METODE *SYMMETRIC ENCRYPTION SUBSTITUTION CIPHER* UNTUK *MULTIPLE* FORMAT DATABASE

Suyanto [1], Dedi Rianto Rahadi [2] , Ahmad Luthfi[3]
Magister Teknik Informatika, Universitas Bina Darma
Jl. Ahmad Yani No.12 Palembang
email: yantox_ska@yahoo.com [1], dedi1968@yahoo.com [2], praboe007@yahoo.com[3]

ABSTRAK

Pertumbuhan teknologi pengolahan data, telah mengakibatkan dalam pelaksanaan sistem komputerisasi di segala bidang. Seluruh bagian dari organisasi atau perusahaan telah menerapkan berbagai aplikasi untuk memenuhi kebutuhan mereka. Aplikasi yang diterapkan secara terpisah menimbulkan masalah sendiri yang memerlukan data yang sama. Sehingga pertukaran data menjadi solusi alternatif. Persyaratan keamanan data pada proses pertukaran database menjadi masalah baru dalam proses. Untuk menjamin keamanan data pada proses pertukaran, metode enkripsi adalah cara terbaik. Dengan penerapan enkripsi pada data yang dipertukarkan, maka data dijamin aman dan hanya dapat dibaca oleh mereka yang membutuhkannya. Aplikasi yang menggunakan database berbeda membutuhkan format standar sehingga semua aplikasi dapat membacanya. Format ini adalah XML. Dengan XML, database (tabel) dapat dibaca oleh aplikasi lain meskipun dengan platform yang berbeda.

Kata Kunci: enkripsi, simetrik, substitusi, cipher, XML.

1 PENDAHULUAN

Pemanfaatan aplikasi komputer dalam suatu perusahaan atau organisasi dewasa ini sudah merupakan suatu kebutuhan. Hampir di semua bidang pekerjaan menggunakan alat bantu berupa aplikasi komputer. Dalam satu perusahaan misalnya, aplikasi komputer sering dibuat sendiri oleh team komputer dari perusahaan tersebut.

Pada perusahaan menengah ke bawah, aplikasi yang dibuat kebanyakan masih merupakan aplikasi yang terpisah-pisah. Hal ini banyak disebabkan karena pengembangan sistem yang tidak terencana dengan baik atau dikarenakan pembuatan aplikasi sering merupakan penyelesaian jangka pendek. Artinya, saat suatu bagian membutuhkan aplikasi maka dibuatlah aplikasi tersebut begitu seterusnya sehingga banyak aplikasi yang jalan sendiri-sendiri.

Dengan berkembangnya pengolahan data, tidak jarang satu aplikasi menggunakan database yang sama untuk aplikasi yang lainnya. Jalan yang bisa ditempuh adalah dengan cara mengkopi database yang dibutuhkan tersebut dari satu aplikasi ke aplikasi yang lain. Hal ini merupakan jalan pintas, agar aplikasi yang membutuhkan data tadi bisa dioperasikan. Begitu seterusnya sehingga kegiatan kopi database merupakan suatu kebutuhan dari aplikasi tersebut.

Melihat kondisi diatas, maka keamanan data menjadi suatu hal yang sangat penting. Karena bukan hal yang mustahil bahwa data yang dipertukarkan akan disalahgunakan oleh pihak-pihak yang tidak bertanggung jawab untuk memenuhi kebutuhan pribadi mereka. Untuk itu perlu teknik khusus untuk mengamankan database yang dipertukarkan tersebut.

Keamanan sistem dibagi menjadi tiga bagian : 1) Keamanan eksternal Keamanan eksternal berkaitan dengan fasilitas komputer dari penyusup dan bencana seperti kebakaran atau bencana alam. 2) Keamanan *interface* pamakai Keamanan *interface* pemakai yang berkaitan dengan identifikasi pemakai sebelum pemakai diizinkan mengakses data atau program. 3. Keamanan internal Keamanan internal berkaitan dengan beragam kendali yang dibangun pada perangkat keras dan perangkat lunak yang menjamin operasi yang handal dan tidak terganngu untuk menjaga integritas data.

