01101001 01101110 01111001
Hasil pergeseran L sejauh 4×(3-1) ke kiri:
01101110 01101001 01100001 01100100 01100001
01101100 01100001 01101001
Hasil pergeseran R sejauh 4×(3-1) ke kanan:
01111001 01101000 01101011 01110101 01101110
01100011 01101001 01101110
Hasil XOR – Kunci internal putaran ke-3:
00010111 00000001 00001010 00010001 00001111
00001111 00001000 00000111

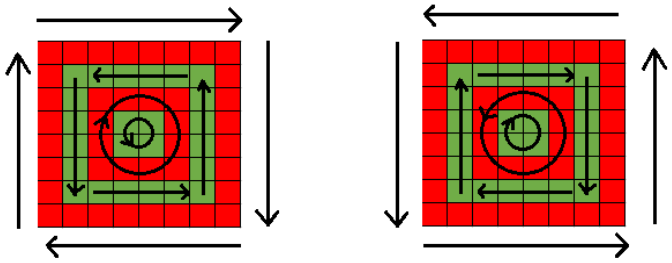
B. *Round Function*

Masukan dari Jaringan Feistel adalah blok plainteks dan kunci eksternal. Blok plainteks 128 bit tersebut kemudian dipecah menjadi dua bagian, yaitu *Left* dan *Right*, yang masing-masing memiliki panjang 64 bit. Jaringan Feistel pada RF1 memiliki 16 putaran dimana masing-masing putaran menggunakan kunci internal yang dibangkitkan dari kunci eksternal melalui proses pada subbab sebelumnya. Pada setiap putaran di Jaringan Feistel, terdapat *round function*. Sebelum bagian dari blok plainteks dimasukkan ke *round function*, dilakukan perubahan representasi masukan dahulu pada *Left* dan *Right* menjadi matriks berukuran 8x8 dimana setiap sel mengandung 1 bit. Pengubahan ke dalam bentuk matriks juga dilakukan pada kunci internal. Pengisian matriks dilakukan secara sekuensial dari kiri sampai kanan, lalu dlanjutkan ke baris berikutnya. *Round function* dari algoritma RF1 terdiri dari tiga tahap, yakni transposisi matriks, perkalian matriks, dan substitusi dengan memanfaatkan S-Box.



Gambar 7. Ilustrasi *round function* pada algoritma RF1

1. Transposisi Matriks
   Transposisi matriks dilakukan dengan melakukan rotasi pada matriks seperti yang ditunjukkan pada Gambar 5. Pada putaran ke-n, apabila n ganjil maka seluruh bit yang terletak pada kotak warna merah akan diputar searah jarum jam dan seluruh bit yang terletak pada kotak warna hijau akan diputar berlawanan jarum jam. Apabila n genap, maka sebaliknya, seluruh bit pada kotak warna merah akan diputar berlawanan jarum jam dan bit pada kotak warna hijau akan diputar searah jarum jam.



Gambar 8. Ilustrasi transposisi matriks

2. Perkalian Matriks
   Hasil dari transposisi matriks pada tahap sebelumnya dikalikan secara *dot product* dengan matriks kunci internal dan menghasilkan matriks dengan ukuran yang sama, yakni 8x8. Selanjutnya, setiap sel pada matriks hasil dimodulo dengan nilai 2 agar menghasilkan matriks yang berisi nilai biner.
3. Substitusi
   Substitusi dilakukan dengan memanfaatkan S-Box. Contoh dari S-Box yang digunakan dapat dilihat pada bagian lampiran. Proses substitusi dilakukan dengan cara mengganti setiap baris pada matriks pesan. Oleh karena itu, proses penggantian dilakukan sebanyak 8 kali. Pada setiap baris, proses penggantian dilakukan dengan mengambil 4 bit pertama sebagai masukan baris pada *S-box* dan 4 bit selanjutnya sebagai masukan kolom pada *S-box*. Keluaran dari *S-box* adalah 8 bit yang kemudian digunakan untuk mensubstitusi baris yang bersangkutan. Misalkan baris pertama pada matriks pesan adalah 1011 0110. Dalam bentuk *hexa* adalah B6. Jika dimasukkan ke *S-box* maka didapatkan keluaran berupa 4E atau dalam bentuk biner adalah 0100 1110. Ke-8 bit tersebut kemudian digunakan untuk mensubstitusi baris pertama.

