OPTIMASI ENKRIPSI PASSWORD MENGGUNAKAN ALGORITMA BLOWFISH

Yani Parti Astuti[1], Eko Hari Rachmawanto[2], Christy Atika Sari[3]
[1,2,3]Teknik Informatika S1, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
Jl. Nakula No 5-11, Semarang,
E-mail : yanipartiastuti@dsn.dinus.ac.id [1], rachmawanto@research.dinus.ac.id[2],
atikasari@research.dinus.ac.id[3]

Abstrak

Seiring dengan perkembangan informasi yang terus meningkat salah satunya penggunaan komputerisasi dalam berbagai bidang. Hal ini erat kaitannya dengan penggunaan password dalam area komputerisasi. Rawannya manupulasi password dinilai sebagai hal yang perlu dicegah menggunakan data hiding. Salah satu teknik dalam data hiding yaitu Kriptografi. Dalam kriptografi terdapat banyak algoritma, salah satnya yang memiliki kehandalan yaitu algoritma blowfish. Sampai saat ini algoritma Blowfish belum ditemukan kelemahan yang berarti hanya adanya weak key dimana dua entri dari S-box mempunyai nilai yang sama. Belum ada cara untuk mengecek weak key sebelum melakukan key expansion, tetapi hal ini tidak berpengaruh terhadap hasil enkripsi. Hasil enkripsi dengan algoritma Blowfish sangat tidak mungkin dan tidak praktis untuk di terjemahkan tanpa bantuan kunci. Sampai kini belum ada Cryptanalysis yang dapat membongkar pesan tanpa kunci yang enkripsi oleh Blowfish. Hasil pengujian menunjukkan bahwa aplikasi bisa menjalankan fungsi-fungsi untuk melakukan proses enkripsi dan dekripsi data dengan baik. Waktu proses untuk enkripsi dan dekripsi untuk masing-masing file mempunyai sedikit perbedaan dikarenakan ukuran antara plainteks dan cipherteks juga berbeda, sedangkan waktu yang diperlukan juga lebih lama. Penelitian ini telah menghasilkan aplikasi enkripsi password yang telah diuji coba dan algoritma blowfish terbukti handal dalam mengamankan password.

Kata kunci : *Blowfish, Key-Expansion, Kriptografi, File.*

Abstract

During the development of information to increase one use of computerization in various fields. It is closely related to the use of passwords in the area of computerization. Manipulation password assessed as things that need to be prevented using the data hiding. One technique in data hiding is Cryptography. There are many algorithms in cryptography, one of which has a reliability is blowfish algorithm. Until now the Blowfish algorithm weakness has not been found, which means only the weak key in which two of the S-box entries have the same value. There is no way to check for weak keys before performing key expansion, but this does not affect the outcome of encryption. Results encryption with Blowfish algorithm is highly unlikely and impractical to translate without the help of a key. Until now there has been no Cryptanalysis can unload without key encryption message by Blowfish. The test results showed that the application can perform functions to perform encryption and decryption of data well. The processing time for encryption and decryption for each file to have a little difference because of the size between plaintext and ciphertext is also different, while the time required too much longer. This research has produced application password encryption algorithm has been tested and proven reliable in securing blowfish password.

Keywords: *Blowfish, Key-Expansion, Cryptography, File.*

1. PENDAHULUAN

Masalah keamanan merupakan salah satu aspek paling penting dalam dunia teknologi informasi. Salah satu hal yang paling sering ingin dimanipuasi yaitu password. Password yang merupakan sandi rahasia seseorang biasanya dapat dengan mudah ditebak maupun di hacker. Seiring pesatnya kemajuan teknologi dan perkembangan algoritma, password dapat dibuat lebih aman dengan cara mengenkripsi.

