Weed Block Cipher

Arung Agamani B.P.[1], **M. Zunan Alfikri**[2].

[1,2] Program Studi Teknik Informatika, Sekolah Teknik Elektro dan Informatika (STEI), Institut Teknologi Bandung (ITB), Jalan Ganesha 10, Bandung 40132
E-mail: [1] 13518005@std.stei.itb.ac.id, [2] 13518019@std.stei.itb.ac.id

Abstrak. Algoritma Weed Block Cipher merupakan suatu block cipher yang memiliki keunggulan dalam pembangkitan S-box yang merupakan key-dependent, namun tetap menjaga sifat nonlinearitas dari S-box yang dihasilkan sehingga tetap sulit untuk diterka melalui analisis frekuensi maupun serangan brute-force. Weed Block Cipher menggunakan jaringan Feistel dengan memanfaatkan operasi substitusi baris, substitusi kolom, substitusi S-box, dan transposisi P-box sebagai operasi yang menyusun fungsi putarannya. Banyak operasi yang ada akan memberi hasil yang semakin sulit untuk diterka, dan merupakan algoritma block cipher yang memenuhi Shannon's Diffusion and Confusion principle, sehingga termasuk algoritma block cipher yang kuat.

Keywords: Block Cipher, Dekripsi, Enkripsi, Jaringan Feistel, Kriptografi, Substitusi, Transposisi

1. Pendahuluan

Pada era industri 4.0, teknologi berkembang sangat pesat. Hal tersebut menyebabkan meningkatnya penggunaan media digital sebagai sarana berkomunikasi pada era ini. Komunikasi menggunakan media digital khususnya internet memiliki beberapa kelemahan diantaranya pesan yang dikirimkan melalui internet bisa saja dibaca atau diubah oleh pihak ketiga karena internet digunakan secara publik dan dapat diakses oleh semua orang.

Untuk mencegah pesan dapat dibaca, diubah, atau dimanfaatkan orang lain, kita dapat menggunakan kriptografi. Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan, integritas data, otentikasi entitas, dan otentikasi asal usul data (Menezes, 1996). Kriptografi dapat dimanfaatkan untuk mengenkripsi pesan sehingga pihak ketiga yang membaca pesan tersebut tidak mengetahui maknanya.

Kriptografi sudah ada sejak zaman dahulu dan terus mengalami perkembangan sampai saat ini. Pada awal kemunculan kriptografi, metode-metode yang digunakan masih relatif sederhana seperti substitusi dan rotasi. Kriptografi pada zaman dahulu sering disebut dengan kriptografi klasik. Kriptografi klasik ini cenderung mudah dipecahkan dengan berbagai teknik kriptanalisis (memecahkan pesan yang telah di enkripsi). Semakin berkembangnya kebutuhan manusia terhadap keamanan informasi, maka metode-metode yang digunakan juga semakin berkembang. Jenis kriptografi yang banyak diterapkan saat ini disebut kriptografi modern.

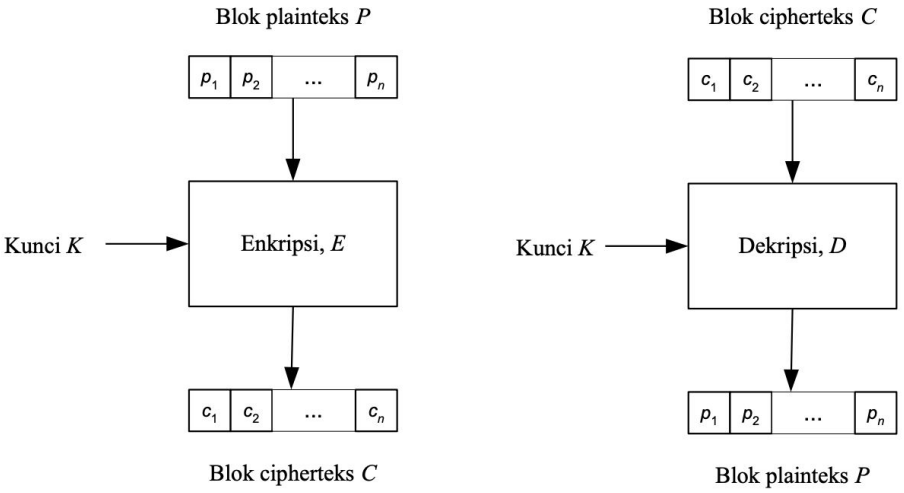
Aplikasi dari kriptografi modern ini sangat banyak, misalnya keamanan komunikasi perbankan, *broadcast* sinyal siaran, pengiriman SMS, dan lain sebagainya. Kriptografi modern terdiri atas dua kategori yaitu *Stream Cipher* dan *Block Cipher*. Stream Cipher beroperasi pada bit tunggal sedangkan Block Cipher beroperasi pada blok bit. Contoh Stream Cipher yaitu algoritma A2, SEAL, WAKE dan contoh Block Cipher yaitu Algoritma DES, AES, RC5.

Pada makalah ini, penulis akan membahas sebuah algoritma Block Cipher bernama Weed Block Cipher. Weed Block Cipher dirancang berdasarkan prinsip prinsip perancangan block cipher yaitu prinsip Confusion dan Diffusion dari Shannon, Cipher berulang, Jaringan Feistel, serta S-Box.

2. Dasar Teori

2.1. Block Cipher

Salah satu algoritma kriptografi modern dengan kunci simetris adalah Block Cipher. Plaintext yang akan dienkripsi akan dibagi sesuai ukuran blok yang digunakan algoritma. Kemudian algoritma akan dilakukan terhadap blok-blok *plaintext* dengan menggunakan kunci enkripsi. Panjang kunci enkripsi umumnya sesuai dengan panjang blok yang digunakan oleh algoritma. Panjang ciphertext yang dihasilkan sama dengan panjang plaintext yang akan dienkripsi. Apabila panjang plaintext bukan kelipatan dari ukuran blok, maka akan ditambahkan *padding* sampai panjang pesan menjadi kelipatan panjang blok. *Padding* yang digunakan berupa bit 0 semua atau bit 1 semua. Berikut ini skema enkripsi dan deskripsi dari *Block Cipher*.