Skema pada Gambar 7, digunakan pada proses enkripsi maupun dekripsi. Karena menggunakan jaringan Feistel, tidak perlu mendefinisikan skema baru untuk dekripsi.

IV. EKSPERIMEN DAN ANALISIS HASIL

Eksperimen dilakukan dengan menerapkan algoritma RF1 yang telah dibuat pada 5 mode, yakni ECB, CBC, CFB, OFB, dan Counter. Kunci yang digunakan adalah **KUNCIKRIPTOGRAFI**. S-Box yang digunakan terdapat pada bagian lampiran. Plainteks yang digunakan adalah sebagai berikut.

> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Cipherteks hasil dekripsi akan ditampilkan dalam bentuk hex untuk mempermudah pembacaan. Representasi dalam bentuk karakter tidak dapat digunakan karena beberapa simbol tidak dapat ditampilkan.

A. Pengujian pada Mode ECB

Pada mode ECB, setiap blok plainteks dienkrip secara individu dan independen. Berikut merupakan hasil enkripsi plainteks dengan menggunakan mode ECB.

> 64 66 66 96 14 11 9A 77 58 47 4C DB 87 B5 B6 85 99 63 38 4B B8 2D 2E 4B EB BB 0E 09 30 97 AC C8 DD F2 7E 3B 83 07 74 03 DA 5B F1 E3 B0 FD 7C AA E0 32 B8 C8 D7 1E A4 D4 FA AA 91 66 56 28 BC 64 9F 59 71 BE 00 C8 5F F2 B5 9F AD C9 A9 E1 BE 8C BF 8B DD DA 40 BD 93 05 CF 6B 80 24 6B 14 AC 3C 96 7D 93 C1 5B 98 45 88 C1 FB D7 60 29 7C 03 13 A9 C2 15 BA 71 44 52 9C ED 6E 68 BF B2 95 AC C3

B. Pengujian pada Mode CBC

Pada mode CBC, sebelum blok plainteks dienkripsi, blok tersebut di-XOR terlebih dahulu dengan IV (*Initialization Vector*). Pada putaran pertama, IV yang digunakan adalah **INITIALIZATIVECT**. Pada putaran selanjutnya, IV yang digunakan adalah blok cipherteks pada putaran sebelumnya. Berikut merupakan hasil enkripsi plainteks dengan menggunakan mode CBC.

> F0 D1 F7 28 10 2D D3 03 F9 FE 0E E7 16 7A 84 23 29 15 8F 09 2A EC 79 0E 99 A6 EC 75 55 44 1C 32 00 9C 3D 4A 82 CB AB 69 7D 98 8B FF 24 9B 20 30 3D 33 3A CB 81 64 A2 66 2E 65 44 C9 06 07 38 D4 C9 C0 B4 05 7F 06 FA F0 6E 27 1D 4E 20 5C DA 4E 37 7E C6 1C 05 59 C6 BE E5 62 62 96 61 E2 E2 4E 37 B5 FC D9 19 85 16 7A 68 4A AF FB 33 6B 69 74 F2 A5 EC 46 9D A5 67 19 C9 73 36 CA 12 EA 7C 7B

C. Pengujian pada Mode CFB

Pada mode CFB, enkripsi dilakukan pada IV (*Initialization Vector*). Blok cipherteks merupakan hasil enkripsi IV yang telah di-XOR dengan blok plainteks. Pada putaran pertama, IV yang digunakan adalah **INITIALIZATIVECT**. Pada putaran selanjutnya, IV yang digunakan adalah blok cipherteks pada putaran sebelumnya. Berikut merupakan hasil enkripsi plainteks dengan menggunakan mode CFB.

