KOMBINASI FUNGSI MD5 DENGAN VARIABEL PENGACAK

Zulkifli

Program Studi Manajemen Informatika Fakultas Ilmu Komputer Universitas Almuslim
Zulladasicupak@gmail.com

ABSTRAK

Era revolusi industri 4.0 telah membuat perkembangan teknologi informasi menjadi primadona dalam aktivitas sehari-hari, teknologi ini telah berpengaruh pada hampir semua aspek kehidupan manusia. Teknologi internet berlangsung telah menembus batas geografis suatu Negara. Dewasa ini untuk berkomunikasi jarak jauh antar negara bukan lagi suatau tantangan yang berat dan sulit bagi seseorang untuk berkomunikasi jarak jauh, mengirimkan data, mencari informasi dan sebagainya. Hanya bermodal dengan jari tangan dan sebuah gawai semua hal tersebut dapat dilakukan dengan internet secara cepat, efisien dan relatif murah. Namun di sisi lain, ternyata internet merupakan jalur yang tidak terlalu aman karena merupakan media komunikasi umum yang dapat digunakan secara bebas oleh siapapun sehingga sangat rawan penyadapan informasi oleh pihak-pihak yang tidak abash. Keamanan data yang disajikan di internet via web semakin hari semakin ditingkatkan. Banyak cara yang dilakukan oleh programmer web, mulai dari tingkat rendah, hingga ke proteksi yang sangat handal guna menghindari serangan dari hacker. Salah satunya dengan mengkombinasikan fungsi ***md5*** *dengan sebuah variabel pengacak dalam hal menjaga kerahasian data seorang user dan passwordnya pada saat login di web. Metode ini bisa diterapkan untuk verifikasi single user, maupun multi user dengan menggunakan sebuah database server. Fungsi* ***md5*** *merupakan kriptografi modern, yang juga merupakan hash satu arah yang berfungsi meng-enkripsi plain text ke chipper text dengan panjang string 32 karakter, yang merupakan kombinasi angka dan huruf acak. Semakin bagus modifikasi unik yang dilakukan pada kombinasi fungsi* ***md5*** *dengan variabel pengacak ini, semakin tinggi juga tingkat keamanan yang hasilkan.*

Kata kunci: Fungsi md5, variabel pengacak, plain text, chipper text.

PENDAHULUAN

Dengan perkembangan dunia di era revolusi Industri 4.0, penggunaan internet dalam kehidupan sehari – hari sudah merupakan hal yang umum. Ketika pengguna internet melakukan browsing atau surfing seringkali ia diminta untuk mengirimkan data pribadinya. Menurut penulis, terdapat 2 jenis data yang kita bicarakan disini, yakni data yang diminta secara implicit maupun data yang diminta secara explicit. Data yang diminta secara explisit contohnya nama user, alamat email, dan sebagainya, pada saat kita hendak memberikan komentar di sebuah web. Data yang diminta secara implist contohnya ketika web browser mengirimkan cookie untuk melakukan autentikasi. Data – data tersebut bersifat pribadi dan rahasia.

Penggunaan data tersebut oleh orang yang tidak berhak dapat menyebabkan kerugian terhadap pemilik data baik secara materil maupun imateril. Contohnya jika terjadi fraud atas sebuah akun bank atau ketika orang lain mengaku sebagai diri kita dan melakukan perbuatan yang dapat menurunkan harga diri kita.

Beberapa metode sudah hadir di sekeliling kita dengan tujuan untuk mencegah data-data penting dan rahasia agar tidak dapat diketahui oleh orang yang tidak berhak. Sebagai contoh penggunaan kombinasi fungsi **md5** dengan variabel pengacak untuk meng-enkripsi user name dan password menjadi 32 karakter yang sukar ditebak oleh si pembajak pada saat seorang user sedang login.

Tulisan ini bertujuan untuk menjelaskan pemakaian fungsi **md5** dengan mengkombinasikannya dengan sebuah variabel pengacak menggunakan bahasa pemrograman web standar yaitu HTML dan PHP script.

METODE PENELITIAN

Langkah-langkah pembuatan message digest secara garis besar adalah sebagai berikut:

1. Penambahan bit-bit pengganjal (padding bits). Pesan ditambah dengan sejumlah bit pengganjal sehingga panjang pesan kongruen dengan 448 modulo 512. Bit pengganjal diisi dengan sebuah bit 1, kemudian diikuti dengan bit 0 untuk sisanya.
2. Penambahan nilai panjang pesan semula. Pesan yang telah diberi bit-bit pangganjal selanjutnya ditambahkan lagi dengan 64 bit yang menyatakan panjang pesan.
3. Inisialisasi penyangga (buffer) MD **md5** membutuhkan 4 buah penyangga yang masing-masingnya berukuran 32 bit. Keempat bit penyangga ini menampung hasil antara dan hasil akhir. Keempat bit penyangga dan inisialisasinya adalah sebagai berikut:

