PENGAMANAN DATA TEKS MENGGUNAKAN ALGORITMA KRIPTOGRAFI RC4 DARI SERANGAN BRUTE FORCE

Kristin D R Sianipar[1], Lia Cintia Purba[2], Septri Wanti Siahaan[3], Indra Gunawan[4]
[1,2,3,4,5]Program Studi Teknik Informatika
[1.2..3.4.5]STIKOM Tunas Bangsa Pematangsiantar
[1.2.3.4.5]Jl. Jend. Sudirman Blok A, No. 1,2 dan 3, Kota Pematangsiantar, Sumatera Utara
[1]kristinsianipar7@gmail.com
[2]liapurba99@gmail.com
[3]septriwanti26@gmail.com
[4]indra@amiktunasbangsa.ac.id

Abstrak

Abstrak— Pada perkembangan ilmu teknologi saat ini, sangat penting untuk kita menjaga keamanan dari keaslian data yang kita miliki. Karena perkembangan ilmu teknologi bukan hanya dimanfaatkan untuk hal positif, namun banyak pihak yang telah menyalahgunakannya. Dengan cara, mengubah data-data yang asli atau mencuri data tersebut untuk kepentingan pribadi yang sangat merugikan dari pihak pemiliknya, Contohnya seperti data teks. Untuk mengurangi rasa khawatir kita pada data yang kita punya, kita dapat mengamankan data kita dengan menggunakan Algoritma Kriptografi RC4 dari serangan-serangan yang membahayakan data milik kita. Seperti serangan Brute Force.

Kata Kunci : *brute force*, kriptografi, RC4, data teks, pengamanan data

1. Pendahuluan

1.1. Definisi Brute Force

Brute Force adalah algoritma yang memecahkan masalah dengan sangat sederhana, langsung dan dengan cara yang jelas/lempang. Penyelesaian permasalahan kode cracking dengan menggunakan algoritma brute force akan menempatkan dan mencari semua kemungkinan kode dengan masukan karakter dan

DOI: https://doi.org/10.29103/techsi.v11i2.1271

panjang kode tertentu tentunya dengan banyak sekali kombinasi kode. Algoritma brute force adalah algoritma yang lempang atau apa adanya. Pengguna hanya tingal mendefinisakan karakter set yang diinginkan dan berapa ukuran dari kodenya. Tiap kemungkinan kode akan digenerate oleh algoritma ini [1]. Maka, dengan menggunakan algoritma brute force *user* dapat lebih mudah dalam memecahkan keamanan pesan teks.

1.2. Definisi Kriptografi

Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi, seperti kerahasiaan data, keabsahan data, integritas data, serta autentikasi data [2]. Kriptografi berasal dari bahasa Yunani, *crypto* dan *graphia*. *Crypto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). Menurut terminologinya kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika dikirim dari suatu tempat ke tempat yang lain [3].

Jenis Kriptografi

Berdasarkan kunci yang digunakan untuk enkripsi dan dekripsi, algoritma kriptografi dibedakan menjadi dua macam algoritma kriptografi, yaitu: [4]

Kriptografi Kunci Asimetri

Kriptografi kunci asimetri yang sering disebut juga kriptografi kunci publik adalah algoritma yang menggunakan kunci yang berbeda untuk proses enkripsi dan dekripsinya. Algoritma asimetri ini disebut kunci publik karena kunci untuk enkripsi dapat dibuat publik yang berarti semua orang boleh mengetahuinya. Pada kriptografi jenis ini, setiap orang yang berkomunikasi mempunyai sepasang kunci, yaitu kunci privat dan kunci publik. Pengirim mengenkripsi pesan dengan menggunakan kunci publik si penerima pesan (*receiver*). Hanya penerima pesan yang dapat mendekripsi pesan karena hanya dia yang mengetahui kunci privatnya sendiri [4]. Dalam sistem ini,

DOI: https://doi.org/10.29103/techsi.v11i2.1271

kunci enkripsi disebut kunci publik, sementara kunci dekripsi sering disebut kunci privat.