> 39 B7 EB 57 26 FD EB C9 52 3B 4F AF A2 77 95 6B BA 43 55 43 25 5B E4 50 14 4A AA 8A 8E 0F BE D7 3C A4 51 6B BF 99 1E 90 2D F5 7A A8 19 64 92 13 0E 66 54 B3 00 31 1A 62 01 45 43 A9 DD D2 CF E6 36 88 F1 8B DE 78 A4 3C 67 BA 37 67 C5 59 AB 9B 3B 99 35 E1 2A 7B 8C C8 44 40 A5 04 64 DF 4A E4 44 51 DF 5B A4 34 91 67 64 C9 29 69 BB B2 F0 A5 AB F8 4D A4 D0 96 89 3D 81 BD BE 1D 44 F6 08 29

D. Pengujian pada Mode OFB

Mode OFB mirip dengan mode CFB. Yang membedakan adalah IV pada putaran kedua dan selanjutnya dari mode OFB menggunakan hasil enkripsi dari putaran sebelumnya. Berikut merupakan hasil enkripsi plainteks dengan menggunakan mode OFB.

39 B7 EB 57 26 FD EB C9 52 3B 4F AF A2 77 95 6B 21 E9 C6 37 AD 0A 82 23 F1 6A 5C EA B8 91 83 91 B7 C5 AD 17 24 BC 25 85 AF 66 EE 2E 0E 87 41 4F B1 F2 4D 1F 97 EE 81 E9 05 29 3E 07 2C 35 1C 01 60 29 9E AA 97 F3 64 B1 52 ED 58 9A CF 2A EB 8F 71 30 E3 59 3E D9 8C 34 25 73 B8 88 42 7A 88 24 36 14 2F E6 E6 62 AC 4F 9A 12 F2 32 42 49 13 4A 2A A0 CC E1 6B 6C 85 AA 6E 7F 12 94 E6 C0 84 B0

E. Pengujian pada Mode Counter

Pada mode ini, enkripsi dilakukan pada Counter. Counter adalah sebuah nilai berupa blok bit yang memiliki ukuran yang sama dengan blok plainteks. Hasil enkripsi akan di-XOR dengan blok plainteks untuk menghasilkan blok cipherteks. Pada putaran selanjutnya, nilai Counter akan dinaikkan sebesar satu sebelum dienkripsi. Berikut merupakan hasil enkripsi plainteks dengan menggunakan mode Counter.

59 F8 FA C7 FE BC 6F 9D A4 BE 35 B9 C6 B3 A7 37 7D 49 85 6E 64 18 F0 57 A2 7D AB 88 B3 BA D2 B5 34 01 90 DC FA D5 43 5B BD 84 50 A6 A7 69 15 36 02 3D CA 1A D5 51 2F E6 10 06 2E 0D 11 0B 63 61 8B EC C4 D5 EE F2 84 3E B5 29 40 A4 78 34 93 9F 24 64 C2 F0 8C 71 E1 FC 7B 52 7F FD 30 71 49 78 03 42 3B 2E E6 9A FE 46 5F 1A 94 9D 48 E0 77 5A 66 9B FF FF 86 53 77 E9 17 4B 02 AB A6 43 37 D6

Selanjutnya, akan dilakukan analisis mengenai pengaruh sedikit perubahan pada plainteks, kunci, dan cipherteks. Analisis hanya dilakukan pada mode ECB dan CBC, dimana ECB mewakili mode yang tidak menggunakan prinsip *chaining* dan CBC mewakili mode yang menggunakan prinsip *chaining*. Untuk mempermudah, plainteks dan cipherteks yang digunakan akan diperpendek. Bagian yang berbeda juga akan dicetak tebal dan diberi warna merah.

A. Analisis Sedikit Perubahan pada Plainteks

Kunci yang digunakan adalah **KUNCIKRIPTOGRAFI**.

