Perbandingan Super-Enkripsi Berulang vs *Vigènere Cipher* Kunci Berlapis Metoda *Triple DES*

Christian Angga – 13508008

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
ca_ers@hotmail.com

***Abstract*— Pada saat ini, telah banyak sekali algoritma-algoritma kriptografi yang beredar di masyarakat, beberapa jenis diantaranya adalah algoritma kriptografi klasik dan algoritma kriptografi modern. Pada dasarnya, perbedaan diantara keduanya hanyalah sistem pengoperasiannya. Pada algoritma kriptografi modern, proses enkripsi dekripsi beroperasi dalam mode bit, sedangkan pada algoritma kriptografi klasik, proses enkripsi dan dekripsi beroperasi dalam mode karakter.**

Pada kenyataannya, *cost* untuk melakukan proses enkripsi dan dekripsi pada algoritma kriptografi modern tergolong lebih murah daripada melakukan proses enkripsi dan dekripsi pada algoritma kriptografi klasik. Hal ini disebabkan karena proses eksekusi enkripsi dan dekripsi pada algoritma kriptografi modern dilakukan bit per bit, sedangkan pada algoritma kriptografi klasik proses eksekusi dilakukan sekaligus byte per byte.

Namun pada kesempatan kali ini, penulis tidak mencoba untuk membandingkan *cost* dari algoritma kriptografi klasik dengan algoritma kriptografi modern, tetapi penulis mencoba melakukan kolaborasi ide-ide yang digunakan algoritma-algoritma klasik dan modern untuk meningkatkan keamanan dari algoritma kriptografi klasik kuno yaitu *Vigènere Cipher*. Ide yang akan dibandingkan penulis ada 2, yang pertama adalah super-enkripsi berulang, dan yang kedua adalah algoritma *Vigènere Cipher* kunci berlapis metoda *Triple DES*.

Pada makalah ini, penulis mencoba memodifikasi dan mengkolaborasikan algoritma klasik yang ada agar lebih sulit untuk di dekripsi seperti layaknya algoritma kriptografi modern. Algoritma super-enkripsi berulang dimaksudkan untuk mengulang sebanyak n kali cipher substitusi dan cipher transposisi, algoritma cipher substitusi yang dipakai adalah *Vigènere Cipher*. Sedangkan *Vigènere Cipher* kunci berlapis metoda *Triple DES* maksudnya adalah melakukan enkripsi dan dekripsi algoritma *Vigènere Cipher* dengan kunci yang berbeda sebanyak n kali juga sesuai dengan metoda *Triple DES*.

***Index Terms*—Super-Enkripsi, *Vigènere Cipher*, Cipher Transposisi, *Triple DES*.**

I. INTRODUCTION

Kriptografi (*cryptography*) berasal dari bahasa Yunani, terdiri dari dua suku kata yaitu *kripto* dan *graphia*. *Kripto* artinya menyembunyikan, sedangkan *graphia* artinya tulisan. Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi, seperti kerahasiaan data, keabsahan data, integritas data, serta autentikasi data (Menezes, Oorschot and Vanstone, 1996). Tetapi tidak semua aspek keamanan informasi dapat diselesaikan dengan kriptografi.

Kriptografi dapat pula diartikan sebagai ilmu atau seni untuk menjaga keamanan pesan. Ketika suatu pesan dikirim dari suatu tempat ke tempat lain, isi pesan tersebut mungkin dapat disadap oleh pihak lain yang tidak berhak untuk mengetahui isi pesan tersebut. Untuk menjaga pesan, maka pesan tersebut dapat diubah menjadi suatu kode yang tidak dapat dimengerti oleh pihak lain.

Semakin cepatnya kemajuan teknologi yang berkembang pada saat ini, perkembangan akan bidang kriptografi pun semakin pesat juga, hal itu didasari untuk menjaga kerahasiaan seseorang akan data yang mereka miliki. Algoritma Kriptografi Klasik yang dulunya hanya menggunakan kertas dan alat tulis saja serta hanya berbasis karakter, kini berkembang yang disebut Algoritma Kriptografi Modern. Algoritma Kriptografi Modern digunakan untuk menyembunyikan pesan digital pada komputer atau perangkat digital lainnya yang biasa disebut dengan metoda enkripsi.