   A = 01234567
   B = 89ABCDEF
   C = FEDCBA98
   D = 76543210
4. Pengolahan pesan dalam blok berukuran 512. Pesan dibagi menjadi L buah blok yang masing-masing panjangnya 512 bit. Setiap blok diproses bersama penyangga MD yang menghasilkan keluaran 128 bit, proses ini di sebut HMD5. Proses **md5** terdiri dari 4 putaran dimana setiap putaran melakukan 16 operasi dasar **md5** dan tiap operasinya memakai sebuah elemen T yang nilainya tertera pada tabel T yang di peroleh dari perihitungan berdasarkan rumus yang telah ditentukan.

HASIL DAN PEMBAHASAN

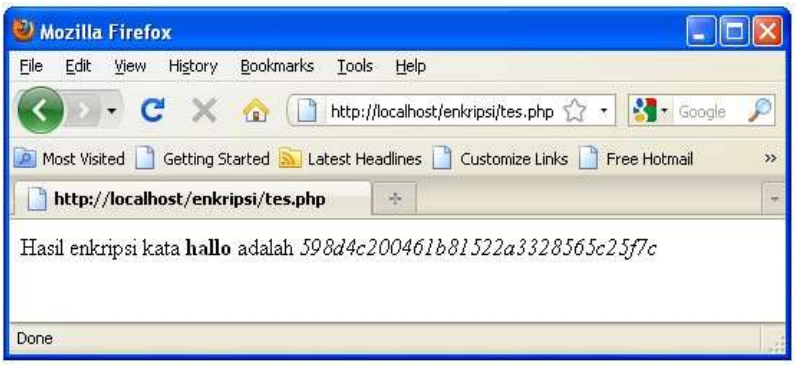
Setiap password pengguna aplikasi hendaknya dienkripsi untuk keperluan faktor keamanan. Mengapa password harus dienkripsi? Atau apa sebenarnya enkripsi itu? Enkripsi adalah teknik penyandian pesan, yang semula pesan tersebut dapat dibaca dan bermakna, setelah dienkripsi menjadi tidak terbaca dan tidak bermakna. Lantas mengapa password harus dienkripsi? Pertanyaan tersebut kita balik sekarang, bagaimana jika password tidak dienkripsi? Apabila password tidak dienkripsi, maka dapat dengan mudah dibaca, dan digunakan oleh orang lain yang tidak berhak untuk masuk ke dalam sistem atau aplikasi.

Dalam PHP, tentu kita tidak asing dengan perintah atau function **md5()**. Function ini sering digunakan para programmer untuk mengenkripsi password sebelum hasil enkripsi tersebut disimpan dalam database sistem, ketika registrasi user baru. Hasil enkripsi **md5()** berupa suatu string acak dengan panjang 32 karakter (256 bit). Sudah amankah penggunaan **md5()?** Artikel ini akan membahasnya, serta memberikan tips bagaimana cara membuat script PHP yang baik untuk mengolah password.

Dalam fungsinya sebagai pengubah (enkripsi) sebuah kata (word) ke dalam bentuk kombinasi huruf dan angka acak sebanyak 32 digit, **md5** dapat dipakai untuk meng-ekripsi kata-kata penting seperti username dan password pada sebuah formulir di internet, sehingga para pembajak sukar untuk menebak atau mengkombinasikan berbagai macam kombinasi yang mungkin untuk membobolkan informasi yang sangat rahasia yang disajikan via web.

Berikut ini akan adalah contoh penggunaan **md5** dengan menggunakan PHP script:

Output dari potongan program di atas adalah sebagai berikut:



Dengan mendeklarasikan fungsi **md5** pada baris ke 3 potongan program di atas, maka kata-kata yang terdapat dalam variabel $word akan di-enkripsi menjadi kombinasi angka dan huruf acak sebanyak 32 digit seperti yang terdapat pada output program.

Berdasarkan contoh di atas, kita dapat memanfaatkan fungsi **md5** untuk mengenkripsi user name dan password seorang user pada web dari serangan hacker. Untuk lebih meningkatkan keamanan tersebut, alangkah lebih baik lagi hasil passing variabel dari fungsi **md5** tersebut dikombinasikan dengan sebuah variabel bebas.

Sekarang apakah sudah aman penggunaan md5() dengan struktur kode di atas? Beberapa periode yang lalu, mungkin penggunaan struktur seperti di atas sudah dirasa aman. Namun saat ini tidak aman lagi, karena sudah banyak tool untuk mendekripsi hasil enkripsi md5(). Apa akibatnya jika password ini didekripsi? Ini sangat bahaya, karena ada kemungkinan password aslinya ketahuan. Tetapi masalah ini tidak perlu dikhawatirkan, kita tetap bisa menggunakan md5() namun perlu sedikit kreatif. Maksudnya adalah bahwa kita perlu mengkombinasikan penggunaan md5() dengan pengacak, misalnya kita gunakan md5() berulangkali, atau menggabungkan password asli dengan suatu string tertentu lalu dienkripsi.