ECB:

Lorem ipsum dolor sit amet, con**s**ectetur adipiscing elit, sed do eiusmod tempor incididunt
64 66 66 96 14 11 9A 77 58 47 4C DB 87 B5 B6 85 **99 63 38 4B B8 2D 2E 4B EB BB 0E 09 30 97 AC C8** DD F2 7E 3B 83 07 74 03 DA 5B F1 E3 B0 FD 7C AA E0 32 B8 C8 D7 1E A4 D4 FA AA 91 66 56 28 BC 64 9F 59 71 BE 00 C8 5F F2 B5 9F AD C9 A9 E1 BE 8C 5C AF 01 DF BE 7A 82 3B FD DB DC 0B 8F F6 DD 07

Lorem ipsum dolor sit amet, con**c**ectetur adipiscing elit, sed do eiusmod tempor incididunt

64 66 66 96 14 11 9A 77 58 47 4C DB 87 B5 B6 85 **D3 64 5E 92 1A 11 F6 89 DB 89 50 58 8F B2 DF F4** DD F2 7E 3B 83 07 74 03 DA 5B F1 E3 B0 FD 7C AA E0 32 B8 C8 D7 1E A4 D4 FA AA 91 66 56 28 BC 64 9F 59 71 BE 00 C8 5F F2 B5 9F AD C9 A9 E1 BE 8C 5C AF 01 DF BE 7A 82 3B FD DB DC 0B 8F F6 DD 07

CBC:

Lorem ipsum dolor sit amet, con**s**ectetur adipiscing elit, sed do eiusmod tempor incididunt
F0 D1 F7 28 10 2D D3 03 F9 FE 0E E7 16 7A 84 23 **29 15 8F 09 2A EC 79 0E 99 A6 EC 75 55 44 1C 32 00 9C 3D 4A 82 CB AB 69 7D 98 8B FF 24 9B 20 30 3D 33 3A CB 81 64 A2 66 2E 65 44 C9 06 07 38 D4 C9 C0 B4 05 7F 06 FA F0 6E 27 1D 4E 20 5C DA 4E 3D 2E 1C E8 1D 80 64 89 2A EA D8 6E 21 EA E1 04**

Lorem ipsum dolor sit amet, con**c**ectetur adipiscing elit, sed do eiusmod tempor incididunt
F0 D1 F7 28 10 2D D3 03 F9 FE 0E E7 16 7A 84 23 **9B 26 04 5F 51 B4 B1 44 FD 1E 2B 95 8D 76 E8 25 C1 D3 1E 79 43 4A 39 6C 31 18 89 6F 1C 45 CF 87 B3 F0 47 C5 4D 53 69 2A 25 93 F7 62 0B AB 9B 23 9C 64 C9 9C B0 49 F9 A7 A7 DA 5F 82 F3 82 BD AA BB D5 7B B7 F8 0F EB 65 25 AC 1A C3 85 EE CA E5**

Pada ECB, perubahan pada plainteks hanya berpengaruh pada blok yang bersesuaian saja. Sedangkan pada CBC, perubahan kecil pada plainteks dapat menyebabkan perambatan kesalahan sampai akhir pesan.

B. Analisis Sedikit Perubahan pada Kunci

Plainteks yang digunakan adalah ‘Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt’.

ECB:

KUNCIKR**I**PTOGRAFI
64 66 66 96 14 11 9A 77 58 47 4C DB 87 B5 B6 85 99 63 38 4B B8 2D 2E 4B EB BB 0E 09 30 97 AC C8 DD F2 7E 3B 83 07 74 03 DA 5B F1 E3 B0 FD 7C AA E0 32 B8 C8 D7 1E A4 D4 FA AA 91 66 56 28 BC 64 9F 59 71 BE 00 C8 5F F2 B5 9F AD C9 A9 E1 BE 8C 5C AF 01 DF BE 7A 82 3B FD DB DC 0B 8F F6 DD 07

KUNCIKR**A**PTOGRAFI
58 8C FE 26 D2 75 F3 D0 CE 62 40 E8 79 1C 21 EF DF C0 9C 54 D5 AE BC 69 38 5C CF 0D 71 D7 0C 07 A2 C0 CC 46 21 0F 92 E5 43 F2 CA C2 2C 85 14 4C AD BD EA 7D 91 A0 D2 99 D5 4C C3 C2 51 00 6B 5A AD 11 0A BB 88 FD 14 1C BD CB 06 04 2B 5B 7C C9 8D 1F C5 D1 BB 86 22 B9 50 0F F2 2A 94 5B A0 D6

CBC:

KUNCIKRIPTOGRAFI
F0 D1 F7 28 10 2D D3 03 F9 FE 0E E7 16 7A 84 23 29 15 8F 09 2A EC 79 0E 99 A6 EC 75 55 44 1C 32 00 9C 3D 4A 82 CB AB 69 7D 98 8B FF 24 9B 20 30 3D 33 3A CB 81 64 A2 66 2E 65 44 C9 06 07 38 D4 C9 C0 B4 05 7F 06 FA F0 6E 27 1D 4E 20 5C DA 4E 3D 2E 1C E8 1D 80 64 89 2A EA D8 6E 21 EA E1 04

KUNCIKRAPTOGRAFI
26 A0 2A 18 99 D9 4F 4B 77 01 92 6B EB 99 03 1F 8F A7 0A C2 DE FE 59 83 F1 52 27 39 3C 38 FF CC E3 61 9F 57 1C B7 5E 5D 45 CC 2F 09 AC D7 60 AE C5 97 1D D9 14 86 44 80 42 FD 89 A4 2A C7 EA 44 AE 16 C8 89 D1 64 C5 4C 2D 41 36 4C 17 CA 20 D7 A0 BD 4C 9D F9 81 7A 96 DA 14 5F 91 3F 53 DE 01

Untuk percobaan ini, perubahan kunci yang sedikit dapat mengubah 100% cipherteks yang dihasilkan. Hal ini disebabkan karena kunci internal yang terbentuk akan jauh berbeda total dengan kunci internal yang sebenarnya. Hal ini juga membuktikan bahwa usaha menebak kunci pada algoritma RF1 mustahil dilakukan, karena kunci yang berbeda sedikit saja akan menyebabkan perubahan yang signifikan.

C. Analisis Sedikit Perubahan pada Cipherteks

Kunci yang digunakan adalah **KUNCIKRIPTOGRAFI**.

ECB:

64 66 66 **96** 14 11 9A 77 58 47 4C DB 87 B5 B6 85 99 63 38 4B B8 2D 2E 4B EB BB 0E 09 30 97 AC C8 DD F2 7E 3B 83 07 74 03 DA 5B F1 E3 B0 FD 7C AA E0 32 B8 C8 D7 1E A4 D4 FA AA 91 66 56 28 BC 64 9F 59 71 BE 00 C8 5F F2 B5 9F AD C9 A9 E1 BE 8C 5C AF 01 DF BE 7A 82 3B FD DB DC 0B 8F F6 DD 07
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt0000000

64 66 66 **90** 14 11 9A 77 58 47 4C DB 87 B5 B6 85 99 63 38 4B B8 2D 2E 4B EB BB 0E 09 30 97 AC C8 DD F2 7E 3B 83 07 74 03 DA 5B F1 E3 B0 FD 7C AA E0 32 B8 C8 D7 1E A4 D4 FA AA 91 66 56 28 BC 64 9F 59 71 BE 00 C8 5F F2 B5 9F AD C9 A9 E1 BE 8C 5C AF 01 DF BE 7A 82 3B FD DB DC 0B 8F F6 DD 07
ŒГ• 9¸ß NŒf ..KÔr sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt0000000

CBC:

F0 D1 F7 28 10 2D D3 03 F9 FE 0E E7 16 7A 84 23 29 15 8F 09 2A EC 79 0E 99 A6 EC 75 55 44 1C 32 00 9C 3D 4A 82 CB AB 69 7D 98 8B FF 24 9B 20 30 3D 33 3A CB 81 64 A2 66 2E 65 44 C9 06 07 38 D4 C9 C0 B4 05 7F 06 FA F0 6E 27 1D 4E 20 5C DA **4E** 3D 2E 1C E8 1D 80 64 89 2A EA D8 6E 21 EA E1 04
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do **eiusmod tempor i**ncididunt0000000

F0 D1 F7 28 10 2D D3 03 F9 FE 0E E7 16 7A 84 23 29 15 8F 09 2A EC 79 0E 99 A6 EC 75 55 44 1C 32 00 9C 3D 4A 82 CB AB 69 7D 98 8B FF 24 9B 20 30 3D 33 3A CB 81 64 A2 66 2E 65 44 C9 06 07 38 D4 C9 C0 B4 05 7F 06 FA F0 6E 27 1D 4E 20 5C DA **FF** 3D 2E 1C E8 1D 80 64 89 2A EA D8 6E 21 EA E1 04
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do **^8íÙÂR¾èÇÓæ8ä** ncididunt000000•

