STUDI DAN IMPLEMENTASI PENGAMANAN BASIS DATA MENGGUNAKAN METODE ENKRIPSI MD5 (Message-Digest Algorihm 5)

Saipul Bahri [1] , Diana [2] , Susan Dian PS[3]
Mahasiswa Universitas Bina Darma, Dosen Universitas Bina Darma, Dosen Universitas Bina Darma
Jalan Jenderal Ahmad Yani No.12 Palembang
Pos-el : saipul_bahri1989@yahoo.com, your2@email[2], your3@email[3]

Absrak
Security problem is one of the challenges to be met in the industry and the research database. Data stored in the database must be secured. Data security can be done in two ways. The first way is setting the permissions of each user by the database administrator. The second way is to secure data from the content data stored in the database. This paper describes the implementation of the security of the data in two ways. Security of data is done using cryptographic techniques md5. Study (study) is done is to find ways to make md5 can be utilized to secure the data and provide convenience for the owner of the data to secure data without needing to know the query - the query that need to be typed or executed. Security of conventional encryption depends on several factors. The first encryption algorithm must be robust enough so that makes it very difficult to decrypt the cipher text with the basic cipher text. Furthermore the security of conventional encryption algorithms rely on the secrecy of the key instead of the algorithm, ie assuming that is not very practical to decrypt the cipher text with basic information and knowledge of the algorithm encryption / description. Or in other words, we do not need to maintain the secrecy of the algorithm, but enough with the secrecy of the key.

Keyword: data security, md5, cipher text

Absrak
Masalah keamanan merupakan salah satu tantangan yang harus dipenuhi di dalam industri dan penelitian basis data. Data yang tersimpan di dalam basis data harus dapat terjamin keamanannya. Pengamanan data dapat dilakukan melalui dua cara. Cara pertama ialah pengaturan hak akses setiap pengguna oleh administrator basis data. Cara kedua ialah pengamanan data dari sisi kandungan data yang tersimpan pada basis data. Makalah ini menguraikan implementasi pengamanan data pada basis data dengan cara kedua. Pengamanan data dilakukan dengan menggunakan teknik kriptografi md5. Penelitian (studi) yang dilakukan ialah untuk mencari cara agar md5 dapat dimanfaatkan untuk mengamankan data serta memberi kemudahan bagi pemilik data untuk mengamankan datanya tanpa perlu mengetahui *query – query* yang perlu diketikkan atau dijalankan.
Keamanan dari enkripsi konvensional bergantung pada beberapa faktor. Pertama algoritma enkripsi harus cukup kuat sehingga menjadikan sangat sulit untuk mendekripsi *cipher teks* dengan dasar *cipher teks* tersebut. Lebih jauh dari itu keamanan dari algoritma enkripsi konvensional bergantung pada kerahasian dari kuncinya bukan algoritmanya, yaitu dengan asumsi bahwa adalah sangat tidak praktis untuk mendekripsikan informasi dengan dasar cipher teks dan pengetahuan tentang algoritma enkripsi / diskripsi. Atau dengan kata lain, kita tidak perlu menjaga kerahasiaan dari algoritma tetapi cukup dengan kerahasiaan kuncinya.

Keyword: keamanan data, md5, *cipher teks*

1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi (TIK) di era globalisasi ini seolah tidak dapat dibendung lagi dalam sisi kehidupan manusia di abad ke-21 ini. Cepatnya pergerakan TIK ini dapat diamati secara jelas pada bidang bisnis, ekonomi dan juga pemerintahan dengan munculnya konsep dan aplikasi berupa *e-goverment, e-commerce, e-community* dan lain sebagainya. Penggunaan komputer dapat menghasilkan pengolahan data yang lebih akurat dan pencarian data yang lebih cepat pada teknologi berkembang saat ini yang menjadikan sistem bagian yang tidak kalah pentingnya dalam perusahaan dan perkantoran, penerapan teknologi dan sistem informasi pada perusahaan dapat menjadi teknologi yang tepat guna. Dalam mengolah data menjadi informasi yang mendukung pengambilan keputusan yang tepat dan dapat memberikan keunggulan baik, sehingga mendapat prioritas yang tinggi dalam mendukung pelaksanaan operasional perusahaan.

