Jurnal Mantik Penusa

Volume 18, No. 2, Desember 2015

p-ISSN 2088-3943

ENKRIPSI DAN DEKRIPSI FILE TEKS MENGGUNAKAN ALGORITMA IMPROVED CAESAR CIPHER

Jonson Manurung

Departemen Komputer,
AMIK Medicom Medan, Jl. Darat No. 74Medan - (061) 4152616 Kode Pos : 20153

jhonson.geo@gmail.com

Abstract

Confidentiality of a text is an important requirement for the community in this day and age to protect their privacy from people who are not entitled to it . To overcome this required an application that is able to meet the needs of the community in which the application is able to encrypt the text that is not known by others . In cryptography there are many algorithms have evolved . But in this paper we prefer to incorporate some modern and traditional algorithms are becoming a new algorithm named Caesar Cipher Algorithm Improved . The purpose of research is to make encryption and decryption application using a Caesar Cipher Improved algorithms are useful for maintaining the confidentiality of text by using a combination of traditional and modern algorithms . Analysis and the need for the user application will be developed using the methodology iteration (iterative) . Each step or stage in this method greatly simplifies the procedure leading analysts working in the project . The results of the analysis and design of applications using a text encryption and decryption algorithm Improved Caesar Cipher is expected to be forwarded to the implementation stage so that these applications can be applied for the purposes of the people who need to maintain confidentiality during transmission and storage of text.

Keywords *: Cryptography, algorithms, Improved Caesar Cipher*

Abstrak

Kerahasiaan dari suatu teks merupakan suatu kebutuhan yang penting bagi kalangan masyarakat di zaman ini untuk melindungi privasi mereka dari orang-orang yang tidak berhak mengetahuinya. Untuk mengatasi hal tersebut dibutuhkan sebuah aplikasi yang mampu memenuhi kebutuhan masyarakat dimana aplikasi tersebut mampu mengenkripsi teks tersebut agar tidak diketahui oleh orang lain. Pada kriptografi terdapat banyak sekali algoritma yang telah berkembang. Tetapi pada skripsi ini kami lebih memilih untuk menggabungkan beberapa algoritma modern dan tradisonal tersebut menjadi satu algoritma baru dengan nama Algoritma *Improved Caesar Cipher*. Tujuan penelitian adalah untuk membuat aplikasi enkripsi dan dekripsi menggunakan algoritma *Improved Caesar Cipher* yang berguna untuk menjaga kerahasian teks dengan menggunakan gabungan algoritma tradisional dan modern. Analisis dan kebutuhan aplikasi bagi pengguna nantinya dikembangkan dengan menggunakan metodologi *iterasi* (*iterative)*. Setiap langkah atau tahapan pada metode ini sangat mempermudah analis dalam menuntun prosedur kerja proyek. Hasil analisis dan perancangan aplikasi enkripsi dan dekripsi teks menggunakan algoritma *Improved Caesar Cipher* ini diharapkan dapat diteruskan ke tahap implementasi sehingga aplikasi ini dapat diaplikasikan bagi keperluan orang-orang yang membutuhkannya untuk menjaga kerahasian saat pengiriman maupun penyimpanan teks..

Keywords : Kriptografi, algoritma, *Improved Caesar Cipher*

1. Pendahuluan

Peranan ini, perkembangan ilmu pengetahuan dan teknologi telah mempengaruhi segala aspek kehidupan, tak terkecuali dalam aspek keamanan teks. Semakin berkembangnya teknologi, keamanan dari suatu teks menjadi berkurang atau tidak aman dalam hal penyimpanan. Tidak menutup kemungkinan ada pihak ketiga yang ingin merubah atau mengambil teks tersebut. Salah satu cara untuk mempertahankan kerahasiaan dari teks tersebut, maka teks tersebut disandikan menjadi kode-kode yang tidak dipahami, sehingga bila ada pihak ketiga yang ingin merubah akan kesulitan dalam menterjemahkan isi teks yang sebenarnya.

Selanjutnya, untuk mengatasi permasalahan di atas, dapat diselesaikan dengan kriptografi. Kriptografi tidak hanya menyediakan alat untuk keamanan pesan, tetapi juga sekumpulan teknik yang berguna untuk keamanan dan kerahasiaan teks.