Pada ECB, kesalahan sedikit pada cipherteks hanya mempengaruhi plainteks yang berkoresponden. Hal tersebut tidak jauh berbeda pada mode CBC, kesalahan sedikit pada cipherteks hanya mempengaruhi plainteks yang bersangkutan dan satu blok plainteks berikutnya.

V. ANALISIS KEAMANAN

Algoritma RF1 menerapkan properti konfusi dan difusi sehingga membuat serangan statistik lebih sulit dilakukan. *Round fucntion* dari algoritma RF1 juga tergolong kompleks karena menerapkan rotasi pada matriks, perkalian matriks, dan substitusi dengan menggunakan S-Box. Diharapkan hal tersebut dapat membuat kriptanalis semakin sulit untuk melakukan serangan. Bagian ini akan membahas mengenai beberapa jenis serangan yang mungkin terjadi seperti *brute force attack* dan analisis frekuensi.

A. *Brute Force Attack*

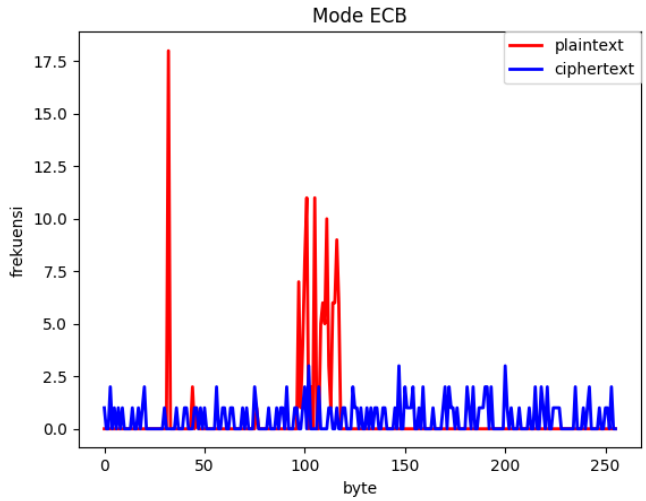
Serangan *brute force* adalah jenis serangan yang digunakan oleh kriptanalis untuk menyerang cipherteks dengan cara menebak dan mencoba setiap kemungkinan kunci yang ada. Serangan ini menggunakan metode *exhaustive search*, dimana kriptanalis mengenumerasi satu persatu kemungkinan kunci. Dengan metode ini, kunci kemungkinan besar dapat ditemukan, namun waktu yang dibutuhkan untuk menemukan kunci yang tepat dapat sangat lama tergantung pada jumlah kombinasi kunci yang ada.

Pada algoritma RF1, kunci yang digunakan berukuran 128 bit. Oleh karena itu, terdapat 2^{128} atau $3{,}4028237\times10^{38}$ kemungkinan kunci yang ada. Jika dengan kemampuan komputer dapat dilakukan percobaan pada 10.000 kunci setiap detik, maka waktu yang dibutuhkan untuk mencoba semua kemungkinan kunci adalah $3{,}40\times10^{32}$ detik atau $3{,}94\times10^{27}$ hari atau $1{,}085\times10^{25}$ tahun. Selain itu, terdapat pula S-Box yang digunakan untuk melakukan substitusi pada *round function*. Waktu yang dibutuhkan untuk menemukan kunci dan S-Box yang tepat sangatlah lama sehingga hampir tidak mungkin untuk dilakukan serangan *brute force* pada algoritma RF1.