Salah satu hal yang penting dalam komunikasi menggunakan *computer* untuk menjamin kerahasian data adalah enkripsi. Enkripsi adalah sebuah proses yang melakukan perubahan sebuah kode dari yang bisa dimengerti menjadi sebuah kode yang tidak bisa dimengerti (tidak terbaca). Enkripsi dapat diartikan sebagai kode atau *chiper*. Sebuah sistem pengkodean menggunakan suatu *table* atau kamus yang telah didefinisikan untuk mengganti kata dari informasi atau yang merupakan bagian dari informasi yang dikirim. Sebuah *chiper* menggunakan suatu algoritma yang dapat mengkodekan semua aliran data (*stream*) *bit* dari sebuah pesan menjadi *cryptogram* yang tidak dimengerti (*unitelligible*). Karena teknik *cipher* merupakan suatu sistem yang telah siap untuk di automasi, maka teknik ini digunakan dalam sistem keamanan komputer dan *network*.

Keamanan dari enkripsi konvensional bergantung pada beberapa faktor. Pertama algoritma enkripsi harus cukup kuat sehingga menjadikan sangat sulit untuk mendekripsi *cipher teks* dengan dasar *cipher* teks tersebut. Lebih jauh dari itu keamanan dari algoritma enkripsi konvensional bergantung pada kerahasian dari kuncinya bukan algoritmanya, yaitu dengan asumsi bahwa adalah sangat tidak praktis untuk mendekripsikan informasi dengan dasar *cipher teks* dan pengetahuan tentang algoritma diskripsi / enkripsi. Atau dengan kata lain, kita tidak perlu menjaga kerahasiaan dari algoritma tetapi cukup dengan kerahasiaan kuncinya.

2. TINJAUAN UMUM

2.1. Keamanan Data

Aspek keamanan data sebenarya meliputi banyak hal yang saling berkaitan, tetapi khusus dalam tulisan ini penulis akan membahas tentang metoda enkripsi dan keamanan proteksi data pada beberapa program-program aplikasi umum. Hampir semua program aplikasi seperti

MS Word, WordPerfect, Excel, PKZip menyediakan fasilitas proteksi data dengan pem-password-an, tapi sebenarnya fasilitas ini mudah untuk dibongkar. Bahkan program khusus proteksi data seperti *Norton Diskreet* (mungkin sekarang sudah jarang digunakan) yang memproteksi data dengan metoda DES ataupun metoda "proprietary" yang lebih cepat, sebenarnya sangat tidak aman. Metoda DES yang digunakan mempunyai kesalahan dalam implementasinya yang sangat mengurangi keefektipan dari metoda tersebut. Walaupun dapat menerima password sampai 40 karakter, karakter ini kemudian diubah menjadi huruf besar semua dan kemudian di-reduce menjadi 8 karakter. Hal ini menyebabkan pengurangan yang sangat besar terhadap kemungkinan jumlah kunci enkripsi, sehingga tidak hanya terbatasnya jumlah password yang mungkin, tetapi juga ada sejumlah besar kunci yang equivalen yang dapat digunakan untuk mendekrip file. Sebagai contoh file yang dienkrip dengan kunci 'xxxxxxx' dapat didekrip dengan 'xxxxxx', 'xxxxyy', 'yyyyxx'. PC Tools (mungkin ini juga sudah sulit ditemukan) adalah contoh lain paket software yang menyediakan fasilitas proteksi data yang sangat tidak aman. Implementasi DES pada program ini mengurangi 'round' pada DES yang seharusnya 16 menjadi 2, yang membuatnya sangat mudah untuk dibongkar.

2.2. Kriptografi

Kriptografi berasal dari bahasa Yunani yakni *kriptos* yang artinya tersembunyi dan *graphia* yang artinya sesuatu yang tertulis, sehingga kriptografi dapat disebut sebagai sesuatu yang tertulis secara rahasia.

Kriptografi merupakan suatu bidang ilmu yang mempelajari tentang bagaimana merahasiakan suatu informasi penting ke dalam suatu bentuk yang tidak dapat dibaca oleh siapapun serta mengembalikannya kembali menjadi informasi semula dengan menggunakan berbagai macam teknik yang telah ada sehingga informasi tersebut tidak dapat diketahui oleh pihak manapun yang bukan pemilik atau yang tidak berkepentingan. Sisi lain dari kriptografi ialah kriptanalisis (*Cryptanalysis*) yang merupakan studi tentang bagaimana memecahkan mekanisme kriptografi.

Bagi kebanyakan orang, kriptografi lebih diutamakan dalam menjaga komunikasi tetap rahasia dan khusus. Seperti yang telah diketahui dan disetujui bahwa perlindungan (proteksi) terhadap komunikasi yang sensitif telah menjadi penekanan kriptografi selama ini. Akan tetapi hal tersebut hanyalah sebagian dari penerapan kriptografi dewasa ini. **Munir (2006:2).**

2.3. Metode Enkripsi MD5

MD5 yang merupakan singkatan dari ***Message-Digest algortihm 5,*** adalah fungsi *hash* (prosedur terdefinisi atau fungsi matematika

yang mengubah variabel dari suatu data yang berukuran besar menjadi lebih sederhana) kriptografik yang digunakan secara luas dengan hash value 128-bit. MD5 dimanfaatkan dalam berbagai aplikasi keamanan, dan umumnya digunakan untuk meguji integritas sebuah *file*. Enkripsi menggunakan MD5 masih mendominasi sebagian besar aplikasi PHP. Enkripsi MD5 dianggap strong karena enkripsi yang dihasilkannya bersifat '*one way hash*'. Berapapaun string yang di enkripsi hasilnya tetap sepanjang 32 karakter.