Kebutuhan keamanan sistem komputer dapat dikategorikan menjadi aspek-aspek sebagai berikut. *Privacy / Confidentiality* Inti utama aspek *privacy* atau *confidentiality* adalah usaha untuk menjaga informasi dari orang yang tidak berhak mengakses. *Privacy* lebih kearah data-data yang sifatnya *private* sedangkan *confidentiality* biasanya berhubungan dengan data yang diberikan ke pihak lain untuk keperluan tertentu (misalnya sebagai bagian dari pendaftaran sebuah servis) dan hanya diperbolehkan untuk keperluan tertentu tersebut. 2. *Integrity* Aspek ini menekankan bahwa informasi tidak boleh diubah tanpa seijin pemilik informasi. Adanya virus, trojan horse, atau pemakai lain yang mengubah informasi tanpa ijin merupakan contoh

ISSN 2088-6519

masalah yang harus dihadapi. Sebuah e-mail dapat saja "ditangkap" (*intercept*) di tengah jalan, diubah isinya (*altered, tampered, modified*), kemudian diteruskan ke alamat yang dituju. Dengan kata lain, integritas dari informasi sudah tidak terjaga. Penggunaan *encryption* dan *digital signature*, misalnya, dapat mengatasi masalah ini. 3.*Authentication* Aspek ini berhubungan dengan metoda untuk menyatakan bahwa informasi betul-betul asli, orang yang mengakses atau memberikan informasi adalah betul-betul orang yang dimaksud, atau *server* yang kita hubungi adalah betul-betul *server* yang asli. 4. *Availability* Aspek *availability* atau ketersediaan berhubungan dengan ketersediaan informasi ketika dibutuhkan. Sistem informasi yang diserang atau dijebol dapat menghambat atau meniadakan akses ke informasi.

Salah satu mekanisme untuk meningkatkan keamanan data dalam database adalah dengan menggunakan teknologi enkripsi. Data-data yang disimpan dalam database diubah sedemikian rupa sehingga tidak mudah dibaca. Jadi enkripsi adalah proses yang dilakukan untuk mengamankan sebuah data (yang disebut *plaintext*) menjadi data yang tersembunyi (disebut *ciphertext*). *Ciphertext* adalah data yang sudah tidak dapat dibaca dengan mudah. Menurut ISO 7498-2, terminologi yang lebih tepat digunakan adalah "*encipher*". Proses sebaliknya, untuk mengubah *ciphertext* menjadi *plaintext*, disebut dekripsi (*decryption*). Menurut ISO 7498-2, terminologi yang lebih tepat untuk proses ini adalah "*decipher*".

Pengetahuan yang mempelajari tentang enkripsi adalah kriptografi. Yang dimaksud dengan kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan data, keabsahan data, integritas data, serta autentikasi data. Sedangkan menurut Prasetyo[6] Kriptografi adalah suatu ilmu pengetahuan (atau lebih tepatnya suatu seni) yang mengenkripsi informasi sehingga sepenuhnya terlihat berbeda dari aslinya. Dalam Kriptografi, harus ada cara bagi user untuk mendekripsi informasi yang terenkripsi dan memperoleh kembali informasi aslinya. Dimana seharusnya hanya ada satu user yang dapat melakukan hal ini, yaitu user yang memegang kunci (*key*) untuk mengenkripsi.

Enkripsi diperlukan terutama dikarenakan adanya kebutuhan untuk pertukaran data dari satu database ke database lain atau adanya kebutuhan pengiriman data. Dengan data yang terenkripsi maka apabila sampai data tersebut disadap orang atau dibuka orang yang tidak tepat maka dia tidak akan bisa menemukan data apapun.

Untuk kebutuhan pertukaran data, maka data yang dipertukarkan diusahakan merupakan data yang berukuran kecil dan mampu dibaca oleh aplikasi ataupun sistem operasi lain. Karena tidak menutup kemungkinan bahwa aplikasi yang menggunakan data tersebut berbeda platformnya. Format yang tepat untuk hal tersebut diatas adalah dengan membuat data tersebut dalam format XML (*Extensible Markup Language*).

XML untuk saat ini bukan merupakan pengganti HTML. Masing-masing dikembangkan untuk tujuan yang berbeda. Kalau HTML digunakan untuk menampilkan informasi dan berfokus pada bagaimana informasi terlihat, XML mendeskripsikan susunan informasi dan berfokus pada informasi itu sendiri. XML terutama dibutuhkan untuk menyusun dan menyajikan informasi dengan format yang tidak mengandung format standard layaknya heading, paragraph, table dan lain sebagainya.