Adapun algoritma yang digunakan yaitu *Improved Caesar Cipher* yang merupakan perbaikan dari algoritma *Julius Caesar* dimana pada algoritma *Julius Caesar*, keamanan dari

suatu teks masih kurang terjaga kerahasiaannya diakibatkan algoritmanya yang terlalu mudah dan sederhana untuk dipecahkan. Tetapi dengan adanya perbaikan pada algoritma ini, maka keamanan teks menjadi sangat terjaga kerahasiaannya karena menggunakan sistem keamanan teks yang lebih kuat berupa penambahan kunci yang memperhitungan spasi dalam enkripsi dan dekripsi teksnya sehingga untuk dapat membobol sistem keamanannya, pihak ketiga harus menembus 2 kunci, yaitu kunci geseran dan kunci spasi-nya.

2. Teori

2.1. Sejarah Kriptografi

Kriptografi mempunyai sejarah yang panjang. Informasi yang lengkap mengenai sejarah kriptografi dapat ditemukan di dalam buku David Kahn yang berjudul The Codebreakers. Buku yang tebalnya 1000 halaman ini menulis secara rinci sejarah kriptografi mulai dari penggunaan kriptografi oleh Bangsa Mesir 4000 tahun yang lalu (berupa hieroglyph yang tidak standard pada pyramid) hingga penggunaan kriptografi pada abad ke-20. Secara historis ada empat kelompok orang yang berkonstribusi terhadap perkembangan kriptografi, dimana mereka menggunakan kriptografi untuk menjamin kerahasiaan dalam komunikasi pesan penting, yaitu kalangan militer (termasuk intelijen dan mata-mata), kalangan diplomatik, penulis buku harian, dan pencinta (lovers). Di antara keempat kelompok ini, kalangan militer yang memberikan konstribusi paling penting karena pengiriman pesan di dalam suasana perang membutuhkan teknik enkripsi dan dekripsi yang rumit.

2.2. Teori Kriptografi

Kriptografi adalah bidang ilmu yang sangat penting keberadaannya untuk menjaga kerahasiaan dan keamanan suatu informasi dan data. “Untuk mendefinisikannya lebih lanjut tentang kriptografi, Munir (2006, hal 2) menjelaskan bahwa kriptografi (Cryptography) berasal dari bahasa Yunani : “cryptos” artinya “secret” (rahasia), sedangkan “graphien” artinya “writing” (tulisan). Jadi, kriptografi (Cryptography) berarti “secret writing” (tulisan rahasia)”.

Kriptografi mempunyai beberapa tujuan. (Munir 2006, hal 9) menyampaikan tujuan kriptografi bahwa untuk memberi layanan keamanan (yang juga dinamakan sebagai aspek-aspek keamanan) sebagai berikut :

Kerahasiaan (confidentiality), adalah layanan yang ditujukan untuk menjaga agar pesan tidak dapat dibaca oleh pihak-pihak yang tidak berhak.

Integritas data (data integrity), adalah layanan yang menjamin bahwa pesan masih asli / utuh atau belum pernah dimanipulasi selama pengiriman.

Otentikasi (authentication), adalah layanan yang berhubungan dengan identifikasi, baik mengidentifikasi kebenaran pihak-pihak yang berkomunikasi (user authentication atau entity authentication) maupun mengidentifikasi kebenaran sumber pesan (data origin authentication).

Nirpenyalahgunaan (non-repudiation), adalah layanan untuk mencegah entitas yang berkomunikasi melakukan penyangkalan, yaitu pengiriman pesan menyangkal melakukan pengiriman atau penerima pesan menyangkal telah menerima pesan.

2.3. Kriptografi Kunci Simetri dan Nirsimetri

Munir (2006, hal 13) menyatakan bahwa Selain berdasarkan sejarah yang membagi kriptografi menjadi kriptografi klasik dan kriptografi *modern*, maka berdasarkan kunci yang digunakan untuk enkripsi dan dekripsi, kriptografi dapat dibedakan lagi menjadi **kriptografi kunci-simetri** (*symmetric-key cryptography*) dan **kriptografi kunci-nirsimetri** (*asymmetric-key cryptography*).

Pada sistem kriptografi kunci-simetri, kunci untuk enkripsi sama dengan kunci untuk dekripsi, oleh karena itulah dinamakan kriptografi simetri. Istilah lain untuk kriptografi kunci-simetri adalah kriptografi kunci privat (*private-key cryptography*), kriptografi kunci rahasia (*secret-key cryptography*), atau kriptografi konvensional (*conventional cryptography*).

Jika kunci untuk enkripsi tidak sama dengan kunci untuk dekripsi, maka kriptografinya dinamakan sistem kriptografi nirsimetri. Nama lainnya adalah **kriptografi kunci-publik** (*public-key cryptography*), sebab kunci untuk enkripsi tidak rahasia dan dapat diketahui oleh siapapun (diumumkan ke publik), sementara kunci untuk dekripsi hanya diketahui oleh penerima pesan (karena itu rahasia).