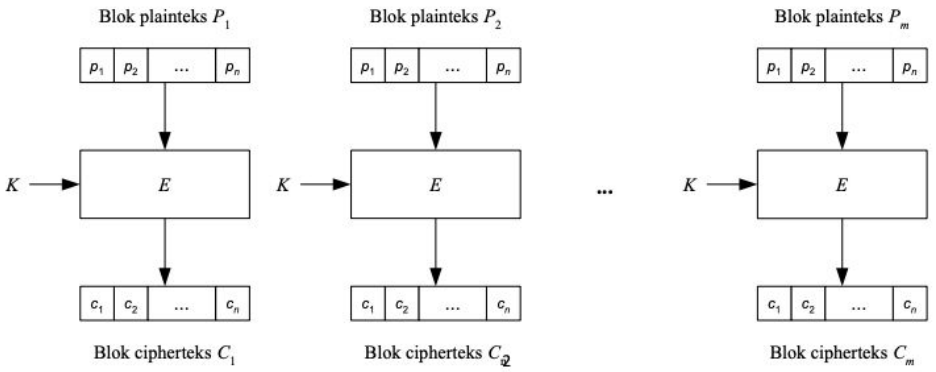
Gambar 1. Skema enkripsi dan dekripsi pada *Block Cipher*

Block Cipher memiliki lima mode operasi. hal ini berkaitan dengan cara blok dioperasikan pada proses enkripsi dan dekripsi. Lima mode tersebut yaitu :

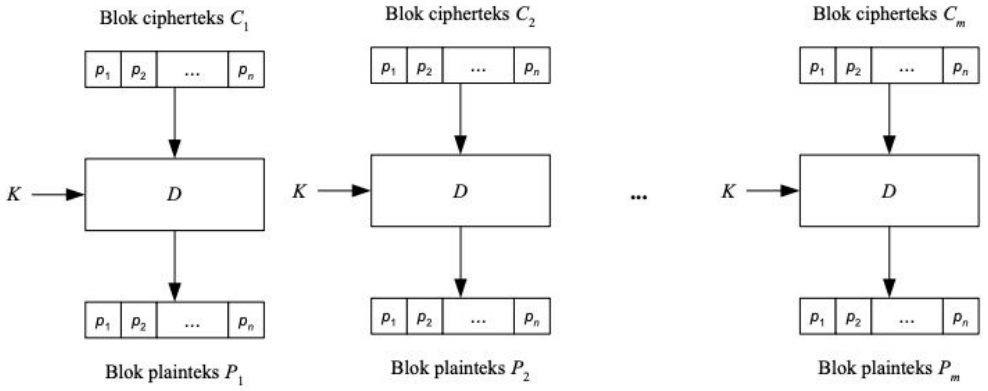
2.1.1. *Electronic Code Book* (ECB)

Pada mode ini setiap blok-blok pesan atau plainteks dienkripsi secara individual dan independen menjadi blok-blok cipherteks. Blok plainteks yang sama selalu dienkripsi menjadi blok cipherteks yang sama. Keuntungan dari mode ECB ini adalah blok-blok plainteks dienkripsi secara individual dan independen sehingga kita tidak perlu mengenkripsi file secara linear. Kekurangan dari mode ECB ini adalah mudah diserang secara statistik karena

blok-blok plainteks yang sama menghasilkan blok-blok cipherteks yang sama. Berikut ini skema enkripsi dan dekripsi pada mode ECB.



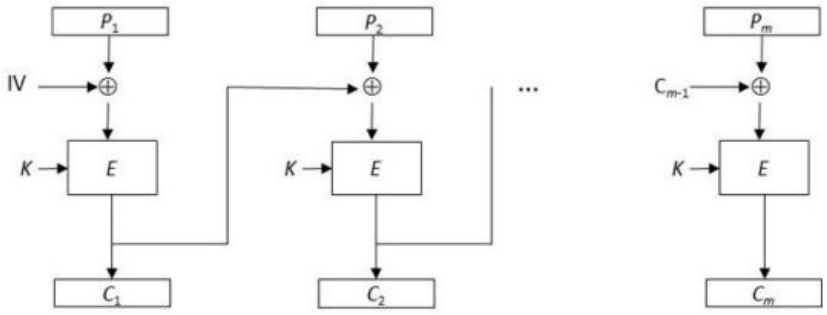
Gambar 2. Enkripsi pada mode ECB



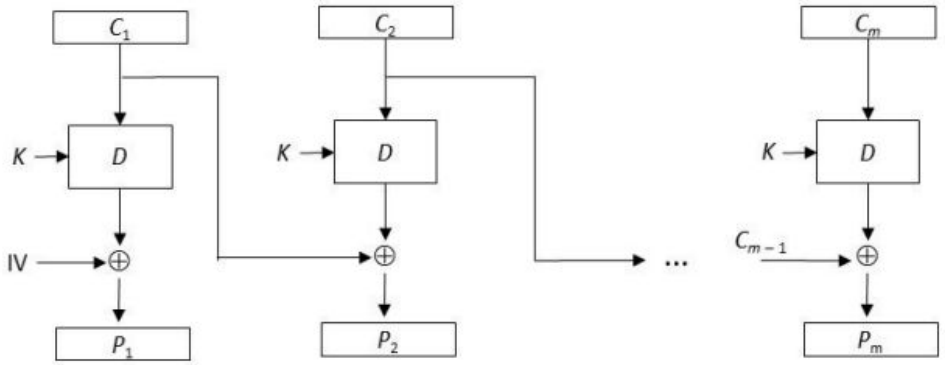
Gambar 3. Dekripsi pada mode ECB

2.1.2. *Cipher Block Chaining* (CBC)

Mode CBC merupakan mode block cipher yang membuat ketergantungan antar blok, dimana setiap blok cipherteks bergantung pada blok plainteksnya dan seluruh blok plainteks sebelumnya. Kelebihan dari mode CBC ini adalah blok plainteks yang sama menghasilkan blok cipherteks yang tidak sama, sehingga kriptanalisis terhadap mode CBC ini akan lebih sulit dibandingkan mode ECB. Kekurangan mode CBC ini adalah kesalahan enkripsi pada sebuah blok plainteks akan menyebabkan kesalahan beruntun pada blok cipherteksnya dan seluruh blok cipherteks berikutnya. Berikut ini skema enkripsi dan dekripsi pada mode CBC.