Hash-hash MD5 sepanjang 128-bit (16-byte), yang dikenal juga sebagai ringkasan pesan, secara tipikal ditampilkan dalam bilangan heksadesimal 32-digit. Berikut ini merupakan contoh pesan ASCII sepanjang 43-byte sebagai masukan dan hash MD5 terkait:

MD5("The quick brown fox jumps over the lazy dog") =
9e107d9d372bb6826bd81d3542a419d6

Bahkan perubahan yang kecil pada pesan akan (dengan probabilitas lebih) menghasilkan hash yang benar-benar berbeda, misalnya pada kata "*dog*", huruf d diganti menjadi c:

MD5("The quick brown fox jumps over the lazy cog") =
1055d3e698d289f2af8663725127bd4b

Hash dari panjang-nol ialah:

MD5("") =
d41d8cd98f00b204e9800998ecf8427e

Ringkasan MD5 digunakan secara luas dalam dunia perangkat lunak untuk menyediakan semacam jaminan bahwa berkas yang diambil (*download*) belum terdapat perubahan. Seorang pengguna dapat membandingkan MD5 sum yang dipublikasikan dengan *checksum* dari berkas yang diambil. Dengan asumsi bahwa *checksum* yang dipublikasikan dapat dipercaya akan keasliannya, seorang pengguna dapat secara yakin bahwa berkas tersebut adalah berkas yang sama dengan berkas yang dirilis oleh para developer, jaminan perlindungan dari *Trojan Horse* dan virus komputer yang ditambahkan pada perangkat lunak. Bagaimanapun juga, seringkali kasus yang terjadi bahwa *checksum* yang dipublikasikan tidak dapat dipercaya (sebagai contoh, *checksum* didapat dari *channel* atau lokasi yang sama dengan tempat mengambil berkas), dalam hal ini MD5 hanya mampu melakukan *error-checking*. MD5 akan mengenali berkas yang didownload tidak sempurna, cacat atau tidak lengkap.

Untuk aplikasi pengujian integritas sebuah *file* atau lebih dikenal dengan istilah **MD5 *Checksum***, dapat menggunakan aplikasi desktop atau aplikasi berbasis *web* MD5 *Checksum* seperti **"MD5 *Checksum Verifier*"** dan sebagainya. *Software* semacam ini akan menghasilkan kode MD5 dari *file* yang diuji integritasnya. Selanjutnya kode MD5 ini akan digunakan untuk menguji apakah *file* tersebut

memiliki integritas ataukah tidak. Artinya jika *file* akan diberikan atau dikirimkan atau diunduh, si penerima dapat mencocokkan dengan yang diterima apakah ukuran, struktur, dan jenis file sesuai dengan yang diberikan oleh si pembuat file. Contohnya jika Anda men-download sebuah file, kemudian diberikan juga kode MD5 *Checksum*-nya, jika diperiksa (divalidasi) dengan tool seperti *MD5 Checksum Verifier*, dinyatakan *valid* atau sama dengan *file* yang diuji, maka dikatakan file tersebut tak mengalami perubahan dari pengirim hingga ke tangan Anda. (perubahan bisa terjadi karena virus dan sebagainya). Pengujian semacam ini ditujukan untuk memastikan suatu file tidak disisipi atau *corrupt* (hilangnya sebagian) atau mungkin terinveksi, baik itu karena virus, malware, atau injeksi software berbahaya lainnya. **Munir (2006:220).**

3. PROFIL OBJEK PENELITIAN

3.1. Kriptografi

Kriptografi adalah ilmu yang mempelajari teknik-teknik matematis yang berhubungan dengan aspek keamanan informasi seperti : keabsahan, integritas data, serta autentifikasi data. Kriptografi tidak berarti hanya memberikan keamanan informasi saja, namun lebih ke arah teknik-tekniknya.