Salah satu teknik yang berhubungan dengan enkripsi yaitu Kriptografi. Dalam kriptografi terdapat beberapa metode yang cukup penting dalam pengamanan data, untuk menjaga kerahasiaan data salah satunya adalah enkripsi *(encryption)*. Enkripsi adalah suatu proses yang dilakukan untuk mengubah pesan asli menjadi chipertext. Sedangkan suatu proses yang dilakukan untuk mengubah pesan tersembunyi menjadi pesan asli disebut dekripsi. Pesan biasa atau pesan asli disebut plainteks sedangkan pesan yang telah diubah atau disandikan supaya tidak mudah dibaca disebut dengan chiperteks. Untuk mengatasi masalah keamanan dokumen ini, maka dilakukan pendekatan teknologi enkripsi data menggunakan algoritma Blowfish atau sering disebut OpenPGP.Cipher.4.

Dalam pengembanagnnya, blowfish dinilai sebagai saah satu algoritma kriptografi symetris yang cepat dan kompak, mempunyai perhitungan sederhana sedangkan panjang kunci yang biasa digunakan yaitu bervariasi mulai dari 32 bit sampai 128 bit. Blowfish dioptimasikan untuk berbagai aplikasi dimana kunci tidak sering berubah, seperti pada jaringan komunikasi atau enkripsi file secara otomatis [1].

Blowfish dirancang dengan kriteria rancangan cepat, tersusun dengan rapi, sederhana, dan panjang kunci digunakan sebagai variable untuk penyandian data. Pada penelitian ini Algoritma blowfish akan diterapkan untuk dievaluasi sejauh mana dapat digunakan dalam mengamankan password.

Dalam pengembangan dan implementasinya, algoritma blowfish telah digunakan dan dinilai handal, seperti pada penelitian berikut: Menurut Sitinjak dkk dalam penelitiannya memaparkan kemampuan yang dimiliki oleh algoritma blowfish dalam hal kecepatan proses enkripsi dan dekripsi jika dikaitkan dengan ukuran sebuah file [2], sedangkan Mandal [3] dalam penelitiannya dapat menyimpulkan bahwa Algoritma Blowfish merupakan algoritma enkripsi yang kuat dengan kecepatan tinggi dan memiliki konsumsi energi yang minimum.

Penelitian lain juga telah dilakukan oleh Utami [4] menyatakan bahwa algoritma blowfish yang optimal dapat dilakukan dengan aplikasi yang tidak sering berubah-ubah kunci serta tidak menggunakan *weak-key*. Sedangkan pada pengembangannya blowfish juga telah diterkapan bersamaan dengan End Of File untuk mendapatkan hasil yang lebih baik, dikarenakan data yang digunakan adaah data gambar [5].

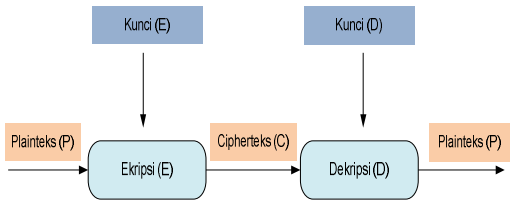
2. METODE

2.1 Kriptografi

Kriptografi berasal dari bahasa Yunani yang terdiri cryptos yang berarti menyembuyikan sedangkan graphia berarti tulisan. Kriptografi merupkan ilmu yang mempelajari teknik

matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan data, antektikasi, integritas dan keabsahan data. Kriptografi juga dapat diartikan sebagai ilmu untuk menjaga kerahasiaan pesan [1].

Kemudian, proses yang akan dibahas dalam penelitian ini meliputi 2 proses dasar pada kriptografi yaitu enkripsi dan dekripsi. Enkripsi yaitu proses mengubah data asli menjadi pesan yang tidak dapat dibaca, sedangkan dekripsi merupakan proses menjadikan data hasil manipulasi menjadi data asli. Berikut ini merupakan ilustarasi sederhana dari proses kriptografi. Pada prinsipnya, Kriptografi memiliki 4 komponen utama yaitu *plaintext*, yaitu pesan yang dapat dibaca. *Ciphertext*, yaitu pesan acak yang tidka dapat dibaca. *Key*, yaitu kunci untuk melakukan teknik kriptografi. *Algorithm*, yaitu metode untuk melakukan enkrispi dan dekripsi.