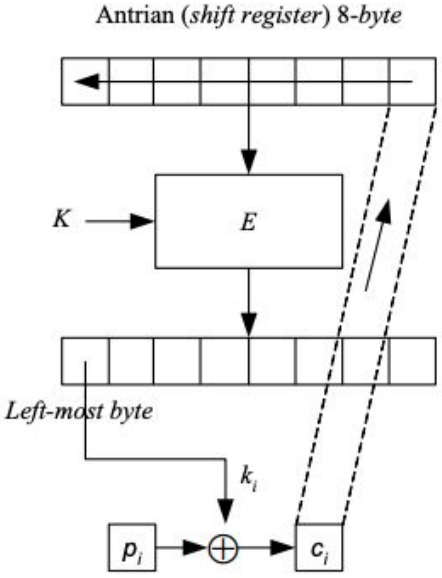
Gambar 4. Enkripsi pada mode CBC



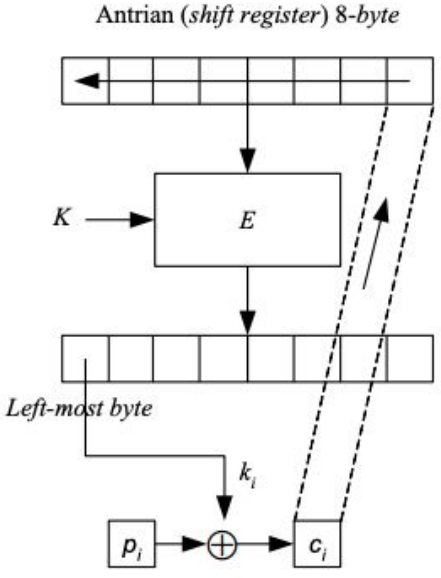
Gambar 5. Dekripsi pada mode CBC

2.1.3. *Cipher Feedback Block* (CFB)

Mode CFB merupakan peningkatan dari mode CBC yang mengatasi kekurangan pada CBC apabila diterapkan pada pengiriman data yang belum mencapai ukuran satu blok. Data pada mode CFB dienkripsi dalam unit yang lebih kecil daripada ukuran blok sepert 1 bit, 2 bit, 4 bit, dan 8 bit. Kekurangan mode CFB ini adalah kesalahan enkripsi pada sebuah blok plainteks akan berpengaruh pada blok cipherteksnya dan seluruh blok cipherteks berikutnya. Berikut ini skema enkripsi dan dekripsi pada mode CFB.



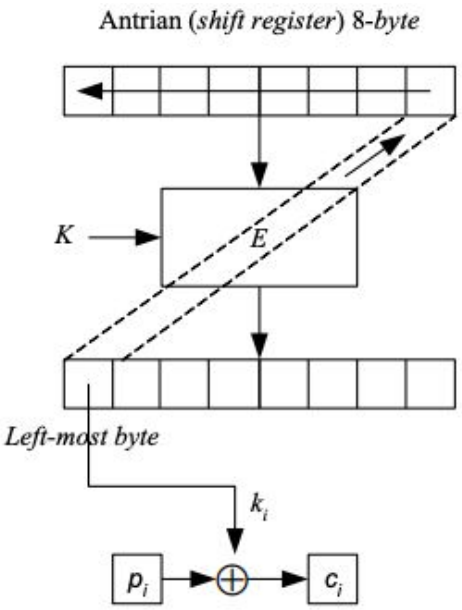
Gambar 6. Enkripsi pada mode CFB



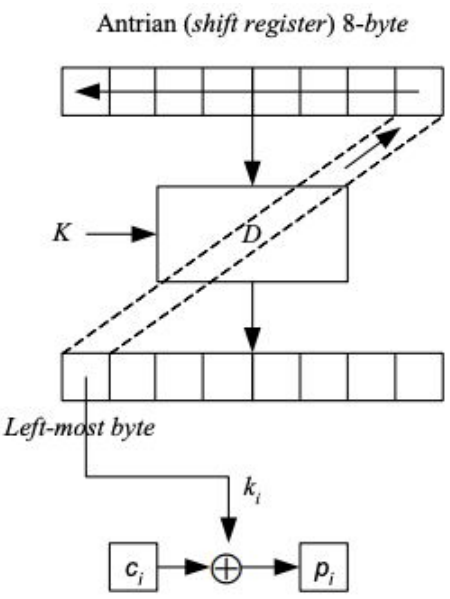
Gambar 7. Dekripsi pada mode CFB

2.1.4. *Output Feedback Block* (OFB)

Mode OFB merupakan pengembangan dari mode CFB dimana n-bit dari hasil enkripsi terhadap antrian disalin menjadi elemen posisi paling kanan pada antrian. Hal tersebut menyebabkan kesalahan enkripsi dan dekripsi pada sebuah blok plainteks hanya akan berpengaruh pada blok cipherteks yang berkoresponden saja. Berikut ini skema enkripsi dan dekripsi pada mode OFB.