XML adalah sebuah bahasa markup yang digunakan untuk mengolah meta data (informasi tentang data) yang menggambarkan struktur dan maksud/tujuan data yang terdapat dalam dokumen XML, namun bukan menggambarkan format tampilan data tersebut. XML adalah sebuah standar sederhana yang digunakan untuk medeskripsikan data teks dengan cara *self-describing* (deskripsi diri). XML juga dapat digunakan untuk mendefinisikan domain tertentu lainnya, seperti musik, matematika, keuangan dan lain-lain yang menggunakan bahasa markup terstruktur.

2 ANALISIS DAN DESAINN

Berdasarkan cara memproses teks (*plaintext*), *cipher* dapat dikategorikan menjadi dua jenis: *block cipher* dan *stream cipher*. *Block cipher* bekerja dengan memproses data secara blok, dimana beberapa karakter / data digabungkan menjadi satu blok. Setiap proses satu blok menghasilkan keluaran satu blok juga. Sementara itu *stream cipher* bekerja memproses masukan (karakter atau data) secara terus menerus dan menghasilkan data pada saat yang bersamaan.

Terdapat beberapa cara untuk melakukan enkripsi yaitu:

2.1 Enkripsi Simetris (*Symmetric Encryption*)

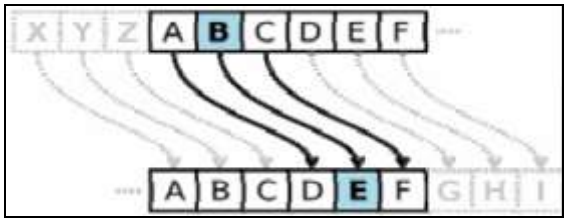
Enkripsi simetris (*symmetric encryption*). Algoritma ini menggunakan sebuah kunci rahasia yang sama (*private key*) untuk melakukan proses enkripsi dan dekripsinya. Algoritma ini termasuk

ISSN 2088-6519

dalam algoritma kriptografi klasik yang terdiri dari: s*ubstitution cipher* dan *transposition cipher*.

2.1.1 *Substitution Cipher*

Cara kerja dari algoritma s*ubstitution cipher* ini adalah dengan menggantikan setiap karakter dari *plaintext* dengan karakter lain. Algoritma ini pertama kali digunakan oleh Julius Caesar, dan disebut juga sebagai *shift cipher*, yaitu dengan cara menggeser urutan abjadnya. Substitusi ini kadang dikenal dengan c3 (untuk caesar menggeser 3 tempat).



Gambar 1. Algoritma *Caesar Chiper*

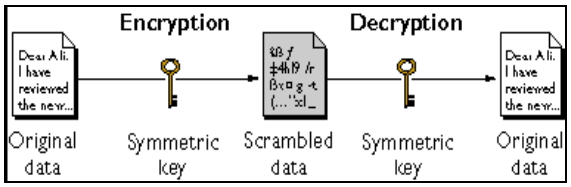
Secara umum sistem *cipher* caesar dapat ditulis sebagai berikut:

$$Zi = Cn(Pi)$$

Dimana Zi adalah karakter-karekter *ciphertext*, Cn adalah transformasi substitusi alfabetik, n adalah jumlah huruf yang digeser, dan pi adalah karakter-karakter plaintext (Kelompok 124 IKI-83408 MTI UI, 2005).

2.1.2 Transposition Cipher

Sedangkan algoritma *transposition cipher* diperoleh dengan mengubah posisi *plaintext*nya. Dengan kata lain, algoritma ini melakukan *transpose* terhadap rangkaian karakter di dalam teks. Nama lain untuk metode ini adalah permutasi, karena *transpose* setiap karakter di dalam teks sama dengan mempermutasikan karakter-karakter tersebut.



Gambar 2. Enkripsi Simetris

2.2 Enkripsi Asimetris (*Asymmetric Encryption*)

Sedangkan pada sistem kunci-asimentris digunakan sepasang kunci yang berbeda, umumnya disebut kunci public (*public key*) dan kunci pribadi (*private key*), digunakan untuk proses enkripsi dan proses dekripsinya. Bila *plaintext* dienkripsi dengan menggunakan kunci pribadi maka *ciphertext* yang dihasilkannya hanya bisa didekripsikan dengan menggunakan pasangan kunci pribadinya. Begitu juga sebaliknya, jika kunci pribadi digunakan untuk proses enkripsi maka proses dekripsi harus menggunakan kunci publik pasangannya.

Asymmetric Encryption memiliki kelemahan dan kelebihan sebagai berikut:

Kelebihan:

1) Hanya *Private key* yang harus benar-benar rahasia/aman.
2) Sangat jarang untuk perlu merubah *public key* dan *private key.*

Kelemahan:

1) Ukuran kunci lebih besar dari pada *symmetric encryption*.
2) Tidak adanya jaminan bahwa public key benar-benar aman.