Enkripsi adalah sebuah proses penyandian yang melakukan perubahan sebuah kode (pesan) dari yang bisa dimengerti (*plainteks*) menjadi sebuah kode yang tidak bisa dimengerti (*cipherteks*). Sedangkan proses kebalikannya untuk mengubah *cipherteks* menjadi *plainteks* disebut dekripsi. Proses enkripsi dan dekripsi memerlukan suatu mekanisme dan kunci tertentu.

II. TEORI DASAR

II.1. Algoritma Kriptografi Klasik

Algoritma kriptografi klasik umumnya berupa algoritma enkripsi yang berbasiskan karakter, yaitu enkripsi terhadap semua huruf alfanumerik pada umumnya. Algoritma kriptografi klasik cukup sederhana dan mudah digunakan. Beberapa algoritma kriptografi klasik yang cukup terkenal antara lain algoritma cipher substitusi dan algoritma cipher transposisi.

A. Cipher Substitusi

Salah satu contoh algoritma cipher substitusi adalah algoritma *Caesar Cipher*. Ide dasar algoritma *Caesar Cipher* adalah dengan menggeser urutan abjad sebanyak n karakter pada setiap huruf di *plainteks*. Algoritma ini rentan dipecahkan dengan *exhaustive search* karena karakter dalam huruf alfabet hanya ada 26. Rumusan untuk enkripsi dan dekripsi pada *Caesar Cipher*, dengan pergeseran huruf sejauh k, dengan asumsi huruf A = 0, B = 1, ... , Z = 25, adalah sebagai berikut :

Enkripsi: ci = E(pi) = (pi + k) mod 26 (1)
Dekripsi: pi = D(ci) = (ci – k) mod 26 (2)
k = kunci rahasia

Kemudian, dikembangkanlah algoritma-algoritma enkripsi lain dengan berdasar pada algoritma *Caesar Cipher*, dengan tujuan meningkatkan keamanan dari enkripsi. Salah satunya algoritma hasil modifikasi dari *Caesar Cipher* adalah *Vigènere Cipher*.

B. Cipher Transposisi

Ide dasar cipher transposisi adalah cipherteks diperoleh dengan mengubah posisi huruf di dalam plainteks. Dengan kata lain, algoritma ini melakukan *transpose* terhadap rangkaian huruf di dalam plainteks. Nama lain untuk metode ini adalah permutasi, karena *transpose* setiap karakter di dalam teks sama dengan mempermutasikan karakter-karakter tersebut.

Contoh: Misalkan plainteks adalah
`DEPARTEMEN TEKNIK INFORMATIKA ITB`

Enkripsi:

```
DEPART
EMENTE
KNIKIN
FORMAT
IKAITB
```

Cipherteks: (baca secara vertikal)
`DEKFIEMNOKPEIRAANKMIRTIATTENTB`

Dekripsi: Bagi panjang cipherteks dengan kunci. (Pada contoh ini, 30 / 6 = 5)

```
DEKFI
EMNOK
PEIRA
ANKMI
RTIAT
TENTB
```

Plainteks: (baca secara vertikal)
`DEPARTEMEN TEKNIK INFORMATIKA ITB`

C. Super-Enkripsi

Pada dasarnya, algoritma super-enkripsi ini adalah menggabungkan cipher substitusi dengan cipher transposisi.

Sebagai contoh:
Plainteks: `HELLO WORLD`

dienkripsi dengan *caesar cipher* menjadi `KHOOR ZRUOG`

kemudian hasil enkripsi ini dienkripsi lagi dengan cipher transposisi (k = 4):

```
KHOO
RZRU
OGZZ
```

Cipherteks akhir adalah: **`KROHZGORZOUZ`**

II.2. Algoritma Kriptografi Modern

Pada dasarnya, perbedaan antara algoritma kriptografi modern dengan algoritma kriptografi klasik adalah algoritma kriptografi modern beroperasi dalam mode bit (algoritma kriptografi klasik beroperasi dalam mode karakter). Kunci, plainteks, cipherteks, diproses dalam rangkaian bit. Pemroresan pada mode bit paling banyak dilakukan oleh operasi bit xor.

Pada dasarnya, algoritma kriptografi modern tetap menggunakan gagasan pada algoritma klasik: substitusi dan transposisi, tetapi lebih rumit (sangat sulit dipecahkan). Perkembangan algoritma kriptografi modern ini didorong oleh penggunaan komputer digital untuk keamanan pesan. Representasikan data dalam biner tersebut pun biasa dilakukan oleh komputer digital itu sendiri.