2.4. Algoritma Julius Caesar

Didalam *cipher* substitusi setiap unit *plaintext* diganti dengan satu *unit ciphertext*. Satu “*unit*” di sini bisa berarti satu huruf, pasangan huruf, atau kelompok lebih dari dua huruf. Algoritma substitusi tertua yang diketahui adalah *Caesar cipher* yang digunakan oleh kaisar Romawi, Julius Caesar (sehingga dinamakan juga

c*aesar cipher*), untuk menyandikan pesan yang dia kirim kepada para gubernurnya (Munir 2006, hal 56). Pada *Caesar cipher*, tiap huruf disubstitusi dengan huruf ketiga berikutnya dari susunan alfabet yang sama. Dalam hal ini kuncinya adalah jumlah pergeseran huruf (yaitu 3). Susunan alfabet setelah digeser sejauh 3 huruf membentuk sebuah tabel substitusi sebagai berikut (Munir 2006, hal 56). Jadi, huruf A pada *plaintext* disubstitusikan dengan D, huruf B disubstitusikan dengan E, demikian seterusnya.

2.5. Algoritma SPICA-XB

Algoritma SPICA-XB adalah algoritma yang menggabungkan sifat dari algoritma Caesar Cipher dengan beberapa sifat dari algoritma kriptografi modern. Idenya adalah dengan mengasumsikan algoritma Caesar Cipher layaknya memutar roda (*spinning*). Berbeda dengan algoritma Caesar Cipher biasa, algoritma SPICA-XB bekerja dengan melakukan proses substitusi dengan kunci yang berputar. Perputaran tersebut akan sangat bergantung dari kunci dasar yang ditentukan oleh pembuat pesan sebelumnya. Selain itu, jika dalam algoritma Caesar Cipher biasa proses substitusi akan dilakukan begitu saja (misalkan mengganti A dengan J), dalam algoritma SPICA-XB ini proses enkripsi akan dilakukan dengan terlebih dahulu merubah huruf-huruf yang ada kedalam bentuk binary ASCII 8 bit. Proses enkripsi pun dilakukan dengan melakukan operasi penjumlahan lalu XOR antara *plaintext* dengan kunci.

2.6. Algoritma *Improved Caesar Cipher*

Algoritma *Improved Caesar Cipher* merupakan algoritma gabungan antara algoritma Caesar Cipher dengan algoritma SPICA-XB dengan penambahan kunci spasi sebagai kunci tambahannya. Adapun tahapan-tahapan dalam Algoritma *Improved Caesar Cipher*:

1. Alur Enkripsi Teks
   a. Pembuat pesan menentukan pesan serta kunci geseran yang akan digunakan, jumlah geseran adalah bebas sesuai dengan keinginan pembuat pesan.
   b. Setelah selesai, sistem akan menggeser setiap karakter teks sesuai dengan jumlah kunci geseran yang telah ditentukan sebelumnya dan menghitung lokasi spasi dari teks untuk digunakan sebagai kunci spasi.
   c. Selanjutnya setiap karakter dalam *plaintext* akan dipasangkan dengan kunci spasi pasangannya. Jika panjang kunci spasi < panjang *plaintext*, maka akan dilakukan proses penggulangan kunci spasi untuk setiap karakter dalam *plaintext*.
   d. Setiap karakter dalam *plaintext* akan dirubah ke dalam bentuk *binary* ASCII berukuran 8 bit.
   e. Sebagai persiapan dalam tahapan enkripsi awal, seluruh angka dalam kunci spasi akan dirubah menjadi karakter yang berhubungan. Misalkan 0 akan diberi nilai A, 1 diberi nilai B, dan seterusnya.
   f. Selanjutnya karakter *plaintext* yang saat itu, akan dienkripsi dan dilakukan operasi penjumlahan dengan kunci spasi dalam bentuk *binary* ASCII 8 bit untuk mendapatkan kunci spasi yang baru. Kemudian karakter *plaintext* dijumlahkan dengan pasangan kunci spasi baru. Proses ini akan terus diulang hingga seluruh karakter terenkripsi.
   g. Setelah tahapan enkripsi awal dilakukan, selanjutnya dilakukan proses enkripsi kedua, yaitu operasi XOR antara hasil enkripsi pertama dengan *binary* ASCII dari kunci spasi.
2. Alur Dekripsi Teks
   a. Pembuka pesan memilih pesan yang akan dibuka serta memasukkan kunci geseran dan kunci spasi yang telah ditentukan oleh pembuat pesan.
   b. Setelah selesai, sistem akan melakukan operasi XOR antara hasil enkripsi kedua (*chipertext* yang sudah dalam bentuk *binary*) dengan *binary* ASCII dari kunci spasi.
   c. Selanjutnya *chipertext* yang sudah dilakukan operasi XOR tersebut akan didekripsi dan dilakukan operasi pengurangan dengan kunci spasi dalam bentuk binary ASCII 8 bit kemudian dibagi dengan 2 (dua). Proses ini akan terus diulang sampai seluruh karakter terdekripsi.
   d. Setelah itu, setiap huruf dalam *chipertext* akan dirubah kembali ke dalam bentuk teks sesungguhnya (*plaintext*) dengan sebelumnya dilakukan pergeseran kembali (searah dengan jarum jam) sesuai dengan kunci geseran yang telah dimasukkan oleh pembuka pesan.