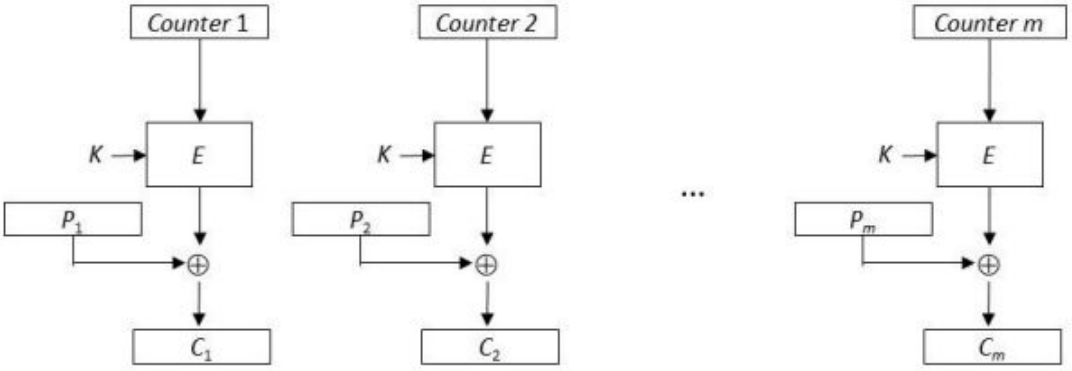
Gambar 8. Enkripsi pada mode OFB



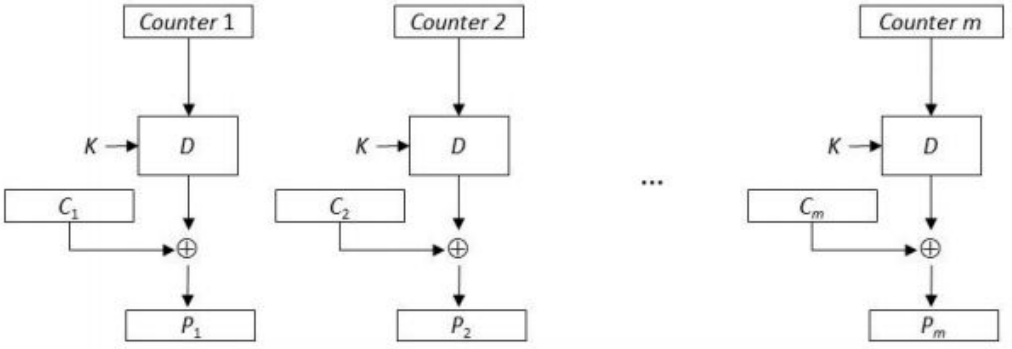
Gambar 9. Dekripsi pada mode OFB

2.1.5. Counter Mode

Coumter Mode merupakan Block Cipher yang tidak melakukan perantaian (*chaining*) seperti pada CBC. Counter adalah sebuah nilai berupa blok bit yang ukurannya sama dengan ukuran blok plainteks. Nilai counter harus berbeda dari setiap blok yang dienkripsi. Pada mulanya, untuk enkripsi blok pertama, counter diinisialisasi dengan sebuah nilai. Selanjutnya, untuk enkripsi blok-blok berikutnya counter dinaikkan nilainya satu. Berikut ini skema enkripsi dan dekripsi pada Counter Mode.



Gambar 10. Enkripsi pada Counter Mode



Gambar 11. Dekripsi pada Couter Mode

2.2. Prinsip Perancangan Block Cipher

Berikut ini empat prinsip yang sering digunakan dalam perancangan Block Cipher.

2.2.1. Prinsip *Confusion* dan *Diffusion* dari Shannon.

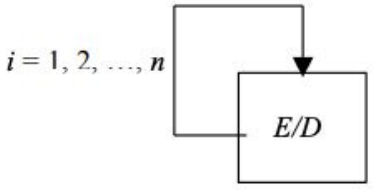
Banyak algoritma kriptografi klasik yang telah berhasil dipecahkan dengan metode statistik karena persebaran kemunculan frekuensi pada suatu bahasa telah diketahui. Pada tahun 1949, Claude Shannon memperkenalkan prinsip confusion dan diffusion pada makalahnya yang berjudul *Communication Theory of Secrecy Systems.*

Confusion merupakan prinsip yang menyembunyikan hubungan apapun yang ada antara plainteks, cipherteks, dan kunci. Prinsip *confusion* membuat kriptanalis frustasi untuk mencari pola-pola statistik yang muncul pada cipherteks. Salah satu contoh algoritma yang menerapkan prinsip *confusion* yaitu *One-Time Pad.*

Diffusion merupakan prinsip yang menyebarkan pengaruh satu bit plainteks atau kunci ke sebanyak mungkin cipherteks. Hal tersebut mengakibatkan pengubahan kecil pada satu atau dua plainteks dapat menghasilkan perubahan yang banyak pada cipher teks. Contoh algoritma yang menggunakan prinsip *diffusion* yaitu DBC dan CFB.

2.2.2. Cipher berulang (*iterated cipher*)

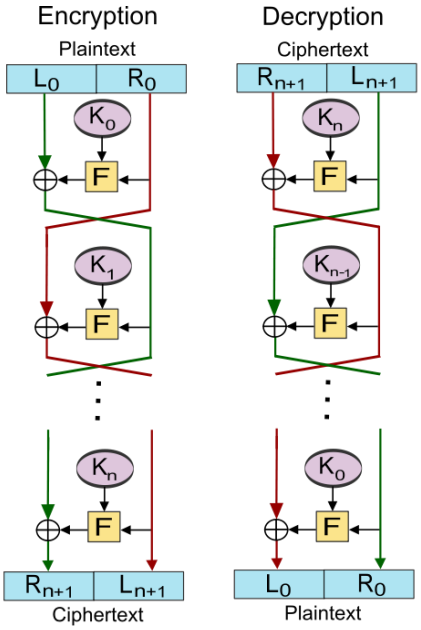
Cipher berulang yaitu fungsi transformasi sederhana yang mengubah plainteks menjadi cipherteks diulang sejumlah beberapa kali. Pada setiap perulangan, digunakan upa-kunci (*subkey*) atau kunci putaran *(round key*) yang dikombinasikan dengan plainteks. Berikut ini skema dari Cipher berulang.