Gambar 1. Proses Penyisipan Pesan pada Kriptografi [1]

Gambar 1 menunjukkan proses penyembunyian pesan menggunakan teknik kriptografi. Pada kriptografi simetris, kunci yang digunakan untuk proses enkripsi dan dekripsi sama sedangkan pada kriptografi asimteris kunci yang digunakan berbeda.
Menurut terminologinya kriptografi merupakan ilmu dan seni untuk menjaga keamanan pesan ketika pesan tersebut dikirim dari suatu tempat ke tempat lain [6]. Jika seorang kriptografer menggunakan enkripsi untuk merahasiakan pesan dan mendeskripsikannya kembali, maka kriptanalis mempelajari metode enkripsi dan cipherteks untuk menemukan plainteksnya [7].

Dalam kriptografi modern tidak mendasarkan kekuatan pada algoritmanya. Kekuatan kriptografinya terletak pada kunci, yang berupa deretan karakter atau bilangan bulat. Kunci ini sama fungsinya dengan sandi-lewat (password) pada system computer yang dijaga kerahasiaannya dan hanya orang yang mengetahui kunci yang dapat melakukan enkripsi dan dekripsi [8].

2.2 Algoritma Blowfish

Blowfish atau disebut juga OpenPGP.Cipher.4 adalah enkripsi yang termasuk dalam golongan Symmetric Cryptosystem. Blowfish merupakan algoritma kunci simetrik cipher blok yang dirancang pada tahun 1993 oleh Bruce Schneier untuk menggantikan *Data Encryption Standard* (DES) [7]. Metode enkripsi ini diciptakan oleh Bruce Schneier, seorang Cryptanalyst Presiden perusahaan Counterpane Internet Security, Inc pada tahun 1993. Dan dipublikasikan tahun 1994. Blowfish dibuat untuk digunakan pada komputer yang mempunyai mikroprosesor besar (32 bit ke atas dengan *cache data* yang besar).

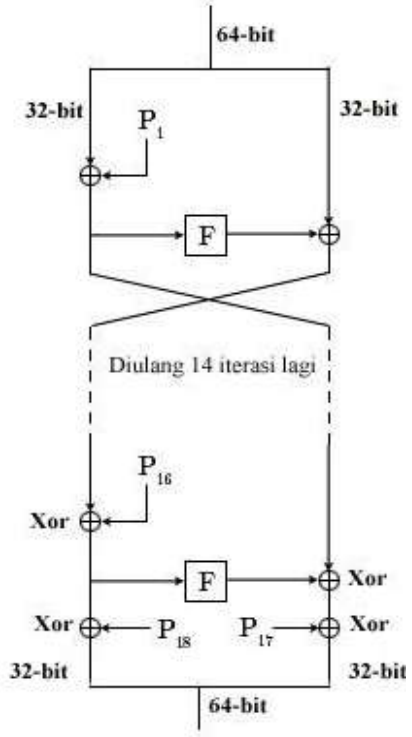
Blowfish merupakan cipher blok. Yang berarti selama proses enkripsi dan dekripsi, Blowfish bekerja dengan membagi pesan menjadi blok-blok bit dengan ukuran sama panjang yaitu 64-bit dengan panjang kunci bervariasi yang mengenkripsi data dalam 8 byte blok [9]. Pesan yang bukan merupakan kelipatan 8 byte akan ditambahkan bitbit tambahan (padding) sehingga ukuran untuk tiap blok sama.

Algoritma dalam Blowfish terbagi menjadi dua bagian, yaitu key expansion dan data encryption. Proses key expansion akan melakukan konversi sebuah kunci mulai dari 56 byte sampai beberapa array sub kunci dengan total mencapai 4168 byte. Blowfish dirancang dan diharapkan mempunyai kriteria perancangan yang diinginkan sebagai berikut :

- Cepat, Blowfish melakukan enkripsi data pada microprocessor 32-bit dengan rate 26 clock cycles per byte.
- Compact, Blowfish dapat dijalankan pada memory kurang dari 5K. Sederhana, Blowfish hanya menggunakan operasi – operasi sederhana, Blowfish hanya menggunakan operasi – operasi sederhana, seperti : penambahan, XOR, dan lookup tabel pada operan32- bit. 4.
- Memiliki tingkat keamanan yang bervariasi, panjang kunci yang digunakan oleh Blowfish dapat bervariasi dan bisa sampai sepanjang minimal 32-bit, maksimal 448 -bit, Multiple 8 bit, default 128 bit.