3. Metodologi

3.1. Metodologi Iterasi (*Iterative*)

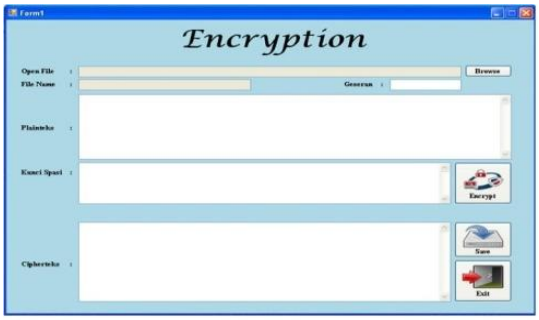
Metodologi yang digunakan dalam pengembangan sistem ini adalah metode iterasi (*iterative).* Proses pengembangan berulang (*iterative*) memerlukan penyelesaian analisis,

desain, dan implementasi karena penting untuk mengembangkan satu bagian sistem baru secara menyeluruh dan menempatkan ke dalam operasi secepat mungkin (Whitten 2004, hal 36). Diagram aktivitas yang terdapat pada sistem adalah sebagai berikut:

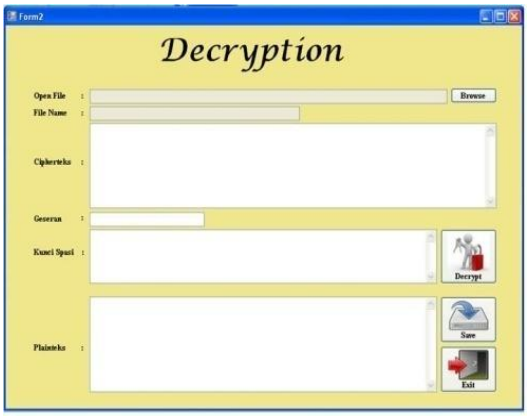
4. Hasil dan Pembahasan

4.1. Rancangan Antarmuka

Rancangan antarmuka (*interface*) merupakan tampilan layar yang didesain dari sistem yang dibangun. Rancangan *interface* digunakan sebagai penghubung antar *user* dengan system, sehingga *user* lebih mudah dan nyaman dalam mengakses sistem.



Gambar 1. Form *Encryption*



Gambar 1. Form *Decryption*

4.2. Pengujian Kunci Geseran

Pada pengujian kunci geseran, didapatkan bahwa *plaintext* hanya dapat menerima kunci geseran minimal 0 dan maksimal 26 geseran karena kunci geseran ini dibuat dengan mengacu pada rumus yang digunakan pada Algoritma Julius Caesar. Tetapi apabila pengguna tidak memasukkan kunci geseran, maka sistem akan mengganggap bahwa pengguna menggunakan kunci geseran 0 atau dengan kata lain pengguna tidak bermaksud untuk menggeser teks tersebut saat dilakukan proses enkripsi. Tetapi apabila pengguna menginput lebih dari 26 geseran maka sistem akan mengubahnya menjadi kunci geseran diantara 1 sampai 26 dengan cara di-*mod* kunci geseran tersebut. Untuk tingkat keamanan pada kunci geseran ini, semakin besar kunci geseran yang dilakukan maka akan semakin rumit hasil *ciphertext* yang dihasilkan.