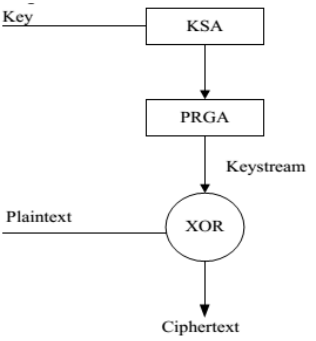
Kriptografi Kunci Simetri

Kriptografi simetri adalah algoritma kriptografi yang menggunakan kunci enkripsi yang sama dengan kunci dekripsinya. Keamanan algoritma simetri tergantung pada kuncinya. Apabila kuncinya diketahui orang lain, maka orang tersebut dapat mengenkrip dan mendekrip pesan.

2. Metode Penelitian

2.1. Algoritma RC4

Algoritma RC4 (*Riverst Cipher 4*) merupakan *stream cipher* yang dirancang di RSA *Security* oleh Ron *Rivest* tahun 1987. Sifat kunci dalam algoritma RC4 adalah simetris serta melakukan proses enkripsi *plain* per digit atau *byte* per *byte* dengan operasi biner (biasanya XOR) dengan sebuah angka semiacak. Namun algoritma ini memiliki kelemahan yaitu mudah diserang dengan teknik *know-plaintext attack* dan *ciphertext-only attack* [5]. Serangan *know-plaintext attack* bisa diartikan jika kriptanalist memiliki potongan *plaintext dan ciphertext*, maka dengan mudah didapatkan aliran kunci dengan cara meng-XOR-kan *plaintext* dengan *ciphertext*. Algortima RC4 bekerja dengan tiga tahap utama yaitu *Key Scheduling Algorithm (KSA), Pseudo Random Generation Algorithm (PRGA)* dan Proses Enkripsi dan Dekripsi [5] dan [6]. Proses di atas, dapat diperlihatkan pada Gambar 1 di bawah ini.



Gambar 1. Blok Diagaram Algoritma RC4

DOI: https://doi.org/10.29103/techsi.v11i2.1271

a. *Key Scheduling Algorithm* (KSA)

Proses KSA merupakan proses pembentukan tabel S-Box (Tabel Array S) dan Kunci (Tabel array [T]) yang di permutasi sebanyak 256 iterasi. *Pseudocode* untuk proses inisialisasi S-Box dan Array T:

```
for (i = 0 ; i<=255; i++){
    S-Box[i] = i
    T[i] = kunci[ i mod panjang_kunci]
}
```

Pseudocode untuk permutasi isi *array* S-Box :

```
j = 0
for (i = 0 ; i<=255; i++){
    j = (j + S-Box[i] + T[i]) mod 256
    Swap( S-Box[i], S[j] )
    j = j
}
```

setelah dua proses ini dilakukan, maka *array* S-Box dan *array* Kunci (T) telah terbentuk.

b. *Pseudo Random Generation Algorithm* (PRGA)

Tabel *array* S-Box akan digunakan pada proses ini untuk menghasilkan *key stream* yang jumlahnya sama dengan jumlah banyaknya karakter *plaintext* kemudian akan di-XOR dengan *plaintext.* Adapun *pseudocode* proses PRGA ini adalah :

```
i = 0; j = i
for (i = 0 ; i <= jlh_karakter_plaintext; i++){
    i = (i + 1) mod 256
    j = (j + S-Box[i]) mod 256
    Swap( S-Box[i], S-Box[j] )
    t = (S-Box[i] + S-Box[j]) mod 256
    Kunci[i] = S-Box[t]
}
```

c. Proses enkripsi atau dekripsi dengan operasi XOR.

Proses enkripsi atau dekripsi di awali dengan merubah setiap nilai *plaintext* ke biner. Formula untuk melakukan proses enkripsi dan dekripsi [7], adalah:

DOI: https://doi.org/10.29103/techsi.v11i2.1271

Formula proses enkripsi:
Formula proses deskripsi
$Ci = Pi \oplus Ki$ (1)
Formula proses dekripsi:
$Pi = Ci \oplus Ki$ (2)

3. Hasil dan Pembahasan

Enkripsi

Enkripsi adalah proses yang dilakukan untuk mengamankan sebuah pesan (yang disebut plaintext) menjadi pesan yang tersembunyi (disebut ciphertext) adalah enkripsi (encryption)[8]. Ciphertext adalah kata/pesan yang telah tersandi dan tidak dapat dibaca dengan mudah.

