Algoritma *Block Cipher*: FIERKES Cipher

Hafizh Dary Faridhan Hudoyo
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung
40132, Indonesia
13514072@std.stei.itb.ac.id

Dandu Satyanuraga
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung
40132, Indonesia
13515601@std.stei.itb.ac.id

***Abstrak*— Block cipher merupakan sebuah algoritma kriptografi modern dengan cara membagi data ke blok-blok terlebih dahulu lalu dienkripsi. Fierkes cipher yang dikembangkan dalam paper ini adalah block cipher dengan menggunakan dasar modifikasi kotak S. Dengan menggunakan transposisi dan subtitusi dalam cipher dan subtitusi pada key yang digunakan maka Fierkes cipher akan menghasilkan ciphertext yang kuat. Jaringan Feistel juga digunakan agar enkripsi dan dekripsi memiliki algoritma yang sama..**

***Kata Kunci*—S-Box; Feistel; Transposisi; Substitusi; XOR; Blok; Cipher.**

I. PENDAHULUAN

Kriptografi (*cryptography*) adalah ilmu yang mempelajari teknik-teknik secara matematis untuk mengenkripsi sebuah informasi penting, seperti dokumen negara. Kekuatan dari sebuah teknik enkripsi ditinjau dari aspek keamanan informasi, seperti integritas data, kerahasiaan data, atau otentikasi data.

Kriptografi telah diciptakan sekitar abad 19 Sebelum Masehi pada peradaban Mesir yang diukir pada batu. Ilmu kriptografi semakin berkembang dan menjadi semakin rumit seiring perkembangan zaman. Pada Perang Dunia I dikembangkan mesin *rotor cipher* dan pada Perang Dunia II telah diciptakan komputer untuk mengirimkan pesan kepada golongan sepihak tanpa dapat dibaca oleh pihak musuh. Kriptografi modern adalah kriptografi yang diciptakan pada era komputer dengan memanfaatkan teori matematis dan aplikasi komputer dengan pengoperasian kepada pesan dalam bentuk bit atau biner.

Seiring perkembangan dunia maya dan penyebaran informasi di dalamnya, keamanan informasi adalah salah satu aspek penting yang perlu diperhatikan oleh pemilik sebuah informasi. Informasi yang telah tersedia di jaringan internet dapat ditransmisikan dari suatu tempat ke banyak tempat sehingga rentan terhadap penyadapan.

Salah saru cara untuk melakukan enkripsi terhadap bit-bit pesan adalah dengan menggunakan metode *Block Cipher*. *Block Cipher* adalah teknik kriptografi yang membagi bit-bit plainteks menjadi blok-blok bit dengan panjang yang sama. *Block Cipher* digunakan untuk meningkatkan keamanan pesan dengan menggabungkan perhitungan atau operasi sederhana seperti XOR atau substitusi yang dilakukan dalam beberapa putaran. Pada makalah ini dibahas rancangan algoritma baru yang dibuat oleh penulis dengan memanfaatkan *Block Cipher* yang bernama FIERKES Cipher. Algoritma ini memanfaatkan berbagai operasi seperti substitusi, XOR, dan transposisi. Dengan digunakannya operasi-operasi ini, diharapkan algoritma ini dapat berkontribusi dalam ilmu kriptografi sebagai salah satu algoritma yang memiliki properti *confusion* dan *diffusion*.