Kategori algoritma (*cipher*) berbasis bit ada 2, yaitu:

- Cipher Aliran (*Stream Cipher*)
  - beroperasi pada bit tunggal
  - enkripsi/dekripsi bit per bit

- Cipher Blok (*Block Cipher*)
  - beroperasi pada blok bit
    (contoh: 64-bit/blok = 8 karakter/blok)
  - enkripsi/dekripsi blok per blok

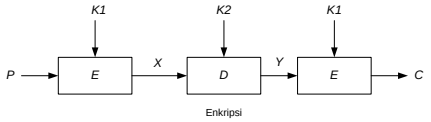
A. *Data Encryption Standard* (DES)

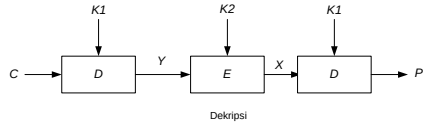
Data Encryption Standard (DES) dikembangkan dan didukung oleh pemerintah AS pada tahun 1977 sebagai standar resmi dan bentuk-bentuk dasar tidak hanya untuk *Automatic Teller Machines* (ATM) PIN otentikasi tetapi varian yang juga digunakan dalam enkripsi sandi UNIX. DES adalah blok cipher dengan ukuran blok 64-bit yang menggunakan kunci 56-bit. Karena kemajuan terbaru dalam teknologi komputer, beberapa ahli tidak lagi mempertimbangkan DES aman terhadap semua serangan, sejak saat itu *Triple-DES* (3DES) telah muncul sebagai metode kuat. Menggunakan enkripsi standar DES, *Triple-DES* mengenkripsi data tiga kali dan menggunakan kunci yang berbeda untuk setidaknya satu dari tiga lewat memberikan ukuran kunci kumulatif 112-168 bit.

B. *Triple DES*

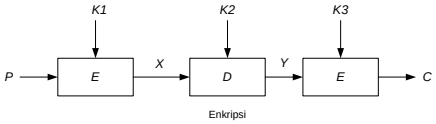
Karena DES mempunyai potensi kelemahan pada *brute force attack*, maka dibuat varian dari DES. Varian DES yang paling luas digunakan adalah DES berganda (*multiple DES*). DES berganda adalah enkripsi berkali-kali dengan DES dan menggunakan kunci ganda. *Triple DES*:

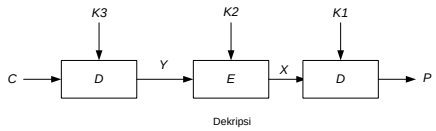
- Menggunakan DES tiga kali
- Bertujuan untuk mencegah *meet-in-the-middle attack*.
- Bentuk umum TDES (mode EEE):
  Enkripsi: $C = E_{K3}(E_{K2}(E_{K1}(P)))$
  Dekripsi: $P = D_{K1}(D_{K2}(D_{K3}(C)))$
- Untuk menyederhanakan TDES, maka langkah di tengah diganti dengan D (mode EDE).
- Ada dua versi TDES dengan mode EDE:
  - Menggunakan 2 kunci





  - Menggunaakn 3 kunci





III. PROSES MODIFIKASI 2 ALGORITMA *VIGÈNERE CIPHER*

A. Super-enkripsi Berulang

Super-enkripsi Berulang adalah algoritma modifikasi tipe pertama terhadap algoritma *Vigènere Cipher* yang menggunakan teknik super-enkripsi namun dilakukan berulang sebanyak 3 kali pengulangan dengan menggunakan kunci yang sama. Algoritma ini pada dasarnya bekerja seperti:

Misalkan plainteks adalah
`saya sedang membuat makalah kripto`

Key adalah `kunci`

Maka pertama-tama, plaintext akan dienkripsi dengan algoritma *Vigènere Cipher* dengan key yang ada dengan cara:

```
saya sedang membuat makalah kripto
kunc ikunci kunciku ncikunc ikunci
---------------------------------
culc aoxnpo wyzdckn zcskfnj sbccvw
```

Setelah dienkripsi dengan cara *Vigènere Cipher*, lalu dilanjutkan dengan pengenkripsian dengan menggunakan model cipher transposisi dengan membaginya menjadi 3 baris dan menyusunnya kembali seperti:

Hasil enkripsi *Vigènere Cipher:*
`culc aoxnpo wyzdckn zcskfnj sbccvw`

dipecah menjadi 3 baris menjadi:

```
cul
c a
oxn
po
wyz
dck
n z
csk
fnj
 sb
ccv
w
```

lalu dilakukan metoda enkripsi transposisi menjadi:
`ccopwdncf cwu xoyc snsc lan zkzkjbv`