Dekripsi

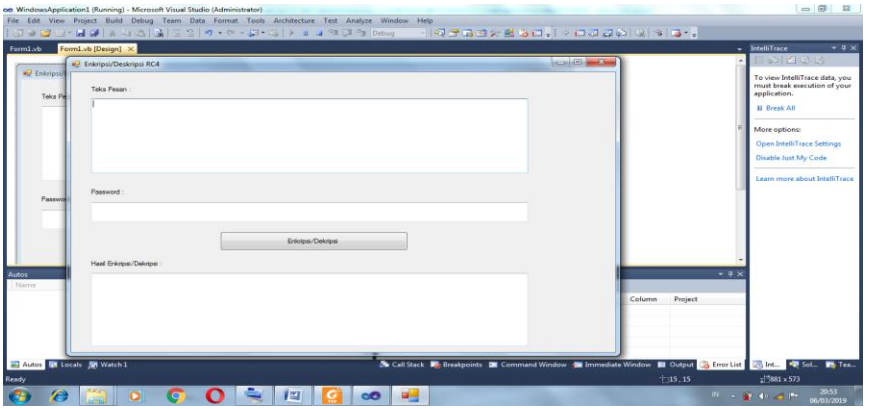
Dekripsi merupakan proses pengembalian data ke bentuk awal dari proses penyandian atau dengan kata lain proses pengubahan ciphertext menjadi plaintext.

3.1 Hasil

Pembangunan form di bawah ini dirancang menggunakan Visual Basic .NET 2010. Pada aplikasi ini proses enkripsi dan dekripsi tidak dipisahkan halamannya.

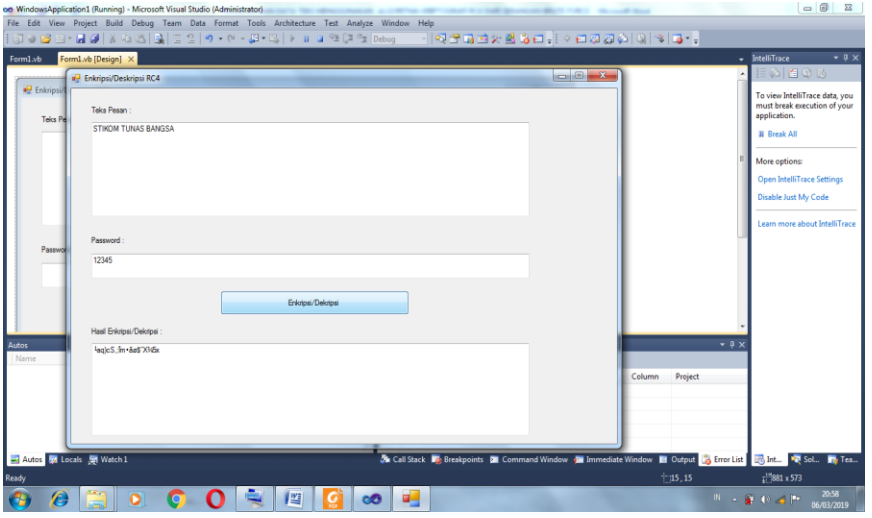
1). Button 1 pada form Enkripsi/Dekripsi RC4 berfungsi sebagai button enkripsi dari data teks yang telah kita ketik, dan juga berfungsi sebagai button dekripsi untuk pengembalian data teks dari ciphertext. Jika button tersebut ditekan maka data teks yang telah kita ketik akan berubah menjadi ciphertext.

DOI: https://doi.org/10.29103/techsi.v11i2.1271



Gambar 1. Form Enkripsi/Dekripsi RC4

Gambar diatas masih tampilan awal pada form enkripsi dan dekripsi RC4.