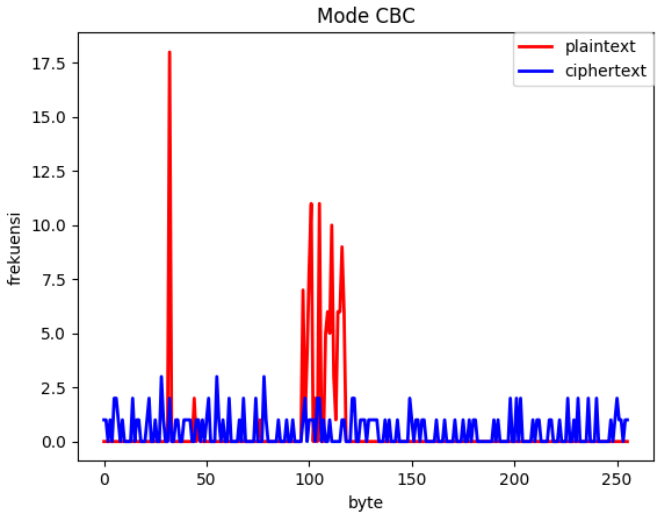
B. Analisis Frekuensi

Analisis frekuensi merupakan jenis serangan yang umum dilakukan pada dunia kriptografi. Pada jenis serangan ini,

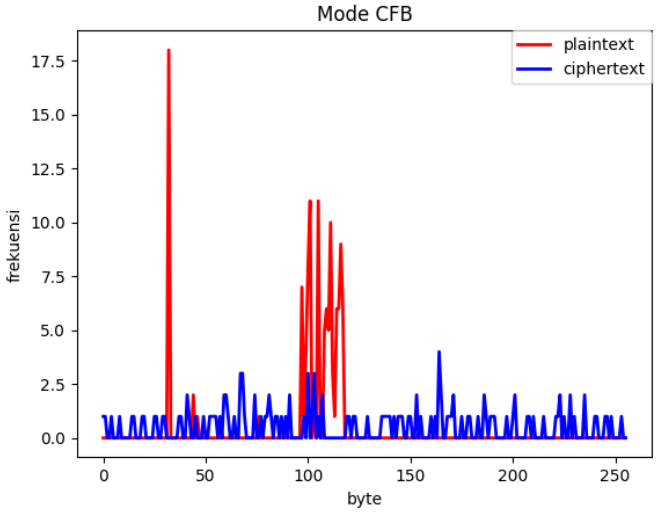
kriptanalis akan menganalisis frekuensi kemunculan setiap karakter pada cipherteks dan mencoba menebak plainteks berdasarkan frekuensi kemunculan huruf pada suatu bahasa. Misalkan pada Bahasa Inggris, huruf 'E', 'T', 'A', 'O' dan 'I' adalah huruf dengan frekuensi kemunculan terbesar. Huruf 'Z', 'Q', dan 'X' adalah huruf dengan frekuensi kemunculan terendah. Huruf-huruf tersebut kemudian dibandingkan dengan karakter yang paling sering muncul atau paling jarang muncul di cipherteks. Pada Gambar 9 - 13, ditunjukkan grafik perbandingan frekuensi kemunculan *byte* (karakter) pada plainteks dengan cipherteks yang dienkripsi menggunakan algoritma RF1.



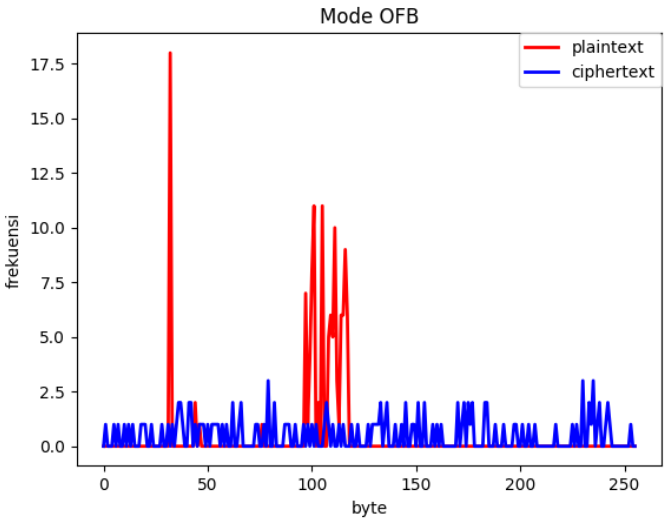
Gambar 9. Perbandingan frekuensi kemunculan *byte* pada mode ECB