Gambar 12. Cipher berulang

2.2.3. Jaringan Feistel *(Feistel Network*)

Jaringan Feistel merupakan sebuah struktur simetris block cipher dalam melakukan enkripsi dan dekripsi. Prinsip jaringan Feistel ini memungkinkan kita untuk tidak perlu membuat algoritma baru untuk dekripsi. Jaringan Feistel sendiri merupakan sebuah cipher berulang dimana fungsi internalnya disebut dengan fungsi putaran. Berikut ini skema dari jaringan feistel.



Gambar 13. Jaringan Feistel
Sumber : https://en.wikipedia.org/wiki/Feistel_cipher

2.2.4. Kotak-S (S-*box*)

Kotak-S atau Kotak-Substitusi dalam block cipher merupakan matriks substitusi yang digunakan untuk mengaburkan hubungan antara kunci dan cipher teks. Sehingga kotak-S ini menerapkan prinsip confusion dari Shannon. Dengan menggunakan kotak-S ini, sejumlah m bit masukan akan ditransformasi menjadi n bit keluaran. Kotak-S diimplementasikan sebagai sebuah tabel pencarian (lookup table).

3. Algoritma Weed Block Cipher

Weed Block Cipher adalah teknik enkripsi berbasis block cipher yang menggunakan jaringan Feistel. Jaringan Feistel mengandung round function yang berisi operasi row substitution, column substitution, S-box substitution, dan transposition di dalamnya, dengan kelebihan berupa pembangkitan S-box yang dilakukan berdasarkan key yang diberikan.

3.1. Pemrosesan Awal

Block Cipher ini bekerja dalam mode 128-bit, yang membuat pesan yang akan dienkripsi akan dipecah menjadi blok-blok sepanjang 128-bit. Apabila pesan tidak cukup panjang untuk membuat blok, maka akan dilakukan *padding* menggunakan karakter ‘0’. Desain dari Block Cipher ini menggunakan jaringan Feistel untuk melakukan enkripsi, dan dekripsi juga menggunakan jaringan Feistel yang dijalankan secara terbalik.

3.2. Desain Jaringan Feistel

Desain dari jaringan Feistel yang digunakan desain yang telah digambarkan pada poin 2.2.3. Pesan dengan panjang 128-bit akan dipecah menjadi dua buah pesan sepanjang 64-bit. Kedua buah pesan ini akan disebut sebagai L dan R. R akan dijadikan sebagai input untuk fungsi putaran dan hasil dari fungsi putaran ini akan dilakukan operasi XOR dengan L sehingga menghasilkan blok pesan yang akan disebut L_. Hasil dari satu putaran pada jaringan Feistel adalah hasil dari konkatenasi antara R dan L_. Proses ini akan dilakukan sebanyak 16 putaran untuk mode CBC dan Counter, dan khusus EBC akan dilakukan sebanyak 50 putaran.

3.3. Desain Fungsi Putaran

Adapun untuk desain fungsi putaran, fungsi ini akan menerima pesan dengan panjang 64-bit. Pesan ini akan dipecah menjadi 16 pesan 4-bit, yang lalu akan disusun menjadi matriks 4x4. Representasi pesan dalam 1-digit hexadecimal untuk mempermudah penulisan. Sebagai contoh, untuk pesan dalam representasi heksadesimal “29e58b47a16d03cf”, akan disusun sebagai berikut :

2	9	e	5
8	b	4	7
a	1	6	d
0	3	c	f

Gambar 14. Penyusunan pesan dalam fungsi putaran

Setelah disusun menjadi matriks, maka akan dilakukan beberapa operasi : row substitution dan column substitution. Row substitution adalah operasi substitusi baris pada matriks yang dilakukan pada baris 2 dan 3. Sedangkan column substitution adalah operasi substitusi kolom pada kolom 1 dan 4.

2	9	e	5
a	1	6	d
8	b	4	7
0	3	c	f

Gambar 15. Hasil dari substitusi baris

5	9	e	2
d	1	6	a
7	b	4	8
f	3	c	0

Gambar 16. Hasil dari substitusi kolom

Setelah itu, akan dilakukan substitusi dengan S-box dan transposisi menggunakan P-box. Pembangkitan P-box dilakukan secara sederhana, dengan menggunakan *pseudo-random number generator* yang akan membangkitkan urutan angka dari 0 sampai 15 secara acak berdasarkan suatu random seed. Seed yang digunakan akan bergantung pada key. Desain dari S-box lebih kompleks dan akan dijelaskan pada bagian selanjutnya.

3.4. Desain S-box

S-box didesain dengan menggunakan input key, sehingga membuat S-box akan berubah-ubah berdasarkan key yang diberikan. Terlebih dahulu akan dibangkitkan urutan angka dari 0 sampai 15 yang merupakan hasil dari fungsi linear. Fungsi tersebut memenuhi bentuk

$$y = (Ax + B) \bmod 16$$

Dengan ketentuan bahwa A adalah koprima dengan 16, A dan B adalah bilangan bulat. Pada desain ini, nilai B adalah hasil penjumlahan setiap karakter yang digunakan sebagai key, di mana representasi karakter yang berada dalam bentuk heksadesimal akan diubah menjadi bentuk desimal terlebih dahulu.

Sebagai contoh, untuk A = 69 dan B = 420, akan diperoleh :

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

Setiap nilai dari y akan berada dalam domain nilai untuk 4-bit integer, sehingga setiap nilai dapat direpresentasikan dalam bentuk bilangan biner dengan panjang 4 digit.

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

Dilakukan transposisi antara bit-2 dan bit-3 untuk setiap nilai, lalu nilai-nilai tersebut kembali disusun menjadi bentuk integer dalam desimal, sehingga akan diperoleh nilai y hasil operasi transposisi.

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

Nilai ini akan disusun menjadi matriks 4x4 sehingga akan diperoleh S-box yang akan digunakan pada fungsi putaran.