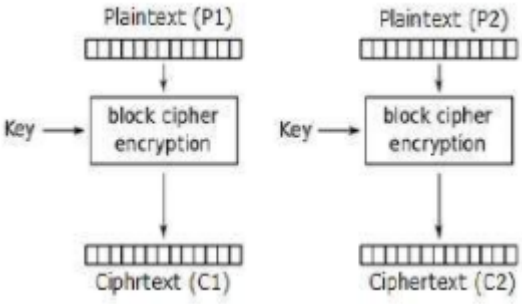
II. DASAR TEORI

A. Block Cipher

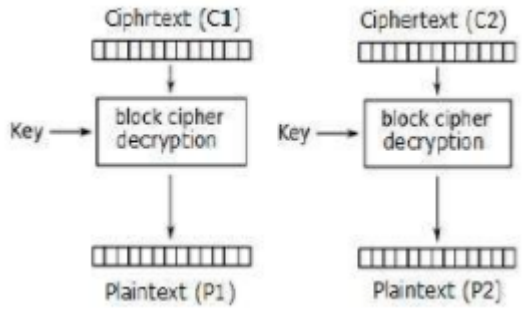
Block Cipher adalah algoritma yang beroperasi pada sekumpulan bit dengan panjang yang sama, yang disebut dengan *block*. Langkah awalnya adalah membagi plainteks menjadi blok-blok yang kemudian dienkripsi dengan kunci yang sama panjangnya dengan blok sehingga menghasilkan cipherteks yang sama panjang dengan teks aslinya. Pada *Block Cipher* dikenal lima mode operasi, yaitu *Electronic Code Book* (ECB), *Cipher Block Chaining* (CBC), *Cipher Feedback* (CFB), *Output Feedback* (OFB), dan *Counter*.

1. *Electronic Code Book* (ECB)

   Electronic Code Book (ECB) adalah teknik enkripsi sederhana di mana masing-masing blok dienkrispsi secara terpisah. Skema enkripsi dan dekripsi ECB digambarkan pada gambar di bawah ini.



Gambar 1: Skema enkripsi ECB

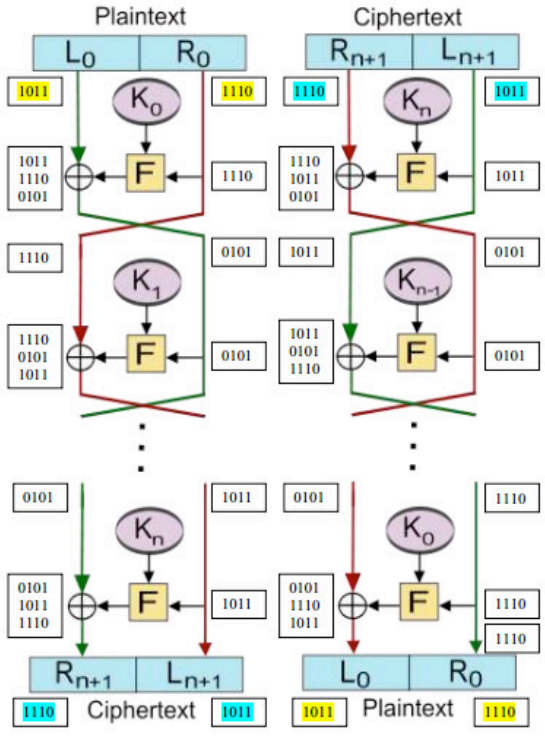


Gambar 2: Skema dekripsi ECB

B. *Jaringan Feistel*

Jaringan Feistel adalah salah satu struktur umum yang digunakan di dalam kriptografi modern. Sistem yang digunakan pada jaringan Feistel adalah dengan membagi plainteks menjadi dua buah bagian, yaitu kiri dan kanan. Masing-masing bagian diperlakukan secara berbeda. Bagian kiri yang baru dihasilkan langsung dari bagian kanan. Sementara bagian kanan dihasilkan dari operasi fungsi tertentu yang menggunakan kunci dan dioperasikan kembali dengan bagian kiri yang sebelumnya.

Proses ini dapat dilakukan berkali-kali untuk menghasilkan cipherteks yang lebih acak dari sebelumnya. Cara untuk mendekripsi jaringan Feistel mirip dengan cara mengenkripsinya. Gambar di bawah menunjukkan proses jaringan Feistel yang lebih detail.



Sumber: https://en.wikipedia.org/wiki/Feistel_cipher

Gambar 7: Proses enkripsi (kiri) dan dekripsi (kanan) pada Jaringan Feistel

C. *Prinsip* diffusion *dan* confusion *dari Shannon*

Confusion dan *diffusion* merupakan dua properti dalam kriptografi yang menjamin keamanan cipherteks dengan membuat serangan secara statistik menjadi lebih rumit. Prinsip ini diperkenalkan oleh Claude Shannon pada tahun 1949 dalam makalahnya yang berjudul *Communication Theory of Secrecy Systems*.