2.3 Fungsi *Hash*

Fungsi *Hash* ini sering juga disebut sebagai fungsi *hash kriptografis*, yaitu fungsi yang secara efisien mengubah string input dengan panjang berhingga menjadi string output dengan panjang tetap yang disebut nilai *hash*. Fungsi ini bersifat satu arah sehingga inputan yang telah dienkripsi tidak dapat dibalikkan atau didekripsikan. Contohnya adalah penggunaan MD5 untuk melindungi *password*.

2.4 Multiple Format Database

Guna mendukung bermacam-macam format database, maka hasil enkripsi kemudian digenerate menjadi format XML. Dalam buku NIIT yang berjudul "*Introduction to Web Content Development"* menjelaskan tentang XML sebagai berikut :

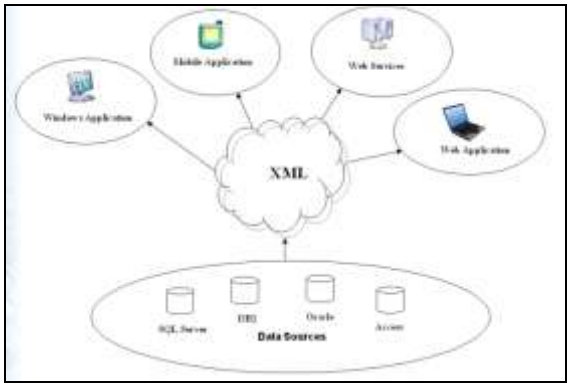
1) XML is a text-based markup language that enables storage of data in a structured format.
2) XML is a cross-platform, hardware and software independent markup language that enables structured data transfer between heterogeneous systems.
3) XML is used as a common data interchange format in a number of applications.

Karena XML bersifat mudah untuk dibaca dan ditulis baik oleh manusia maupun komputer, maka XML merupakan sebuah format yang dapat digunakan untuk pertukaran data (*interchange*) antar aplikasi dan platform yang berbeda (*platform independent*). Metode deskripsi data XML (*self-*

ISSN 2088-6519

describing) membuatnya menjadi pilihan efektif untuk bisnis ke bisnis, solusi antar jaringan, *e-business*, dan aplikasi terdistribusi. XML juga bersifat dapat diperluas (*extensible*), dapat digunakan pada semua bahasa pemrograman, dan datanya dapat ditransfer dengan mudah melalui protokol standar internet seperti HTTP tanpa dibatasi oleh *firewall*.

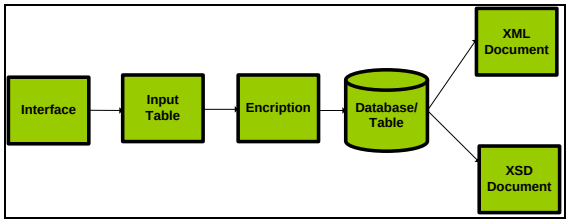
Kelebihan lain yang dimiliki XML adalah bahwa informasi bisa di pertukarkan dari satu system ke system lain yang berbeda *platform*. Misalnya dari Windows ke Unix, atau dari PC ke Machintosh bahkan dari internet ke handphone dengan teknologi WAP.



Gambar 3. Penggunaan XML Dokumen

3 KERANGKA PIKIR DAN HASIL RANCANGAN

Kerangka pikir pengembangan perangkat lunak yang akan dibangun dapat digambarkan sebagai berikut :



Gambar 4. Kerangka Pikir Pengembangan

Dari kerangka pikir dan uraian serta analisis diatas maka dihasilkan rancangan interface, rancangan algoritma enkripsi dan dekripsi sebagai berikut :

3.1.1 Rancangan Interface

Rancangan interface dari perangkat lunak enkripsi sebagai berikut :

Gambar 5. Rancangan *Interface*

Dari tampilan interface diatas, dapat dijelaskan logika programnya sebagai berikut:

1) Pertama-tama kita tentukan nama file database yang akan dienkripsi dengan cara menekan tombol *Browse*. Dengan menekan tombol tersebut maka program akan membawa ke menu browse untuk memilih file sesuai letak folder yang ada.
2) Setelah menentukan nama file database yang benar, maka pada menu daftar tabel (*Table List*) akan muncul nama-nama tabel dari database yang dipilih. Langkah berikutnya adalah memilih tabel yang akan dienkripsi.
3) Kemudian menentukan nilai pergeseran karakter yang akan diterapkan dalam enkripsi yaitu dengan menentukan nilai Cn dengan mengklik spin sesuai keinginan user.
4) Setelah menentukan nilai n (Cn) maka langkah berikutnya adalah melakukan proses Enkripsi dengan cara menekan tombol *Encrypt*. Dari proses enkripsi tersebut akan dihasilkan satu file tabel baru yang berisi data-data dari tabel yang telah ditentukan sebelumnya yang sudah dalam keadaan terenkripsi.
5) Langkah berikutnya adalah melakukan *generate* untuk menghasilkan file XML dan atau file XSD yang nantinya bisa dipertukarkan dengan atau untuk aplikasi lain yang membutuhkan data yang sama tersebut. Sebelum mengklik tombol *Generate* maka tentukan terlebih dahulu lokasi penyimpanan untuk file XML dan atau XSD tersebut diatas, baru setelah itu klik tombol *Generate*.
6) Selesai.

3.1.2 Rancangan Algoritma Enkripsi

Algoritma enkripsi yang digunakan adalah enkripsi simetris subtitusi, yaitu dengan menggeser

ISSN 2088-6519

karakter sebanyak n. Nilai n ini bisa diubah-ubah sesuai kebutuhan karena n disini bersifat parametris.

Sedangkan algoritma enkripsi simetris subtitusinya sebagai berikut :

Algoritma ESubstitutionChiper(plaintext,n)

1) Baca plaintext
2) Ulang sebanyak panjang karakter plaintext
   a. Baca karakter demi karakter
   b. Convert karakter ke nilai integer
   c. Tambah dengan nilai n
   d. Kembalikan ke karakter

3.1.3 Rancangan Algoritma Dekripsi

Algoritma yang digunakan adalah simetris enkripsi, sehingga kunci yang digunakan dalam melakukan enkrispi dan dekripsi adalah sama. Oleh karena itu pada algoritma dekripsi ini, harus menggunakan nilai n yang sama dengan proses enkripsi agar data yang dihasilkan merupakan data yang sesuai dengan data asalnya.

Dan algoritma Dekripsinya dijelaskan sebagai berikut :

Algoritma DSubstitutionChiper(plaintext,n)

1) Baca plaintext hasil enkripsi
2) Ulang sebanyak panjang karakter plaintext
   a. Baca karakter demi karakter
   b. Convert karakter ke nilai integer
   c. Kurangi dengan nilai n
   d. Kembalikan ke karakter

Kedua algoritma diatas merupakan algoritma makro, artinya algoritma yang dibuat diatas hanya menjelaskan pokok-pokok langkah yang akan dikerjakan dengan kata lain hanya ditulis langkah-langkah globalnya saja.

4 KESIMPULAN

Dari hasil rancangan diatas, maka diperoleh kesimpulan sebagai berikut :

1) Dihasilkan rancangan berupa *tools* untuk mengenkripsi *database* (tabel)
2) Hasil enkripsi berupa file XML yang mempunyai format standar database sehingga bisa dipertukarkan dari satu system ke system lain yang berbeda *platform*. Misalnya dari Windows ke Unix, atau dari PC ke Machintosh bahkan dari internet ke handphone dengan teknologi WAP.
3) File XML aman untuk dipertukarkan karena sudah dalam bentuk enkripsi.

REFERENSI

[1] Hariyanto, Bambang. 1997. *Sistem Operasi, Jilid I*. Informatika Bandung.

[2] Kelompok 124 IKI-83408 MTI UI. 2005. *Cryptography*, http://bebas.vlsm.org/v06/Kuliah/MTI-Keamanan-Sistem-Informasi/2005/124/124P-04-draft-Cryptography.pdf. Diakses Tanggal 20 Nopember 2010

[3] NIIT. *Introduction to Web Content Development.* India

[4] Prasetyo, Didik Dwi. 2006. *Pemgr Apl. Database Vb.net 2005 + Cd*. Jakarta Elex Media Komputindo.

[5] Rahardjo, Budi. 2002. *Keamanan Sistem informasi Berbasis Internet*. Bandung : PT Insan Komunikasi Indonesia.

[6] Suryana, Aulya dkk. 2007. *Enkripsi*. Sekolah Tinggi Manajemen Informatika Dan Teknik Komputer Bali (STMIK) – STIKOM.

ISSN 2088-6519