Algoritma kriptografi berkembang terus dan terbagi atas dua bagian yaitu algoritma kriptografi klasik dan modern. Pada kriptografi klasik, kriptografer menggunakan algoritma sederhana, yang memungkinkan cipherteks dapat dipecahkan dengan mudah (melalui penggunaan statistik, terkaan, intuisi, dan sebagainya). Algoritma kriptografi modern dibuat sedemikian kompleks sehingga kriptanalis sangat sulit untuk memecahkan cipherteks tanpa mengetahui kunci. Algoritma kriptografi modern umumnya beroperasi dalam mode bit. Algoritma ini dapat dikelompokkan menjadi dua kategori yaitu cipher aliran (stream cipher – beroperasi dalam bentuk bit tunggal) dan cipher blok (block cipher – beroperasi dalam bentuk blok bit). Pengelompokan algoritma juga dilakukan berdasarkan kunci enkripsi – dekripsi yang digunakan, yaitu simetris (menggunakan kunci yang sama untuk proses enkripsi – dekripsi) dan asimetris atau kunci – publik (menggunakan kunci yang berbeda untuk proses enkripsi – dekripsi).

Ada empat tujuan dari ilmu kriptografi, yaitu :

1. Kerahasiaan, adalah layanan yang digunakan untuk menjaga isi dari informasi dari siapapun kecuali yang memiliki otoritas,
2. Integritas data, adalah berhubungan dengan penjagaan dari perubahan data secara tidak sah. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi data oleh pihak-pihak yang tidak berhak, antara lain menyangkut penyisipan, penghapusan, dan pensubtitusian data lain ke dalam data yang sebenarnya.
3. Autentikasi, adalah berhubungan dengan identifikasi, baik secara kesatuan sistem

maupun informasi itu sendiri. Dua pihak yang saling berkomunikasi harus saling memperkenalkan diri. Informasi yang dikirimkan melalui kanal harus diautentikasi keaslian, isi datanya, waktu pengiriman, dan lain-lain,

4. Non-repudiasi, yang berarti begitu pesan terkirim, maka tidak akan dapat dibatalkan.

3.2. Enkripsi

Proses utama dalam suatu algoritma kriptografi adalah enkripsi dan dekripsi. Enkripsi merubah sebuah *plaintext* ke dalam bentuk *ciphertext*. Pada mode ECB (*Elekctronic Codebook*), sebuah blok pada *plaintext* dienkripsi ke dalam sebuah blok *ciphertext* dengan panjang blok yang sama. Blok cipher memiliki sifat bhahwa setiap blok harus memiliki panjang yang sama (misalnya 128 bit). Namun apabila pesan yang dienkripsi memiliki panjang blok terakhir tidak tepat 128 bit, maka diperlukan mekanisme padding, yaitu penambahan bit-bit dummies untuk menggenapi menjadi panjang blok yang sesuai; biasanya padding dilakukan pada blok terakhir *plaintext.* Padding bada blok terakhir bisa dilakukan dengan berbagai macam cara, misalnya dengan penambahan bit-bit tertentu. Salah satu contoh penerapan padding dengan cara menambahkan jumlah total padding sebagai *byte* terakhir pada blok terakhir plaintext. Misalnya panjang blok adalah 128 bit (16 *byte*) dan pada blok terakhir terdiri dari 88 bit (11 *byte*) sehingga jumlah padding yang diperlukan adalah 5 *byte*, yaitu dengan menambahkan angka nol sebanyak 4 byte, kemudian menambahkan angka 5 sebanyak satu byte. Cara lain dapat juga menggunakan penambahan karakter *end-of-file* pada *byte* terakhir lalu diberi padding setelahnya.

3.3. Dekripsi

Dekripsi merupakan proses kebalikan dari proses enkripsi, merubah *ciphertext* kembali ke dalam bentuk *plaintext*. Untuk menghilangkan padding yang diberikan pada saat prpses enkripsi, dilakukan berdasarkan informasi jumlah padding yaitu angka pada *byte* terakhir.

3.4. Teknik Kriptografi

Pada umumnya terdapat dua teknik yang digunakan dalam kriptografi, yakni: kunci simetrik dan kunci asimetrik (*public-key*).

3.4.1. Kunci Simetrik

Skema enkripsi akan disebut *symmetric-key* apabila pasangan kunci untuk proses enkripsi dan dekripsinya sama. Pada skema enkripsi kunci simetrik dibedakan lagi menjadi dua kelas, yaitu block-cipher dan stream-cipher. Block-cipher adalah skema enkripsi yang akan membagi-bagi plaintext yang akan dikirimkan menjadi sting-string (disebut blok) dengan panjang t, dan mengenkripsinya per-blok. Pada umumnya block-cipher memproses plaintext dengan blok yang relatif panjang lebih dari 64 bit dengan tujuan untuk mempersulit

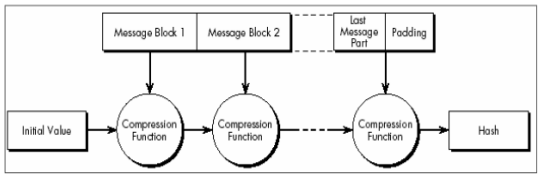
penggunaan pola-pola serangan yang ada untuk membongkar kunci. Sedangkan skema stream cipher pada dasarnya juga block-cipher, hanya dengan panjang bloknya adalah satu bit.