Namun menurut [7], jika algoritma diterapkan dengan kunci yang sering berubah akan membutuhkan proses penurunan baru pada iterasi yang panjang, hal ini akan membuat waktu kerja Blowfish lebih panjang, sedangkan penggunaan *weak key* dapat mengganggu hasil enkripsi dan dekripsi. *Weak key m*embuat hasil enkripsi/dekripsi menjadi tidak konsisten. Tingkat keamanan algoritma Blowfish ditentukan oleh jumlah iterasi dan panjang serta kerahasiaan kunci yang digunakan jumlah iterasi yang digunakan semestinya membuat jaringan feistel pada Blowfish bekerja semestinya, pengurangan jumlah iterasi akan mengurangi tingkat kesulitan suatu data untuk dipecahkan, sedangkan peran panjang dan kerahasiaan kunci menjadi sangat krusial. Kunci yang panjang menjadi sama tingkat kebutuhannya dengan iterasi yang tidak dikurangi karena proses pembangkitan *sub key* akan menjadi lebih acak dan membutuhkan waktu lama untuk dipecahkan.

Ekspansi kunci pada algoritma blosfish berfungsi untuk merubah kunci (Minimum 32-bit, Maksimum 448-bit) menjadi beberapa array subkunci (subkey) dengan total 4168 byte. Algoritma Blowfish terdiri dari iterasi fungsi sederhana (*Feistel Network)* sebanyak 16 kali putaran. Setiap putaran terdiri dari permutasi kunci-dependent dan substitusi kunci- dan datadependent. Semua operasi adalah penambahan *(addition)* dan XOR pada variabel 32-bit seperti ditunjukkan pada Gambar 2 dibawah ini.

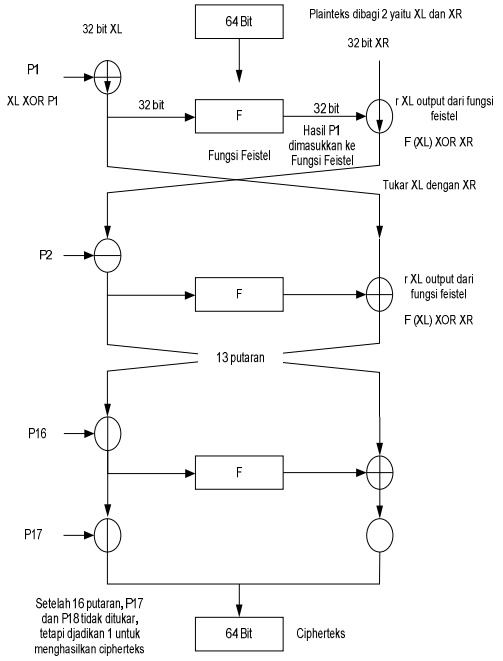


Gambar 2. Algoritma Blowfish [6]

Operasi tambahan lainnya hanyalah empat penelusuran tabel *(table lookup)* array berindeks untuk setiap putaran. Untuk alur algoritma enkripsi dengan metoda Blowfish dijelaskan sebagai berikut : 1. Bentuk inisial array P sebanyak 18 buah (P1,P2, ……..P1 masing-msing bernilai 32-bit. Array P terdiri dari delapan belas kunci 32-bit subkunci : P1,P2,…….,P18 2. Bentuk S-box sebanyak 4 buah masing-masing bernilai 32-bit yang memiliki masukan 256. Empat 32-bit S-box masing-masing mempunyai 256 entri :
S1,0,S1,1,……..,S1,255
S2,0,S2,1,……..,S2,255
S3,0,S3,1,……..,S3,255
S4,0,S4,1,……..,S4,255
3. Plainteks yang akan dienkripsi diasumsikan sebagai masukan, Plainteks tersebut diambil sebanyak 64-bit, dan apabila kurang dari 64-bit maka kita tambahkan bitnya, supaya dalam operasi nanti sesuai dengan datanya. 4. Hasil pengambilan tadi dibagi 2, 32-bit pertama disebut XL, 32-bit yang kedua disebut XR. 5. Selanjutnya lakukan operasi XL = XL xor Pi dan XR = F(XL) xor XR 6. Hasil dari operrasi diatas ditukar XL menjadi XR dan XR menjadi XL. 7. Lakukan sebanyak 16 kali, perulangan yang ke- 16 lakukan lagi proses penukaran XL dan XR. 8. Pada proses ke-17 lakukan operasi untuk XR = XR xor P17 dan XL = XL xor P18. 9. Proses terakhir satukan kembali XL dan XR sehingga menjadi 64-bit kembali.