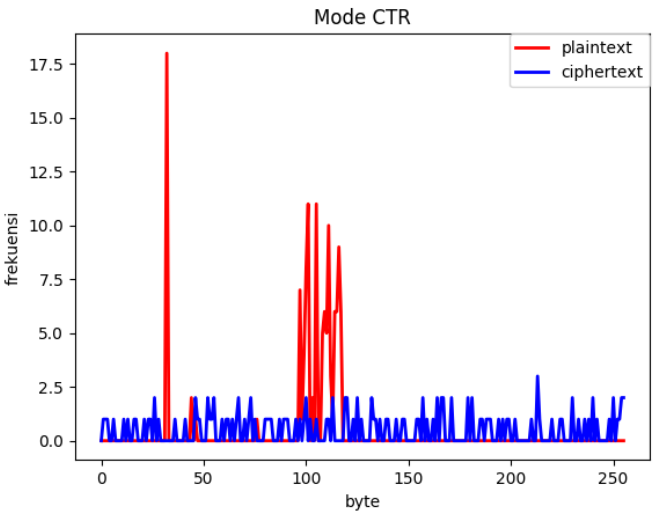
Gambar 10. Perbandingan frekuensi kemunculan *byte* pada mode CBC



Gambar 11. Perbandingan frekuensi kemunculan *byte* pada mode CFB



Gambar 12. Perbandingan frekuensi kemunculan *byte* pada mode OFB



Gambar 13. Perbandingan frekuensi kemunculan *byte* pada mode Counter

Sumbu X mereprensentasikan *byte* dari karakter ASCII pada plainteks maupun cipherteks. Sumbu Y merepresentasikan frekuensi kemunculan dari sebuah karakter. Pada plainteks, frekuensi kemunculan karakter tersebar secara tidak merata. Hal ini wajar dikarenakan karakter yang umum digunakan pada plainteks adalah huruf tertentu dan beberapa tanda baca saja. Pada cipherteks, dapat dilihat bahwa persebaran karakter cukup merata pada semua mode yang digunakan. Hal ini menunjukkan bahwa karakter pada cipherteks memiliki frekuensi kemunculan yang tidak serupa dengan karakter pada plainteks.

Berdasarkan analisis tersebut, dapat disimpulkan bahwa algoritma RF1 kuat terhadap serangan analisis frekuensi. Serangan analisis frekuensi bergantung pada keterhubungan distribusi frekuensi kemunculan karakter pada cipherteks dan plainteks. Pada algoritma RF1, karakter pada cipherteks memiliki distribusi frekuensi kemunculan yang merata dan tidak memiliki keterhubungan dengan frekuensi kemunculan karakter di plaintext.

VI. KESIMPULAN DAN SARAN

Algoritma RF1 yang dikembangkan merupakan algoritma Block Cipher yang sederhana namun kuat terhadap serangan. Algoritma ini juga memperhatikan prinsip *confusion* dan *diffusion* dalam proses pembuatannya. Berdasarkan berbagai analisis yang dilakukan, keamanan dari algoritma ini cukup kuat dan sulit diserang menggunakan *brute force attack* maupun analisis frekuensi. Kedepannya, semoga algoritma ini dapat memberikan inspirasi bagi kriptografer dalam merancang algoritma baru dan berkontribusi di dunia kriptografi khususnya Block Cipher.

DAFTAR REFERENSI

[1] Chandrasekaran, J. et al. 2011. A Chaos Based Approach for Improving Non Linearity in the S-Box Design of Symmetric Key Cryptosystems. In Meghanathan, N. et al.

[2] Munir, Rinaldi. 2015. Slide Kuliah IF4020 Kriptografi: Algoritma Kriptografi Modern.

[3] Munir, Rinaldi. 2015. Slide Kuliah IF4020 Kriptografi: Pengantar Kriptografi.

[4] Trapper, W & Washington, L. C. 2005. Diffusion and Confusion. Introduction to Cryptography and Coding Theory.

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, 16 Maret 2018

Micky Yudi Utama - 13514011

Varian Caesar - 13514041

Lampiran

1. **S-Box**

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16