Prinsip *confusion* bekerja dengan menyembunyikan hubungan yang ada antara plainteks, cipherteks, dan kunci. Prinsip *confusion* dapat direalisasikan dengan menggunakan algoritma substitusi yang kompleks.

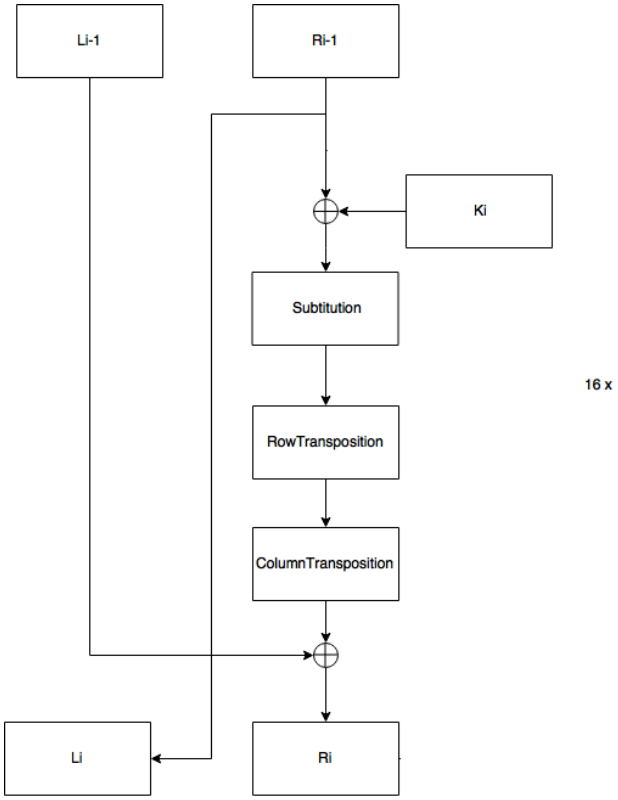
Sementara itu, prinsip *diffusion* menyebarkan pengaruh 1 bit plainteks atau kunci ke sebanyak mungkin cipherteks. Misalnya jika perubahan dilakukan pada satu bit plainteks, maka secara statistik sebagian besar bit pada cipherteks juga akan berubah, begitu pula sebaliknya.

Kedua prinsip ini merupakan panduan dalam merancang berbagai algoritma kriptografi dan juga menjadi konsep yang penting dalam merancang fungsi *hash* dan *pseudorandom generator*.

III. Rancangan Algoritma

A. *Jaringan Feistel*

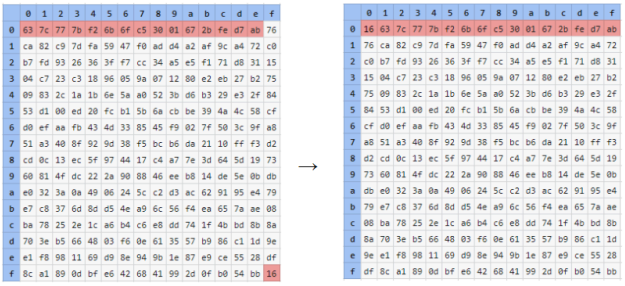
Algoritma baru yang penulis rancang bernama FIERKES Cipher. Rancangan *block cipher* yang penulis tawarkan adalah menggunakan Jaringan Feistel yang didalamnya terdapat fungsi-fungsi berikut ini.



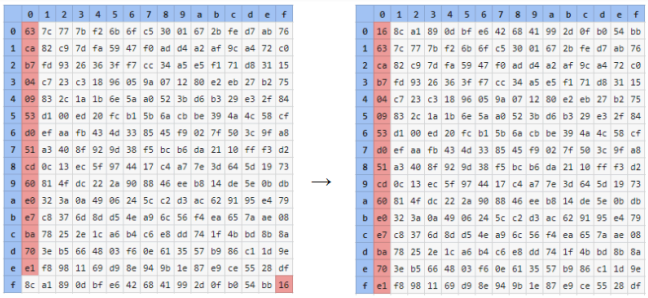
Gambar 8: Jaringan Feistel FIERKES Cipher

Dari gambar di atas, Li dihasilkan langsung dari Ri-1. Ri dihasilkan dari lima tahap enkripsi, yaitu menerapkan operasi XOR pada Ri-1 dengan sebuah kunci yang dihasilkan dari sebuah S-Box, substitusi hasil XOR dengan S-Box yang lain, transposisi baris dari hasil substitusi, transposisi kolom dari hasil transposisi baris, dan XOR dengan Li-1. Tahap-tahap ini dilakukan sampai 16 kali pengulangan.