Setelah melakukan enkripsi *Vigènere Cipher* dan enkripsi transposisi, itu berarti proses pengenkripsian algoritma super-enkripsi telah selesai dijalankan, namun dikarenakan dilakukannya modifikasi akan algoritma super-enkripsi tersebut, maka pada modifikasi tipe pertama ini, proses super-enkripsi

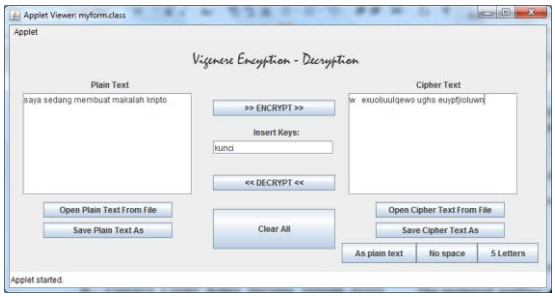
tersebut dilakukan berulang sebanyak 3 kali. Dengan melakukannya berulang sebanyak 3 kali, maka cipherteks yang berhasil digenerate adalah:

```
w    exuoliuulqewo ughs euypfjioluwn
```

Untuk memperjelas cara kerja algoritma Super-enkripsi Berulang ini, dapat memperhatikan contoh pseudo-code dalam bahasa java berikut:

```
    private static String encrypt(String pT, String
keys) {
        String cT = new String();
        char tempChar;
        int tempInt;
        int key_iter = 0;
        for (int i = 0; i < pT.length(); i++) {
            if (pT.charAt(i)==' ') {
                cT = cT + pT.charAt(i);
            }
            else {
                tempInt = (((int)pT.charAt(i) +
(int)keys.charAt(key_iter % keys.length()) - 97));
                if (tempInt > 122)   //melewati z
                    tempInt -= 26;
                tempChar = (char) tempInt;
                cT = cT + tempChar;
                key_iter ++;
            }
        }

        while (cT.length() % 3 != 0)
        {
            cT = cT + ' ';
            padding++;
        }

        String cTlast = new String();
        for (int i = 0; i < 3; i++)
        {
            for (int kelipatan = 0; kelipatan <
cT.length()/3; kelipatan++)
            {
                cTlast = cTlast +
cT.charAt(kelipatan*3+i);
            }
        }

        return cTlast;
    }

     //encrypt function, return the cipher text
    private static String decrypt(String cTlast, String
keys) {
        String pT = new String();
        for (int i = 0; i < cTlast.length()/3; i++)
        {
            for (int kelipatan = 0; kelipatan < 3;
kelipatan++)
            {
                pT = pT +
cTlast.charAt(kelipatan*cTlast.length()/3+i);
            }
        }

        while (padding != 0)
        {
            padding = 0;
        }

        String pTlast = new String();
        char tempChar;
        int tempInt;
        int key_iter = 0;
        for (int i = 0; i < pT.length(); i++) {
            if (pT.charAt(i)==' ') {
                pTlast = pTlast + pT.charAt(i);
            }
            else {
                tempInt = (((int)pT.charAt(i) -
(int)keys.charAt(key_iter % keys.length()) + 97));
                if (tempInt < 97)   //melewati a
                    tempInt += 26;
                tempChar = (char) tempInt;
                pTlast = pTlast + tempChar;
                key_iter ++;
            }
        }

        return pTlast;
}
```

Hasil tampilan yang digenerate oleh program java tersebut adalah:



Gambar 1. Interface Algoritma Super-enkripsi Berulang

B. *Vigènere Cipher* Kunci Berlapis Metoda *Triple DES*

Vigènere Cipher Kunci Berlapis Metoda *Triple DES* merupakan modifikasi tipe kedua terhadap algoritma *Vigènere Cipher*. Pada dasarnya, modifikasi *Vigènere Cipher* ini menggunakan teknik *Triple DES* yang menggunakan lebih dari satu kunci untuk mengenkripsinya maupun untuk mendekripsinya, namaun enkripsi dan dekripsi yang dilakukan semuanya menggunakan algoritma *Vigènere Cipher*.