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

Gambar 17. S-box hasil pembangkitan

4. Eksperimen dan Analisis

4.1. Implementasi

Implementasi program dilakukan menggunakan bahasa Python dengan menggunakan beberapa pustaka pembantu. Program dipecah menjadi beberapa komponen yang direpresentasikan dalam beberapa file, yang dapat dilihat pada representasi berikut :

```
.
└── Weed Cipher/
    ├── feistel.py
    ├── round_func.py
    ├── pbox.py
    ├── sbox.py
    └── vis.py
```

Proses enkripsi dapat dilakukan dengan memanggil fungsi pada feistel.py. Fungsi yang tersedia berkorespondensi dengan mode yang dilakukan, seperti `ebc_encrypt` untuk mode EBC, dan `cbc_encrypt` untuk mode CBC. Setiap fungsi enkripsi akan memanggil fungsi `encrypt_feistel` yang akan melibatkan fungsi putaran yang didefinisikan di dalam round_func.py. Fungsi putaran ini akan memanggil fungsi-fungsi lainnya pada modul-modul lainnya untuk melakukan operasi-operasi yang telah dijelaskan pada bagian sebelumnya.

Dibuat juga fungsi untuk melakukan visualisasi yang menggunakan matplotlib sebagai pustaka utamanya, dan juga beberapa fungsi utilitas, seperti `hex_string` untuk mengubah pesan menjadi dalam bentuk representasi heksadesimal.

4.2. Hasil Eksperimen

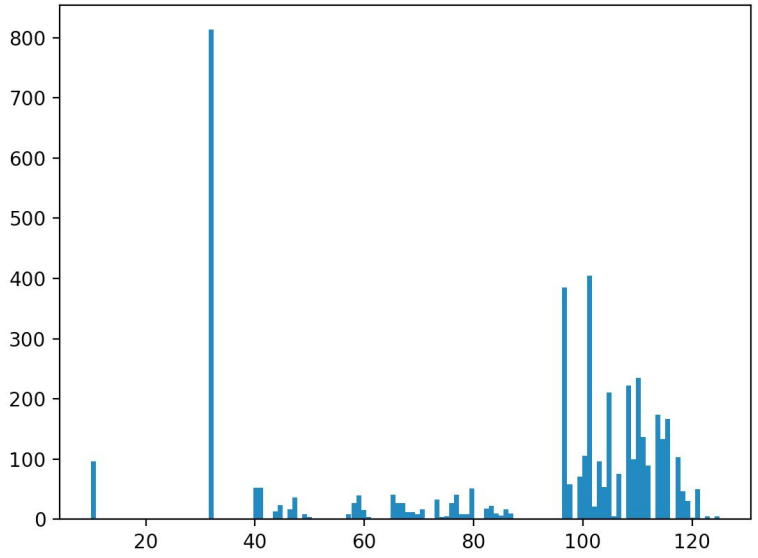
Pada eksperimen kali ini, digunakan plainteks dengan 6417 karakter sebagai berikut.

minimax.py

```
player1 : Player;
player2 : Player;
currentPlayer : Player;
gameBoard : BoardView;
procedure Start() { membuat papan permainan dan menginisialisasi state awal permainan }
    VARIABEL LOKAL
    board : Board;

    ALGORITMA
    board <- new Board();
    player1.agent <- new Agent();
    player2.agent <- new Agent();
    currentPlayer <- player1;
    playet1.StartTurn();

procedure OnEndTurn() { melaksanakan prosedur saat pemain mengakhiri gilirannya }
    DELKARASI VARIABEL DAN FUNGSI
    resetPrevPos : Pos; // Variabel bertipe Pos yang hanya akan menyimpan nilai default
untuk turn awal. Inisialisasi dengan X dan Y sebagai -99
    InitializeDefaultPrevPos() : Pos; // Mengembalikan variabel Pos dengan X dan Y
bernilai -99
    CheckWin() : Boolean; // Melakukan pengecekan apakah kondisi permainan sudah berada
dalam kondisi memiliki pemenang atau tidak
    WinningRoutine() : void; // Melakukan aksi apabila telah ditemukan pemenang

    ALGORITMA
    if (CheckWin() = True) then
        WinningRoutine()
    else
        if (currentPlayer = player1) then
```

Gambar 18. Plainteks untuk dienkripsi

Plainteks tersebut memiliki persebaran frekuensi huruf sebagai berikut.



Gambar 19. Persebaran frekuensi pada plainteks

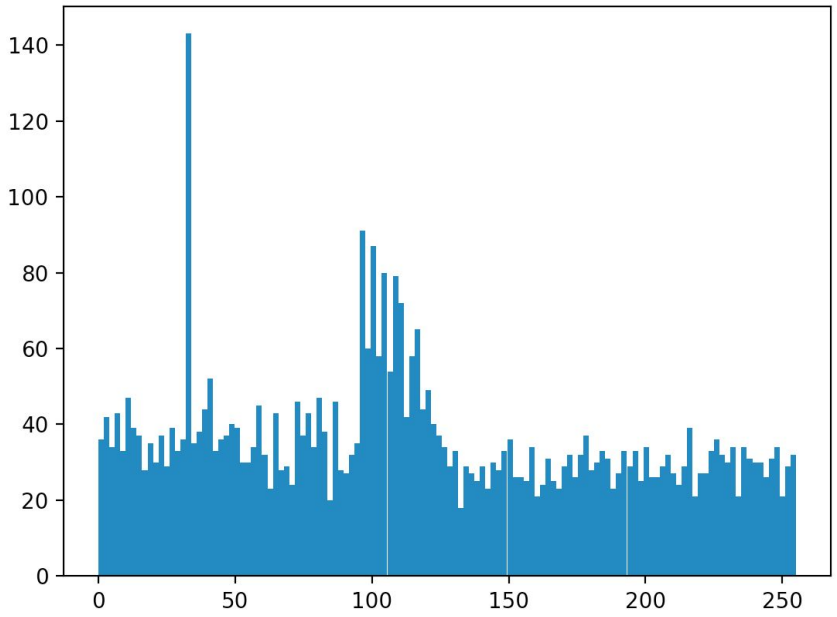
Pada eksperimen kali ini, penulis menggunakan 3 mode Block Cipher yaitu EBC, CBC, dan Counter Mode. Kunci yang digunakan penulis untuk ketiga mode tersebut sama, yaitu “institut amazing” . Berikut ini hasil eksperimen dari masing masing mode.