Setiap pengulangan, S-Box untuk kunci digeser secara horizontal dan S-Box untuk substitusi digeser secara vertikal. Gambar di bawah menjelaskan pergeseran horizontal dan vertikal tersebut secara detail.



Gambar 9: Pergeseran S-Box secara horizontal (ke kanan)



Gambar 10: Pergeseran S-Box secara vertikal (ke bawah)

Berikut adalah penjelasan dari tahap-tahap pada algoritma yang penulis rancang.

1. Tahap Operasi XOR dengan kunci

   Sebelum masuk tahap ini, bagian kanan plainteks dibagi dalam beberapa matriks 8x8 yang berisi bit, atau bisa dikatakan sebagai *array* yang berisi delapan karakter (1 karakter = 8 bit). Kunci yang dipakai berasal dari delapan elemen pertama pada S-Box. Lalu, matriks plainteks dioperasikan dengan operator XOR dengan kunci yang telah dihasilkan.

   Setelah plainteks dioperasikan, S-Box untuk kunci digeser secara horizontal ke kanan sejauh satu elemen.

2. Substitusi dengan S-Box

   Teks yang dihasilkan dari proses sebelumnya disubstitusikan dengan S-Box yang baru (bukan S-Box kunci). Setiap karakter disubstitusikan dengan cara melihat indeks matriks pada S-Box yang diwakilkan oleh karakter plainteks yang sudah terbagi dalam dua buah bilangan 4 bit (0 sampai F). Setelah dicari indeksnya, setiap karakter pada teks lalu disubstitusikan dengan isi S-Box pada indeks yang sudah didapat.

3. Transposisi Baris

   Setelah itu, plainteks dibagi menjadi beberapa matriks 4x4. Jika tidak memungkinkan (karakter sisa pembagian berjumlah kurang dari 16), beberapa karakter terakhir tidak dimasukkan ke dalam matriks. Setelah itu, baris pada matriks ditukar satu sama lain sehingga menjadi susunan teks yang berbeda. Pada contoh di bawah, data pada baris 1 dan 2 ditukar dengan data pada baris 3 dan 4.

6a	cb	be	39			c4	a7	7e	3d
45	f9	02	7f	--	\	bc	b6	da	21
bc	b6	da	21	--	/	6a	cb	be	39
c4	a7	7e	3d			45	f9	02	7f

Gambar 11: Transposisi baris

4. Transposisi Kolom

   Proses ini sama dengan proses sebelumnya, namun pertukaran dilakukan berdasarkan kolom matriks 4x4.

6a	cb	be	39			be	cb	39	6a
45	f9	02	7f	--	\	02	f9	7f	45
bc	b6	da	21	--	/	da	b6	21	bc
c4	a7	7e	3d			7e	a7	3d	c4

Gambar 12: Transposisi kolom

5. Tahap Operasi XOR dengan *left*

   Setelah itu, hasil teks dari proses-proses sebelumnya dioperasikan dengan bagian *left* menggunakan operasi XOR seperti pada Jaringan Feistel yang umum.

IV. HASIL

Pengujian yang dipakai berupa teks sepanjang 56 karakter, yaitu "This adalah tes untuk mengenkripsi dengan FierkesCipher." Setelah diterapkan algoritma FIERKES Cipher, cipherteks yang dihasilkan adalah:

Plainteks:

"This adalah tes untuk mengenkripsi dengan FierkesCipher."