Untuk memudahkan pengguna menghafal kunci yang dimilikinya, maka masukan kunci yang perlu digunakan hanyalah satu, namun dari satu kunci tersebut, digenerate menjadi 3 buah kunci yang berbeda namun memiliki pola. Algoritma ini pada dasarnya bekerja seperti:

Misalkan plainteks adalah
`saya sedang membuat makalah kripto`

Key adalah `kunci`

Maka pertama-tama, plaintext akan dienkripsi dengan algoritma *Vigènere Cipher* dengan key yang ada dengan cara:

```
saya sedang membuat makalah kripto
kunc ikunci kunciku ncikunc ikunci
--------------------------------
culc aoxnpo wyzdckn zcskfnj sbccvw
```

Setelah selesai dieksekusi, pada dasarnya algoritma *Vigènere Cipher* biasa telah selesai digenerate, dan plainteks yang ada pun telah dienkripsi. Namun algoritma *Vigènere Cipher* ini akan dimodifikasi dengan menggunakan teknik *Triple DES*. Cara pembangkitan kunci pertama sampai dengan seterusnya

adalah dengan menggeser 1 karakter paling depan kunci menjadi karakter paling belakang kunci seperti:

```
kunci -> uncik -> nciku -> cikun -> …
```

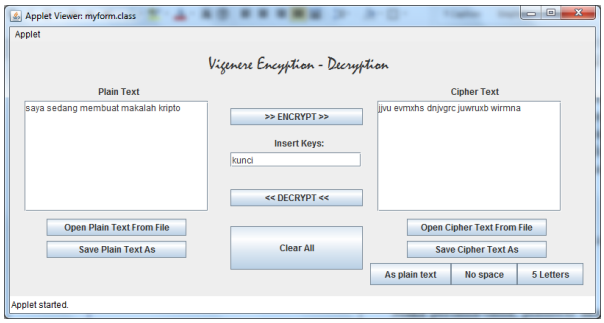
dengan demikian, dengan menggunakan *Vigènere Cipher* Kunci Berlapis Metoda *Triple DES* ini, dihasilkan cipherteks berupa:

```
jjvu evmxhs dnjvgrc juwruxb wirmna
```

Untuk memperjelas cara kerja algoritma *Vigènere Cipher* Kunci Berlapis Metoda *Triple DES* ini, dapat memperhatikan contoh pseudo-code dalam bahasa java berikut:

```
    private static String encrypt(String pT, String
keys) {
        String cT = new String();
        char tempChar;
        int tempInt;
        String newkey = new String();

        if (ulang == 0)
        {
            return cT;
        }
        else
        {
            int key_iter = 0;
            for (int i = 0; i < pT.length(); i++) {
                if (pT.charAt(i)==' ') {
                    cT = cT + pT.charAt(i);
                }
                else {
                    tempInt = (((int)pT.charAt(i) +
(int)keys.charAt(key_iter % keys.length()) - 97));
                    if (tempInt > 122)  //melewati z
                        tempInt -= 26;
                    tempChar = (char) tempInt;
                    cT = cT + tempChar;
                    key_iter ++;
                }
            }

            for (int key = 1; key < keys.length();
key++)
            {
                newkey = newkey + keys.charAt(key);
            }
            newkey = newkey + keys.charAt(0);

            ulang--;
            encrypt(cT, newkey);
        }
        return cT;
    }

     //encrypt function, return the cipher text
    private static String decrypt(String cT, String
keys) {
        String pT = new String();
        char tempChar;
        int tempInt;
        String newkey = new String();

        if (ulang == 3)
        {
            return pT;
        }
        else
        {
            newkey = newkey +
keys.charAt(keys.length()-1);
            for (int key = 0; key < keys.length()-1;
key++)
                {
                    newkey = newkey + keys.charAt(key);
                }

            int key_iter = 0;
            for (int i = 0; i < cT.length(); i++) {
                if (cT.charAt(i)==' ') {
                    pT = pT + cT.charAt(i);
                }
                else {
                    tempInt = (((int)cT.charAt(i) -
(int)keys.charAt(key_iter % keys.length()) + 97));
                    if (tempInt < 97)  //melewati a
                        tempInt += 26;
                    tempChar = (char) tempInt;
                    pT = pT + tempChar;
                    key_iter ++;
                }
            }
            ulang++;
            decrypt(pT, newkey);
        }
        return pT;
}
```

Hasil tampilan yang digenerate oleh program java tersebut adalah:



Gambar 2. Interface Algoritma *Vigènere Cipher* Kunci Berlapis Metoda *Triple DES*

IV. ANALISIS DAN PERBANDINGAN

A. Super-enkripsi Berulang

Jika panjang plainteks 9, dan panjang kunci adalah 3, maka yang akan terjadi:

Plaintext: `abcdefghi`
Kunci: `xyz`
Proses:

```
a = a + x
b = b + y
c = c + z
d = d + x
e = e + y
f = f + z
g = g + x
h = h + y
i = i + z
```

tukar urutan menjadi:

```
a = a + x + x
d = d + x + y
g = g + x + z
b = b + y + x
e = e + y + y
h = h + y + z
c = c + z + x
```

```
f = f + z + y
i = i + z + z
```

tukar urutan menjadi:

```
a = a + x + x + x
b = b + y + x + y
c = c + z + x + z
d = d + x + y + x
e = e + y + y + y
f = f + z + y + z
g = g + x + z + x
h = h + y + z + y
i = i + z + z + z
```

Dapat dilihat dari contoh kasus di atas, jika panjang plainteks adalah 9, serta panjang kunci hanyalah 3, maka akan menimbulkan suatu deret yang dapat memudahkan kriptanalis mengenkripsi cipherteks yang ada. Namun jika kasus yang terjadi adalah plainteks cukup panjang, dan kunci pun panjang, serta makin banyak pengulangan akan metoda super-enkripsi tersebut, maka cipherteks yang dihasilkan pun makin kuat.

B. *Vigènere Cipher* Kunci Berlapis Metoda *Triple DES*

Analisis metoda *Triple DES* dengan dasar algoritma *Vigènere Cipher* dengan panjang plainteks 10 karakter dan kunci 5 karakter:

Plaintext: `abcdefghij`
Kunci: `kunci`
Proses:

```
abcdefghij -> plainteks
kuncikunci -> kunci 1
uncikuncik -> kunci 2
ncikunciku -> kunci 3
==================================
```

Hasil:

```
a = a + k + u + n
b = b + u + n + c
c = c + n + c + i
d = d + c + i + k
e = e + i + k + u
f = f + k + u + n
g = g + u + n + c
h = h + n + c + i
I = i + c + i + k
j = j + I + k + u
```

Dari kasus diatas, bisa dilihat adanya deretan kunci yang dihasilkan. Hal tersebut menyebabkan munculnya peluang bagi kriptanalis untuk mencari deretan kunci tersebut, lalu mendekripsinya. Jika metoda *Triple DES* ini diulang sebanyak panjang kunci, maka akan menghasilkan algoritma *Caesar Cipher* yang merupakan algoritma kuno yang sangat tidak aman, bahkan lebih tidak aman daripada algoritma *Vigènere Cipher*, karena semua kunci sama, hanya dilakukan pergeseran karakter sejumlah kunci.

V. CONCLUSION

Baik algoritma Super-enkripsi Berulang maupun algoritma *Vigènere Cipher* Kunci Berlapis Metoda *Triple DES*, masing-masing memiliki kekurangan dan kelebihan masing-masing. Pada algoritma Super-enkripsi Berulang, karena ini merupakan gabungan antara algoritma substitusi dan algoritma transposisi, jika dilakukannya berulang-ulang, maka tingkat keamanan akan meningkat juga. Namun dikarenakan algoritma ini membagi plainteks menjadi 3 bagian, maka jikalau jumlah plainteks kelipatan 3 karakter, dan panjang kunci kelipatan 3 karakter juga, itu akan menyebabkan terjadinya sebuah deret yang memudahkan kriptanalis mendekripsi pesan. Berbeda kasus dengan algoritma *Vigènere Cipher* Kunci Berlapis Metoda *Triple DES*, jika algoritma ini dilakukan berulang-ulang, hasil enkripsi yang dihasilkan tidak akan bertambah kuat, malah jika diulang sebanyak kelipatan panjang kunci yang ada, algoritma tersebut akan menjadi sangat tidak aman dibandingkan algoritma standart yang dipake (*Vigènere Cipher*).

REFERENCES

The Vigenere Cipher
http://www.cs.trincoll.edu/~crypto/historical/vigenere.html
Waktu akses : 1 Maret 2011.

High-Speed DES and Triple DES Encryptor/Decryptor
http://www.xilinx.com/support/documentation/application_notes/xapp270.pdf
Waktu akses : 1 Maret 2011.

Kriptografi
http://www.scribd.com/doc/49181550/20080916-KRIPTOGRAFI
Waktu akses : 1 Maret 2011.

Definisi Kriptografi
http://sandi.math.web.id/?p=3
Waktu akses : 8 Maret 2011

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, 11 Maret 2011

Christian Angga
13508008