4.2.1. Mode ECB

Hasil enkripsi:

Gambar 20. Hasil enkripsi mode EBC.

Dari hasil enkripsi tersebut, didapatkan persebaran frekuensi huruf sebagai berikut.



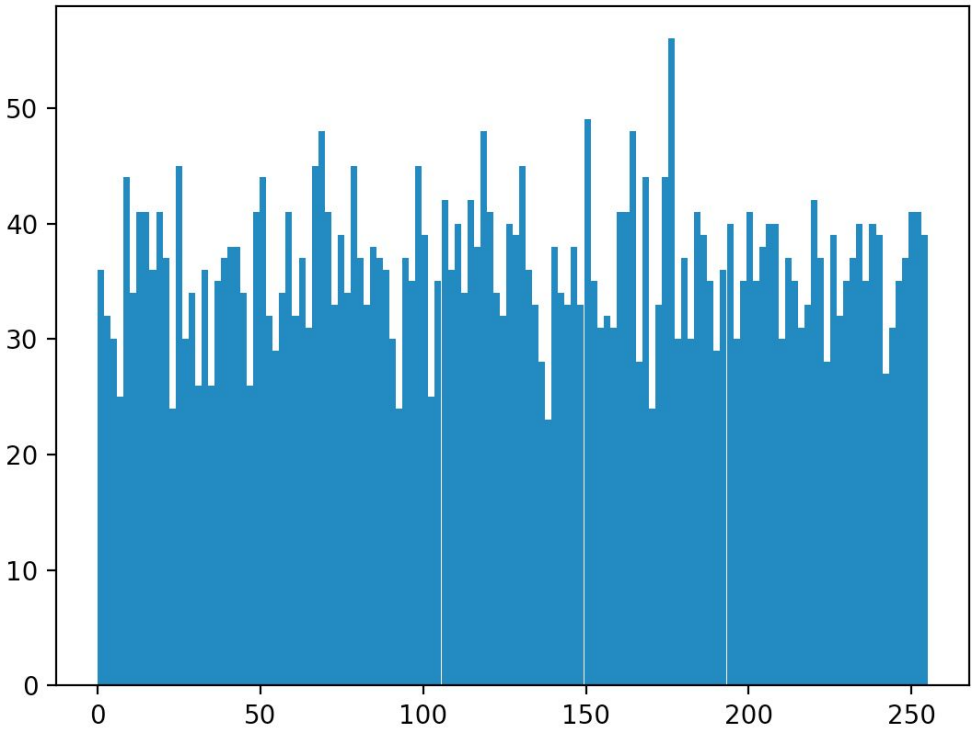
Gambar 21. Grafik frekuensi hasil enkripsi Mode EBC

4.2.2. Mode CBC

Hasil enkripsi:

Gambar 22. Hasil enkripsi mode CBC.

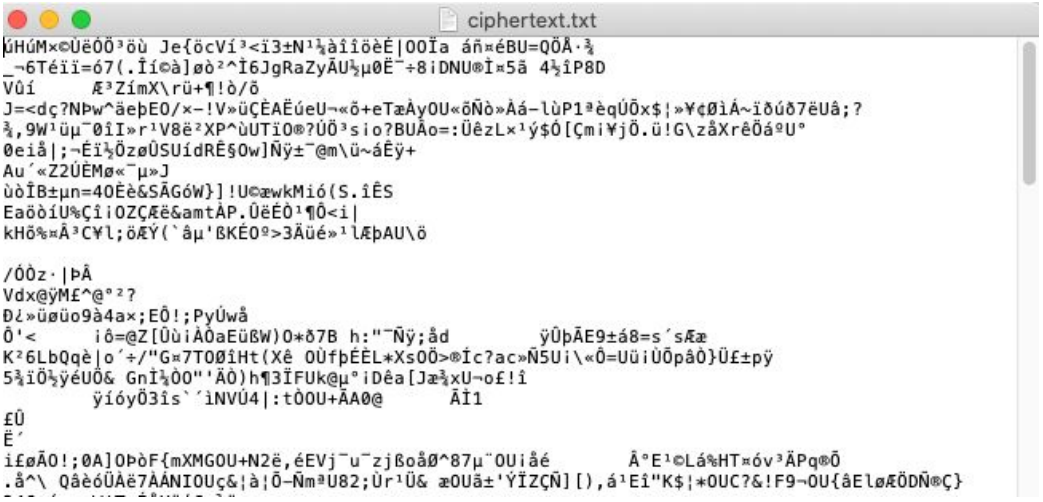
Dari hasil enkripsi tersebut, didapatkan persebaran frekuensi huruf sebagai berikut.