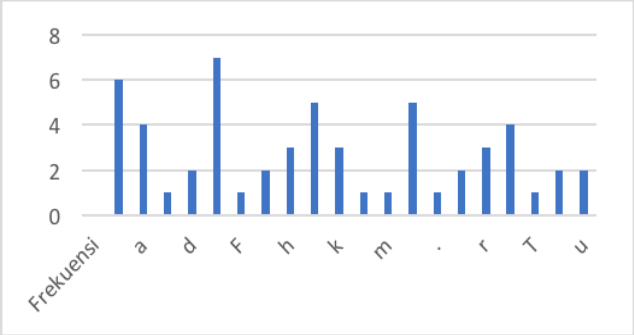
Cipherteks:

```
¾~)7; ¾ ¾¬7°r Å÷")±ZS ä"ÌK0W Gç]þ'XbR¯O¦ øn\ßÞf >~½mÒ£
```

Hasil dekripsi:

```
This adalah tes untuk mengenkripsi dengan FierkesCiphe
```

Histogram:



Gambar 13: Histogram Plaintext

Gambar 14: Histogram Chipertext

Waktu yang dibutuhkan: 4 ms. Dengan menggunakan bahasa pembrograman JAVA.

V. Analisis

Jika bit pada cipherteks diubah sebanyak satu bit paling belakang, hasil yang dikeluarkan dari proses dekripsi adalah:

```
This adalah tes untuk menge~kripsi dengan FierkesCipher□
```

Terlihat bahwa karakter yang berubah dari teks aslinya hanya dua buah.

Jika bit pada kunci diubah sebanyak satu bit, hasil yang dikeluarkan dari proses dekripsi adalah

```
□Ê□?Â-¢l³l'Â;□È æ«
□E□]¼□□□9}óép8j!¢üçö□x□9jë³□❸ÌX(□□gÄ□□
```

Terlihat bahwa teks berubah seutuhnya sehingga tidak memiliki makna seperti plainteks aslinya.

Pada histogram terlihat bahwa plainteks dan ciphertext sangat berbeda histogramnya dan cipherteks memiliki histogram yang relatif datar. Hal tersebut membuat Fierkes cipher tidak mampu dipecahkan melalui analisi frekuensi.

Untuk pengujian menggunakan bruteforce, menggunakan algoritma bruteforce untuk mencari hasil dekripsi dari teks yang di enkripsi menghasilkan:

Brute-force attack cracking time estimate

Machine	Time
Standard Desktop PC	About 745 septentrigintillion years
Fast Desktop PC	About 186 septentrigintillion years
GPU	About 75 septentrigintillion years
Fast GPU	About 37 septentrigintillion years
Parallel GPUs	About 4 septentrigintillion years
Medium size botnet	About 745 quinquatrigintillion years

Gambar 14: Hasil algoritma bruteforce.
Sumber: http://password-checker.online-domain-tools.com/

dengan data yang didapat di atas Fierkes cipher mampu bertahan sangat baik melawan algoritma *bruteforce*.

VI. Kesimpulan dan Saran

Setelah menerapkan algoritma ini, dapat disimpulkan bahwa algoritma ini belum menerapkan prinsip *diffusion* sepenuhnya pada perubahan cipherteks yang jika didekripsi menghasilkan teks yang sebagian besar sama dengan plainteks. Selain itu, plainteks *input* masih harus memiliki panjang yang berjumlah genap, karena jika ganjil pembagian kiri dan kanan tidak konsisten.

Ke depannya, penulis ingin mengembangkan lebih dalam lagi agar pengubahan bit pada cipherteks memengaruhi seluruh karakter pada cipherteks dan menghasilkan plainteks yang sangat acak.

VII. Ucapan Terima Kasih

Pertama-tama penulis mengucapkan terima kasih kepada Allah SWT karena telah memberikan rahmat dan berkat sehingga penulis dapat menyelesaikan makalah ini. Terima kasih juga kepada dosen pengajar Mata Kuliah IF4020 Kriptografi yaitu Dr. Ir. Rinaldi Munir, M.T. atas segala bimbingannya selama perkuliahan di semester 8 ini. Terima kasih juga kepada teman-teman mahasiswa Teknik Informatika yang selalu mendukung dalam kuliah maupun luar kuliah.

VIII. Referensi

[1] https://www.tutorialspoint.com/cryptography/block_cipher_modes_of_operation.htm diakses tanggal 16 Maret 2018 pukul 11:00 WIB

[2] https://www.linkedin.com/pulse/feistel-cipher-encryption-decryption-process-shudarshon-chaki/ diakses tanggal 16 Maret 2018 pukul 11:23 WIB