1399 cipher

Anwar Ramadha / 13514013

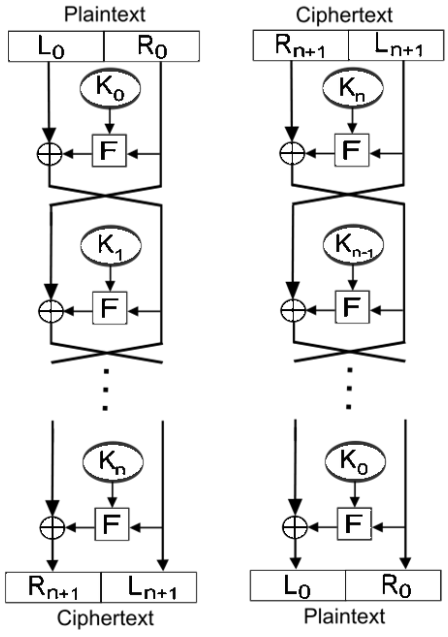
Drestanto M. D. / 13514099

Abstract

Enkripsi pesan banyak digunakan untuk menjaga keamanan komunikasi. Sudah banyak algoritma enkripsi yang telah dikembangkan. Algoritma tersebut harus dirancang serumit mungkin agar sukar untuk dipecahkan. Block Cipher merupakan algoritma yang beroperasi pada sekumpulan bit dengan panjang sama yang disebut block. Algoritma enkripsi yang akan kembangkan merupakan algoritma block cipher yang menawarkan solusi dengan keamanan yang berlapis namun tetap sederhana. 1399 chiper menerapkan jaringan feistel pada proses enkripsi dan dekripsinya. Algortima ini memiliki S-Box berukuran 16 x 16 seperti algoritma AES yang dibangkitkan dinamik secara pseudorandom berdasarkan seed tertentu. Algoritma ini juga memiliki jumlah perputaran (round) yang berbeda bergantung pada kunci eksternal.

Metodologi

Struktur yang digunakan dalam block cipher ini adalah struktur Feistel. Sehingga enkripsi dan dekripsi akan memiliki struktur yang sama. Feistel yang kami gunakan membagi plainteks sama rata menjadi plainteks kiri dan plainteks kanan.



Plainteks akan dibagi menjadi L (plainteks kiri) dan R (plainteks kanan). Plainteks kanan dimasukkan ke dalam sebuah fungsi F (akan dijelaskan pada halaman berikutnya) dan menjadi plainteks kiri untuk iterasi berikutnya. Plainteks kiri di-XOR-kan dengan plainteks kanan yang sudah dimasukkan ke dalam fungsi F dan menjadi plainteks kanan untuk iterasi berikutnya.

Iterasi dilakukan sebanyak n kali. Jumlah n ditentukan dengan cara berikut.

Key awal yang dimiliki berbentuk string. Setiap character dari string akan direpresentasikan dalam angka-angka dan dijumlahkan. Hasil penjumlahan akan di-mod dengan panjang string dan ditambah 1 untuk mendapatkan nilai n.

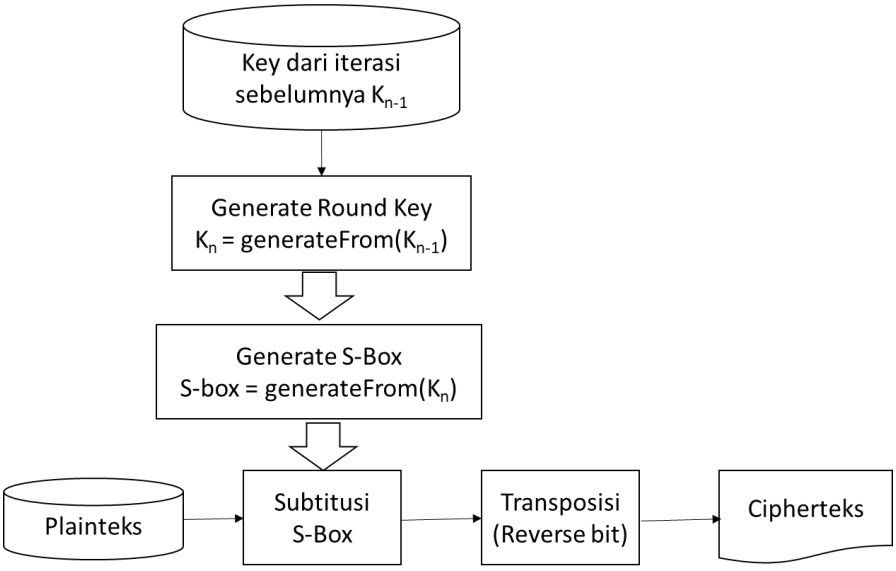
Contoh :

Key = “tes” -> panjang kunci = 3

n = (‘t’ + ‘e’ + ‘s’) mod 3 + 1 = 116+101+115 mod 3 + 1 = 3

Maka iterasi dilakukan sebanyak 3 kali.

Fungsi F pada struktur Feistel di-detilkan dengan gambar di bawah ini



Langkah-langkah yang dilakukan pada metode ini:

1. Generate round key
   Key pertama menjadi seed untuk pseudo random generator. Kemudian melakukan 3 kali generate random 0-255 sehingga didapatkan 3 buah karakter. 3 buah karakter ini akan digunakan untuk menjadi round key yang akan mengenerate S-Box sendiri.
   Untuk putaran Feistel berikutnya, key yang digunakan untuk mengenerate round key adalah menggunakan round key sebelumnya.
   Contoh : awalnya key adalah
2. Generate S-Box
   Ukuran S-Box adalah 16 x 16 heksadesimal.
   S-Box degenerate dengan menggunakan round key sebagai seed dari pseudo random generator. Dilakukan generate bilangan random 0-15.
   Hasil pertama pseudo random dijadikan nomor baris pertama, hasil kedua menjadi nomor kolom pertama. Sehingga baris dan kolom pada S-Box tersebut bernilai “00” heksadesimal. Setelah itu, random kedua dan ketiga menjadi baris dan kolom untuk “01” heksa, begitu seterusnya. Jika didapat hasil random dengan nomor baris dan nomor kolom yang sudah terisi, maka sisa S-Box yang belum terisi akan diisi secara sekuensial.
3. Subtitusi S-Box
   Untuk setiap blok, dikelompokkan menjadi pasangan heksadesimal. Pasangan tersebut menjadi baris dan kolom pada S-Box, kemudian dilakukan subtitusi seperti biasa
4. Transposisi
   Untuk setiap blok-nya, dilakukan reverse bit-bit nya. Bit paling depan dalam blok menjadi bit paling belakang, bit paling belakang menjadi bit terdepan. Bit kedua menjadi bit kedua terakhir, dan seterusnya (misalnya blok 8 bit berisi 10011011 maka di-reverse menjadi 11011001) untuk mendapatkan cipherteks.