3.4.2. Kunci Asimetrik

Skema ini adalah algoritma yang menggunakan kunci yang ber beda untuk proses enkripsi dan dekripsinya. Skema ini disebut juga sebagai sistem kriptografi Public-key karena kunci untuk enkripsi dibuat secara umum (public-key) atau dapat diketahui oleh siapa saja, tetapi untuk proses dekripsinya yang dibuat satu saja, yakini hanya oleh yang berwenang untuk mendekripsinya (disebut private-key),. Keuntungan skema model ini, untuk berkorespondensi secara rahasia dengan banyak pihak tidak diperlukan kunci rahasia sebanyak jumlah pihak tersebut, cukup membuat dua buah kunci (disebut public-key) bagi para koresponden untuk mengenkripsi pesan, dan private-key untuk mendekripsi pesan. Berbeda dengan skema kunci simetrik yang jumlah kunci yang dibuah adalah harus sebanyak jumlah pihak yang berkorespondensi.

3.5. Message Digest

MD2, MD4, dan MD5 termasuk ke dalam algoritma message-digest, atau kadang juga dikenal dengan hash function. Definisi dari hash function, diambil dari salah satu buletin RSA adalah sebagai berikut, “suatu hash function, atau lebih tepatnya cryptographic hash function, atau juga algoritma message digest, beroperasi pada sebuah string input dengan panjang tidak tentu, dan menghasilkan string output dengan panjang yang sudah ditentukan. Output ini biasa dikenal dengan sebuatn hash value, atau message digest. Secara umum, algoritma dari sebuah hash function adalah sebagai berikut :



Gambar 1. Skema Umum Hash Function

3.5.1. Latar Belakang MD5

Algoritma MD5 disusun oleh Profesor Ronald L. Rivest, dari MIT. Pada RFC 1321, Prof. Ron Rivest memberikan penjelasan awal mengenai MD5, yaitu suatu algoritma yang inputnya berupa sebuah pesan yang panjangnya tidak tertentu, dan menghasilkan keluaran sebuah message digest dari pesan inputnya dengan panjang tepat 128 bit. Diperkirakan (conjectured) tidak mungkin untuk menghasilkan dua pesan dengan message digest yang sama. Algoritma MD5 dimaksudkan untuk aplikasi tanda tangan digital (digital signature), dimana sebuah pesan yang besar harus dipadatkan / di compress dengan cara yang aman sebelum di enkripsi dengan private key dalam sebuah sistem key seperti RSA1. Pada intinya, MD5 adalah sebuah cara untuk

melakukan verifikasi integritas data, dan dapat lebih diandalkan daripada metode yang lebih umum digunakan, seperti checksum. RFC 1321 dikeluarkan pada bulan April 1992, namun MD5 sendiri sebenarnya sudah mulai dikenal pada tahun 1991. MD5 sebenarnya merupakan perbaikan dari pendahulunya, yaitu MD4. Terdapat 6 perbedaan utama antara MD5 dan MD4, yaitu :

- Penambahan tahap ke-empat
- Fungsi pada tahap ke-dua diubah dari XY v XZ v YZ menjadi XZ v YZ'
- Urutan pembacaan input pada tahap ke-dua dan ke-tiga diubah
- Jumlah pergeseran bit pada setiap tahap tidak ada yang sama
- Setiap tahap memiliki penambahan konstanta yang unik
- Untuk mendapatkan hasil akhir, output dari setiap tahap ditambahkan ke tahap setelahnya

Empat perubahan pertama merupakan solusi yang ditawarkan berdasarkan serangan yang terjadi pada metode enkripsi MD4, sedangkan dua perubahan yang terakhir merupakan suatu cara untuk meningkatkan tingkat keamanan enkripsi MD5. Perubahan yang terakhir memiliki efek yang cukup serius, penambahan hasil dari tahap sebelumnya memungkinkan terjadinya collision untuk fungsi kompresi MD5. Dalam makalahnya, den Boer dan Bosselaers memberikan penjelasan mengenai hal ini. Namun serangan ini bukanlah terjadi pada keseluruhan fungsi MD5, oleh karena itu, kadangkala serangan semacam ini disebut pseudo-collision.