4.3. Pengujian Kunci Spasi

Pada pengujian kunci spasi, kunci spasi didapatkan dari lokasi spasi teks tersebut. Dimana semakin panjang teks yang akan dienkripsi dan semakin banyak spasi yang digunakan pada teks tersebut maka tingkat keamanan pada teks tersebut pun semakin kuat. Dari hasil pengujian didapatkan pula, jika karakter yang akan dienkripsi pada *plaintext* hanya 1 (satu) karakter maka pada *ciphertext* akan dihasilkan 8 (delapan) karakter berupa biner dan jika karakter yang akan dienkripsi pada *plaintext* hanya 5 (lima) karakter maka pada *ciphertext* akan dihasilkan 40 (empat puluh) karakter berupa biner dan jika karakter yang akan dienkripsi pada *plaintext* hanya 10 (sepuluh) karakter maka pada *ciphertext* akan dihasilkan 80 (delapan puluh) karakter berupa biner dan jika karakter yang akan dienkripsi pada *plaintext* hanya 15 (lima belas) karakter maka pada *ciphertext* akan dihasilkan 120 (seratus dua puluh) karakter berupa biner dan jika karakter yang akan dienkripsi pada *plaintext* hanya 20 (dua puluh) karakter maka pada *ciphertext* akan dihasilkan 160 (seratus enam puluh) karakter berupa biner dan seterusnya dengan kelipatan 8 (delapan) karakter untuk 1 (satu) karakter yang akan dienkripsi.

4.4. Pengujian Kecepatan Proses Enkripsi dan Dekripsi

Bagi suatu algoritma, kecepatan dari proses enkripsi dan dekripsi sangat diutamakan karena faktor penentu dari suatu algoritma dikatakan bagus atau tidak terletak pada kemampuan kecepatannya Semakin cepat suatu algoritma dalam mengenkripsi dan mendekripsi teks, maka semakin bagus pula algoritma yang digunakan. Pada algoritma *Improved Caesar Cipher* yang penulis gunakan ini, kecepetan proses enkripsi dan dekripsi nya sudah termasuk cukup cepat karena sebagian besar waktunya digunakan untuk operasi pembentukan kunci dimana melakukan 2 kali pembentukan kunci, yaitu pada saat pembentukan kunci geseran dan kunci spasi, serta melakukan beberapa kali proses enkripsi dan dekripsi yang memerlukan waktu yang relatif sedikit lama.

5. Kesimpulan

Berdasarkan hasil penelitian yang telah penulis lakukan menggunakan Algoritma Improved Caesar Cipher, dapat ditarik beberapa kesimpulan, yaitu:

1. Aplikasi ini berhasil mengenkripsi dan mendekripsi teks berupa isi dari file .txt. File yang telah dienkripsi berhasil teracak sehingga file tersebut tidak bisa dimengerti oleh pihak lain, dan hasil dekripsi teks tersebut sama dengan file asli sebelum dienkripsi.
2. Penerapan Algoritma Improved Caesar Cipher untuk enkripsi dan dekripsi teks ini dapat meningkatkan keamanan teks. Teks yang terenkripsi tidak dapat dimengerti maknanya jika tidak didekripsi menggunakan kunci geseran dan kunci spasi yang benar. Sehingga hanya penerima yang berhak yang dapat membacanya.
3. Semakin besar kunci geseran dan semakin banyak kunci spasi yang digunakan untuk mengenkripsi teks, maka semakin sulit bagi pihak ketiga untuk memecahkan atau mengetahui isi dari teks tersebut (mendekripsi teks)

6. Refeerensi

[1] Ariyus, Dony 2006, *Kriptografi : Keamanan Data dan Komunikasi*, Graha Ilmu, Yogyakarta.
[2] Ariyus, Dony 2006, *Computer Security*, Andi, Yogyakarta.
[3] Ariyus, Dony 2009, *Keamanan Multimedia*, Andi, Yogyakarta.
[4] Komputer, Wahana 2008, *Cepat Menguasai Visual Studio .Net 2008 Express*, Andi Offset, Yogyakarta.
[5] Komputer, Wahana 2010, *Membuat Aplikasi Client Server dengan Visual Basic 2008*, Andi Offset, Yogyakarta.
[6] Kurniawan, Yusuf 2004, *Kriptografi : Keamanan Internet dan Jaringan Telekomunikasi*, Informatika, Bandung.
[7] Munir, Rinaldi 2006, *Kriptografi*, Informatika, Bandung.
[8] A.S., Rosa., Shalahuddin M. 2011, *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*, Modula, Bandung.
[9] Simarmata, Janner 2006, *Pengamanan Sistem Komputer*, Andi Offset, Yogyakarta.