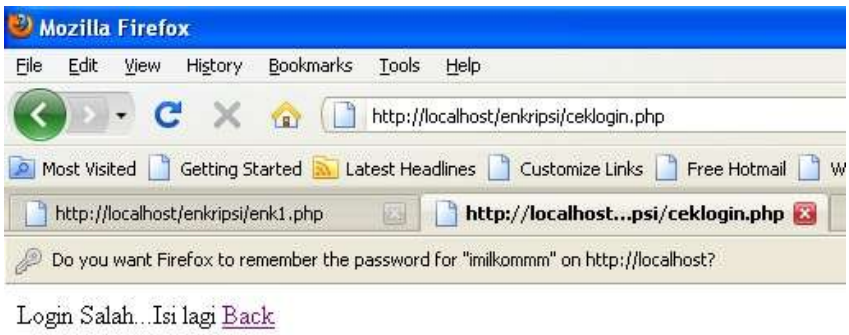
Sekarang kita dapat mengubah isi pengacak atau mungkin mengubah format enkripsinya menjadi model lain, misalnya menggabungkan 3 atau lebih md5() dalam enkripsi. Dalam hal ini, hanya kita yang tahu format enkripsi atau pengacaknya. Intinya adalah jangan mengenkripsi password menggunakan md5() secara langsung, karena hal ini rawan untuk dihack pada saat ini.

Berikut ini adalah potongan program yang terdiri dari 2 file sebagai contoh kombinasi penggunaan fungsi **md5()** dengan sebuah variabel pengacak:

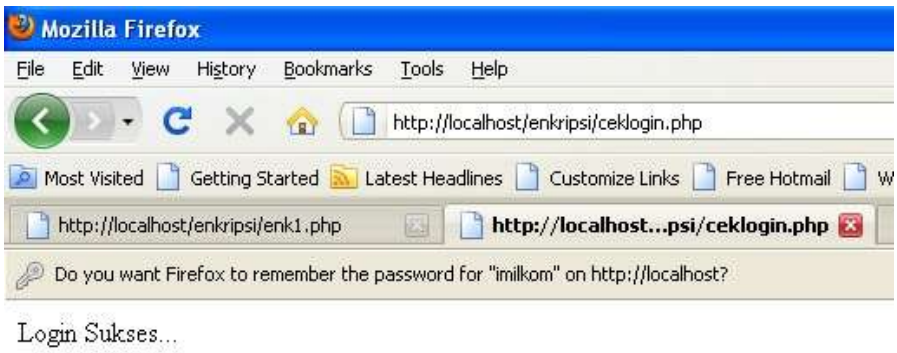
File1: login.php

File2 : ceklogin.php

Output kalau username dan password salah.



Output kalau username dan password benar.



PENUTUP

Dengan melakukan kombinasi fungsi **md5** dengan sebuah variabel pengacak, maka keamanan data di web akan lebih terjaga kerahasiaannya, karena kombinasi tersebut sangat susah diacak oleh orang-orang yang tak bertanggung jawab.

DAFTAR PUSTAKA

Schneier,B. 1994. *Applied Cryptography*, John Wiley & Sons, New York.

Black,J et al. 2006. *A study of the MD5 attacks: Insight and Improvements*, Colorado.

Analisis konsep Balanced Scorecard (BSC) untuk mengukur kinerja Pegawai dengan Pendekatan IT, *jurnal Lentera*, Vol 1, No.03, Bireuen 2017, ISSN2548-7663.

Thomas, Tom., *Network Security First Step*, Penerbit Andi, 2004.

Teknik pengolahan Citra untuk mendeteksi kematangan Boh Giri menggunakan metode Nave bayes, *Jurnal Variasi majalah ilmiah universitas almuslim*, vol.10.nomor 5 Desember 2018.

Munir, Rinaldi. 2004, *Fungsi Hash Satu Arah dan Algoritma MD5*, Institut Teknologi Bandung.

Mulya, Megah, 2008, *Bahan Ajar Kriptografi S1*, Universitas Sriwijaya.

http://www.priatama.com/2010/12/ssl-tls-pengamanan-pengiriman-dan-penerimaan- data/

http://budi.paume.itb.ac.id/courses/ec5010/2015/purnomo-report2.pdf

1http://www.informatika.org/~rinaldi/Kriptografi/2007-

2008/Makalah2/MakalahIF5054-2007-B-065.pdf