3.5.2. Algoritma MD5

Algoritma yang diberikan disini diambil dari RFC 1321, yang disusun oleh Ron Rivest. Dimisalkan kita memiliki pesan sepanjang "b"-bit, dan akan dicari message digestnya. Untuk menghitung message digest dari sebuah pesan, pada MD5 dilakukan tiga langkah sebagai berikut :

1. Penambahan Panjang Bit.

   Pesan diperpanjang sampai sebesar 448 bit, dengan modulo 512. artinya jika panjang pesan telah melebihi 448 bit ini, maka perpanjangan pesan akan dilakukan sampai sebesar 512 + 448 bit, dan begitu seterusnya. Penambahan panjang pesan ini dilakukan dengan cara sebagai berikut, sebuah bit "1" ditambahkan ke dalam pesan. Kemudian bit "0" ditambahkan sampai panjang pesan menjadi 448 bit. Tujuan dari penambahan pesan ini adalah membuat panjang pesan menjadi (kelipatan) 512 bit, dikurangi 64 bit. Kekurangan 64 bit ini akan diatasi pada tahap kedua.

2. Penambahan Panjang Pesan Total

Representasi sebesar 64 bit dari “b” (panjang pesan awal) ditambahkan ke dalam pesan. Jika representasi “b” ini ternyata lebih besar dari 64 bit, maka yang akan diambil hanyalah 64 bit awal (low-order) saja. Panjang pesan total sampai pada tahap ke-dua ini sebesar (kelipatan dari) 512 bit. Tujuan dari penambahan ukuran pesan sampai sebesar kelipatan dari 512 bit ini adalah agar pesan memiliki panjang tepat kelipatan dari 16 word (satu word memiliki ukuran 32 bit). Pengolahan pesan pada tahap keempat nanti akan dilakukan untuk setiap blok sebesar 16 word.

3. Inisialisasi Buffer MD

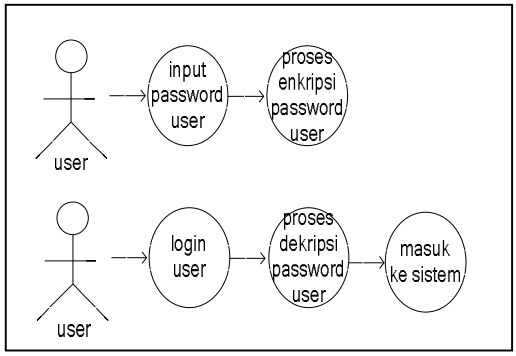
Pada tahap ini digunakan 4 buah register sebagai buffer untuk perhitungan pesan (A, B, C, dan D). Setiap buffer ini memiliki ukuran 32 bit. Ke empat register ini diinisialisasi dengan nilai-nilai berikut (LSB di sebelah kiri):

A	= 01	23	45	67
B	= 89	ab	cd	ef
C	= fe	dc	ba	98
D	= 76	54	32	10

4. ANALISA DAN PERANCANGAN SISTEM

4.1. Perancangan

4.1.1. Rancangan Use Case Diagram

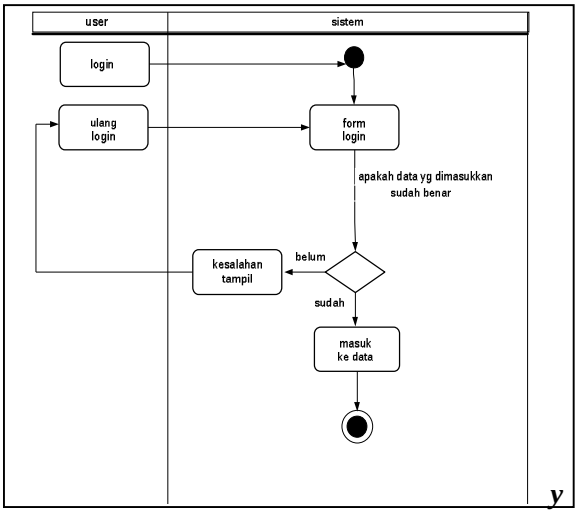


Gambar 2. Diagram Proses User

Proses *usecase* diatas hanya memiliki 1 aktor yaitu user yang mempunyai task/tugas memasukkan data-datanya (*username* dan *password*) untuk dapat login kemudian user tersebut juga bisa mengganti password yang telah ia inputkan. Setelah user menginputkan username dan password, proses enkripsi akan dikerjakan dengan hanya mengenkripsi password user bersangkutan sedangkan username tetap dalam data yang tidak dienkripsi. Dilihat dari gambar diatas bahwa pertama user memasukkan passwordnya kemudian akan di enkripsi oleh sistem, gambar berikutnya user melakukan login dengan password yang sama agar bisa masuk ke dalam sistem/menu utama. Sistem akan membaca/meng dekripsi apakah password yang dimasukkan sesuai dengan field password yang tersimpan jika sesuai sistem akan

membaca apakah username yang dimasukkan juga sesuai, jika username dan password tersebut sesuai dengan pembacaan sistem, maka user bisa masuk atau login ke menu utama.