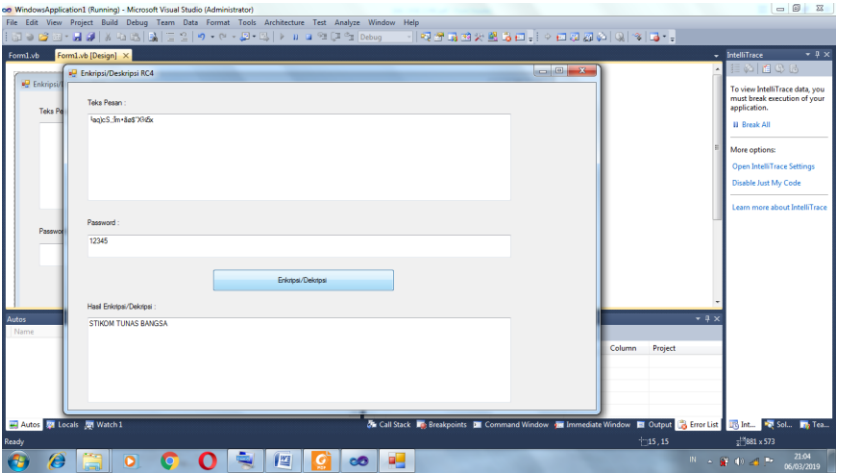
Gambar 2. Form Hasil Enkripsi

Pada Gambar diatas kita ketik data teks 'STIKOM TUNAS BANGSA' dan kita ketikkan password '12345' maka akan menghasilkan ciphertext

————————————aq)cS,;Îm•ãø$"X¾5x

DOI: https://doi.org/10.29103/techsi.v11i2.1271

2). Pada button1 juga kita dapat mengubah data teks dari ciphertext menjadi plaintext. Jika kita *copy* hasil ciphertext yang telah kita enkripsi tadi ke textbox teks pesan, maka hasil dekripsi akan kembali ke plaintext yang kita ketik pertama kali tadi.



Gambar 3. Form Hasil Dekripsi

Pada Gambar diatas kita *paste* hasil ciphertext ——————————————aq)cS,;Îm•ãø$"X¾5x untuk membuktikan hasil dekripsi kita yang akan diubah menjadi plaintext diawal tadi. Lalu, kita ketik password '12345' maka akan menghasilkan plaintext yang kita ketik pertama kali yaitu 'STIKOM TUNAS BANGSA' .

4. Kesimpulan

Penggunaan Algoritma Kriptografi RC4 dapat digunakan untuk memberi pengamanan pada data teks, sehingga dengan menggunakan Algoritma Kriptografi RC4 kita dapat mengamankan semua data dari serangan brute force.

Daftar Pustaka

Purba, L.C., Sianipar, K.R.D, Siahaan, S.W., Gunawan, I. & Sumarno. "PENGAMANAN FILE GAMBAR

MENGGUNAKAN FUNGSI ALGORITMA STEGANOGRAFI LSB DARI SERANGAN BRUTE FORCE". TECHSI: Vol. 10, No. 1, April 2018.

Menezes, A.J., Oorschot, P.V. & Vanstone, S. 1996. *Handbook of Applied Cryptography*. CRC Press: New York

Ariyus, D. 2008. *Pengantar Ilmu Kriptografi*. Andi Offset: Yogyakarta.

Munir, R. 2006. *Kriptografi*. Informatika: Bandung.

SETYANINGSIH, E. 2009. Penyandian Citra Menggunakan Metode Playfair Cipher, J. Teknol., vol. 2, no. 2, pp. 213–219.

AGUNG, H. & BUDIMAN. 2015. Implementasi Affine Chiper dan RC4 Pada Enkripsi File Tunggal. Prosiding SNATIF, pp. 243–250.

ZEBUA, T. 2013. Analisa dan Implementasi Algoritma Triangle Chain pada Penyandian Record Database. Pelita Inform. Budi Darma, vol. 3, no. 2, pp. 3749.

Aulia, N. APLIKASI ENKRIPSI DAN DEKRIPSI MENGGUNAKAN VISUAL BASIC 2012 DENGAN ALGORITMA TRIPLE DES,20 Mei 2016.

DOI: https://doi.org/10.29103/techsi.v11i2.1271