Gambar 23. Grafik frekuensi hasil enkripsi Mode CBC

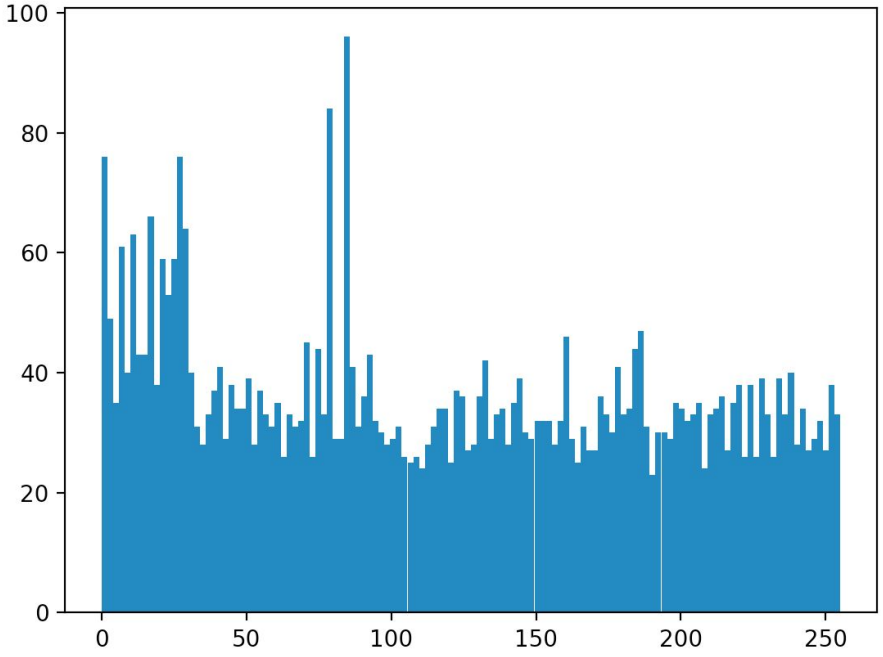
4.2.3. Counter Mode

Hasil enkripsi:



Gambar 24. Hasil enkripsi Counter Mode

Dari hasil enkripsi tersebut, didapatkan persebaran frekuensi huruf sebagai berikut.



Gambar 25. Grafik frekuensi hasil enkripsi Counter Mode

4.3. Analisis Keamanan

4.3.1. Shannon's Confusion

Untuk menganalisis prinsip confusion, penulis membandingkan kemunculan dari frekuensi huruf sebelum enkripsi dan sesudah enkripsi. Terlihat dari hasil eksperimen pada bagian 4.2.1, 4.2.2, dan 4.2.3, terlihat bahwa persebaran frekuensi hasil enkripsi terlihat merata. Hal ini menyebabkan sulit dilakukan kriptanalisis dengan analisis statistik pada ciphertext tersebut.

4.3.2. Shannon's Difusion

Pada analisis shannon's diffusion, digunakan dua kata yang sama dengan 1 huruf berbeda untuk mengetahui seberapa jauh perbedaan pada cipherteks hasil enkripsi. Kata yang digunakan pada analisis kali ini yaitu "abcdefghijklmnop" dan "zbcdefghijklmnop". Berikut ini tabel hasil analisis diffusion.

Mode	Enkripsi "abcdefghijklmnop"	Enkripsi "zbcdefghijklmnop"
ECB	";&kldùÚq'<YzÅß3"	"á klÚû_]`q@ïcÊ"
CBC	"± v]+êI`Dö5"	"O^ o´ÐÞ"Gu+9iz"
Counter Mode	"ZæNÂð»ê²ùïµ"	"[åOÍñºë±øðõ"

Dari hasil analisis tersebut, Weed Block Cipher telah menerapkan prinsip Shannon's Diffusion.

4.3.3. Serangan Brute Force

Algoritma Weed Block Cipher menggunakan kunci dengan panjang minimal 128 bit. Sehingga terdapat minimal 2^128 kunci yang harus dicoba untuk dapat menemukan kunci yang digunakan.

4.3.4. Good S-box Criteria

Kriteria dari S-box yang baik adalah pembangkitan nilai tidak dapat direpresentasikan sebagai sebuah fungsi linear secara langsung, dan juga merupakan bijungsi antara seluruh nilai yang mungkin. Untuk desain dari S-box, dilakukan analisis terhadap setiap bitplane yang dibentuk pada hasil.

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

Apabila dilihat dengan saksama, maka untuk bitplane 1-3 terdapat pola yang dapat diidentifikasi, dengan pola 01 berulang untuk bitplane 1, pola 10100101 berulang untuk bitplane 2, dan pola 0011 berulang untuk bitplane 3. Setiap bitplane ini dapat direpresentasikan dalam suatu fungsi linear, terlebih karena terdapat suatu pola yang berulang.

Namun untuk bitplane 4, karena tidak ada pola yang berulang, maka representasi fungsi untuk bitplane 4 tidak akan bisa menjadi suatu fungsi linear, sehingga terdapat satu fungsi nonlinear. Mengutip Tavares, "We can guarantee nonlinearity, then, simply by forcing at least one of these functions to be nonlinear."[3] pada papernya yang berjudul "The Structured Design of Cryptographically Good S-boxes", maka desain yang diberikan akan memberi hasil yang nonlinear, sehingga memenuhi salah satu kriteria nonlinearity.

5. Kesimpulan dan Saran Pengembangan

Weed Block Cipher yang dirancang sudah cukup baik dalam melakukan enkripsi dan dekripsi pesan. Algoritma telah memenuhi prinsip *confusion* dan *diffusion*, serta ukuran key yang panjang membuat kunci sulit dipecahkan dengan serangan *brute force*.

Pengembangan kedepannya dapat berfokus pada pembuatan S-box yang jauh lebih nonlinear, dapat menggunakan beberapa metode seperti penggunaan Bent Function. Selain itu, desain block cipher dapat dibuat lebih efisien lagi, terkhusus untuk EBC, agar bisa tercapai prinsip diffusion dan confusion dengan menggunakan banyak putaran yang lebih sedikit. Panjang block cipher juga dapat diekspansi sehingga bisa mencakup 256-bit ataupun panjang bit yang lebih tinggi.

6. References

[1] Munir, Rinaldi. 2020. Slide Kuliah IF4020 Kriptografi : Kriptografi Modern (Bagian 3: Block Cipher).

[2] Munir, Rinaldi. 2020. Slide Kuliah IF4020 Kriptografi : Review Beberapa Block Cipher dan Stream Cipher (Bagian 4: Advanced Encryption Standard (AES))

[3] Carslie Adams and Stafford Tavares 1990 *The Structured Design of Cryptographically Good S-Boxes* Journal of Cryptology 3 p 34