4.1.2. *Activit Diagram*



Gambar 3. Diagram Proses Enkripsi

Activity diagram diatas menggambarkan bagaimana alur dari user mulai dari user melakukan login sampai user memasuki menu utama dari sistem. Proses enkripsi diatas dapat dijelaskan sebagai berikut: user melakukan login melalui form login, jika data yang dimasukkan user sudah sesuai dengan data yang ada pada database maka proses akan turun kebawah dan selesai, jika data yang dimasukkan user tidak sesuai dengan data yang ada pada database maka user akan mengulangi proses menginput login melalui form login diatas.

4.2. Rancangan Basis Data

4.2.1. Design File dan Data Login

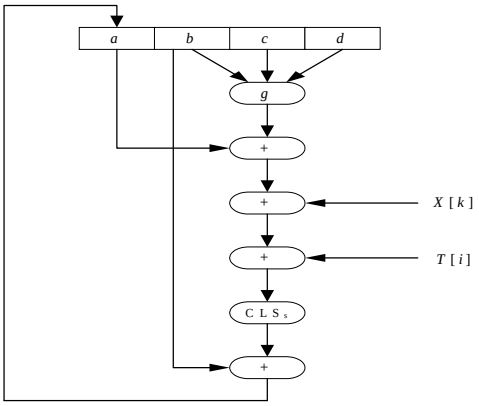
Tabel 1. Data File dan Data Login

No	Nama Field	Type	Width	Keterangan
1 2 3	username password level	varchar varchar char	25 25 1	Username password level hak akses

Tabel Login diatas merupakan table inti dari proses system enkripsi dan deksripsi ini. Tabel hanya terdiri dari 3 buah field yaitu username dengan type varchar dan panjang 25 karakter, password dengan type varchar dan panjang 25 karakter dan level dengan type varchar dan panjang 1 karakter. Field level ini digunakan untuk membedakan user tingkat apa saja yang melakukan login ke dalam sistem, contoh level 1 digunakan untuk ”menandai” bahwa user tersebut sebagai admin, level 2 digunakan untuk ”menandai” bahwa user tersebut sebagai manager dst.

4.3. Operasi Dasar MD5

Operasi dasar *MD5* diperlihatkan pada Gambar berikut:



Gambar 4. Opesasi Dasar MD5

Operasi dasar *MD5* yang diperlihatkan pada Gambar di atas dapat ditulis dengan sebuah persamaan sebagai berikut:

$a \leftarrow b + \text{CLS}s(a + g(b, c, d) + X[k] + T[i])$

keterangan:

a, *b*, *c*, *d* = empat buah peubah penyangga 32-bit *A, B, C, D*

g = salah satu fungsi *F*, *G*, *H*, *I*

CLS*s* = *circular left shift* sebanyak *s* bit

X[*k*] = kelompok 32-bit ke-*k* dari blok 512 bit *message* ke-*q*.

Nilai *k* = 0 sampai 15.

T[*i*] = elemen Tabel *T* ke-*i* (32 bit)

+ = operasi penjumlahan modulo 232

Karena ada 16 kali operasi dasar, maka setiap kali selesai satu operasi dasar, penyangga-penyangga itu digeser ke kanan secara sirkuler dengan cara pertukaran sebagai berikut:

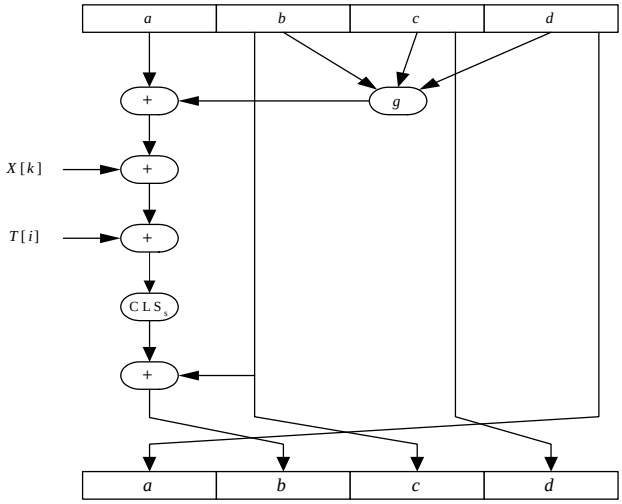
temp ← d

d ← c

c ← b

b ← a

a ← temp



Gambar 5. Opesai Dasar MD5 (Message-Digest Algorihm 5)

Tabel 2. Fungsi-Fungsi Dasar MD5

Nama	*Notasi*	g(b, c, d)
$\mathbf{f_F}$	**F(b, c, d)**	$(b \wedge c) \vee (\sim b \wedge d)$
$\mathbf{f_G}$	**G(b, c, d)**	$(b \wedge d) \vee (c \wedge \sim d)$
$\mathbf{f_H}$	**H(b, c, d)**	$b \oplus c \oplus d$
$\mathbf{f_I}$	**I(b, c, d)**	$c \oplus (b \wedge \sim d)$

Catatan: operator logika AND, OR, NOT, XOR masing-masing dilambangkan dengan $\wedge, \vee, \sim, \oplus$

5. HASIL DAN PEMBAHASAN

5.1. Hasil

Berdasarkan hasil penelitian yang telah dilakukan, maka didapatkan sebuah implementasi pengamanan basis data menggunakan metode enkripsi md5 message-digest algorithm 5 yang dapat berjalan sesuai design dan pembahasan pada bab-bab sebelumnya implementasi pengamanan basis data menggunakan metode enkripsi md5 message-digest algorithm 5 ini dapat diimplementasikan dalam pengamanan basis data khususnya basis data yang berhubungan dengan login ke sistem. Sistem pengamanan basis data ini dibuat menggunakan bahasa pemrograman *PHP* dan *MySQL* sebagai *databasenya*.

5.2. Pembahasan

Setelah dikemukakan hasil dari pembuatan sistem informasi yang telah dibuat, maka pada bagian pembahasan ini penulis akan menguraikan proses-proses yang terjadi pada sistem pengamanan basis data tersebut. Melalui sistem ini diharapkan dapat memberikan suatu kemudahan-kemudahan kepada pihak yang membutuhkan pengamanan untuk basis datanya terutama dalam mengolah data-data login user yang bersangkutan. Sistem informasi ini terbagi menjadi dua halaman yaitu halaman utama dan halaman ganti password. Halaman utama berfungsi untuk register atau pendaftaran member dan login member ke menu ganti password. Halaman ganti password dapat diakses setelah member mendaftar dan login. Berikut ini merupakan Tampilan Utama dari *Website* ini.

5.2.1. Tampilan Halaman Utama

Gambar 6. Menu Utama

Dalam menu utama diatas terdapat dua menu lainnya, yaitu menu signup dan menu

login, kedua menu tersebut dapat dijelaskan sebagai berikut :

1. SignUp, berfungsi pendaftaran member.
2. Login, berfungsi login setelah member mendaftar dan masuk ke dalam menu penggantian password .

6. SIMPULAN

6.1. Simpulan

Setelah melakukan penelitian dan merumuskan pemecahan masalah, maka dapat diambil beberapa kesimpulan yang menyangkut pelaksanaan dan pemanfaatan komputer khususnya dalam pengolahan manajemen keamanan basis data yaitu :

1. Sistem yang dihasilkan adalah implementasi pengamanan basis data menggunakan metode enkripsi md5 (message-digest algorithm 5) secara komputerisasi melalui suatu program khusus yang dirancang menggunakan aplikasi pemrograman berbasis web yaitu php dan menggunakan database mysql yang diharapkan dapat mempermudah dalam pengolahan keamanan basis data.
2. Dengan adanya sistem yang dibangun ini dapat memberikan keamanan data yang tersimpan khususnya data login.
3. Proses pengolahan keamanan data dapat dilakukan dengan cepat karena menggunakan metode md5.

DAFTAR PUSTAKA

Anhar. (2010). *Panduan Menguasai PHP dan MySQL Secara Otodidak*. Jakarta: MEDIA KITA.

Kurniawan, Yusuf. (2004). *Kriptografi Keamanan Internet dan Komunikasi.* Bandung: INFORMATIKA.

Munir, Rinaldi. (2006). *Kriptografi*. Bandung: INFORMATIKA.

Nugroho Bunafik. (2004). *Aplikasi Pemograman Web Dinamis dengan PHP dan MySQL*. Yogyakarta: Gava Media.

Nugroho, Adi. (2005). *Rational Rose Untuk Pemodelan berorientasi Objek.* Bandung: INFORMATIKA.

Nugroho, Bunafik. (2004). *Aplikasi Pemograman Web Dinamis dengan PHP dan MySQL*. Yogyakarta: GAVA MEDIA.

Pressman, Roger S. (2002). *Rekayasa Perangkat Lunak*. Yokyakarta: ANDI.

http://id.wikipedia.org/wiki/. Tanggal akses: 12 novenber 2011.