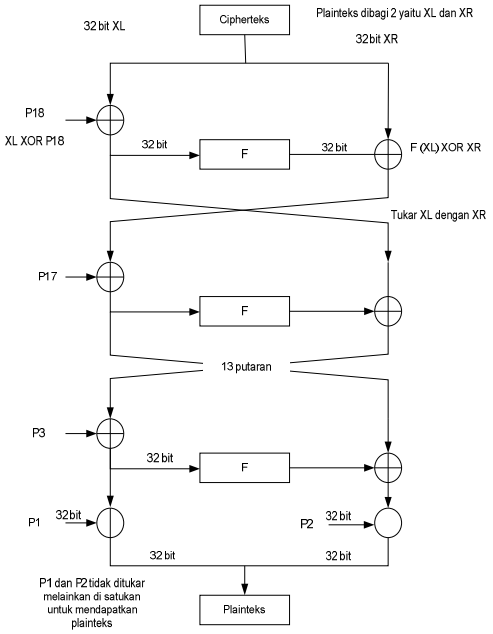
3. HASIL DAN PEMBAHASAN

Berikut ini merupakan flowchart penelitian yang digunakan.



Gambar 3. Flowchart Proses Penyisipan File menggunakan algoritma Blowfish

Sedangkan proses ektraksi file dapat dilihat pada Gambar 4 berikut ini.



Gambar 4. Flowchart Proses Ekstraksi File menggunakan algoritma Blowfish

Aplikasi keamanan password ini dibuat dengan bahasa pemrograman Visual Basic 6.0.

Aplikasi ini berfungsi untuk memanipulasi password sebelum digunakan, sehingga orang lain tidak mengetahui. Disisi lain, password yang digunakan mempunyai 2 kali pertahanan.

Gambar 5. Tampilan Awal Aplikasi Enkripsi Password

Gambar 5 merupakan tampailan awal aplikasi pengamanan password menggunakan algoritma blowfish. Terdapat pilihan opsional untuk mengaktifkan kunci yang digunakan, yaitu hexastring atau alfabet. Setelah kunci ditetapkan dari kunci menggenerate berdasarkan opsi yang telah dipilih maka pengguna harus mengetikkan plainteks yang akan digunakan.

Gambar 6. Proses Enkripsi Password

Gambar 6 merupakan proses enkripsi menggunakan algoritma blowfish. Pada percobaan yang dilakukan, opsi dari model kunci yang digunakan adalah mode alfabet.

Gambar 7. Proses Ekstraksi Password

Pada Gambar 7, proses ekstraksi dilakukan untuk mengevaluasi apakah proses enkripsi sudah dilakukan dengan benar. Dalam percobaan yang telah dilakukan, eksraksi berjalan dengan baik dan menghasilkan deciphered yang sama dengan plainteks (file induk asli).

4. KESIMPULAN

Algoritma blowfish merupakan algoritma yang cepat, tersusun secara rapi, dpaat dengan mudah dijalankan, sederhana, dan terjamin keamanannya. Sampai saat ini belum ada cryptanalyst yang berhasil menembus keamanan yang dibuat oleh algoritma blowfish dengan 16 kali putaran. Dari hasil percobaan yang telah dilakukan dalam penelitian ini, maka dapat disimpulkan bahwa algoritma blowfish merupakan algoritma yang handal untuk mengamankan password.

DAFTAR PUSTAKA

[1] S. Kromodimoeljo, *Teori dan Aplikasi Kriptografi*. SPK IT Consulting, 2009, p. 458.

[2] S. Sitinjak, Y. Fauziah, and Juwairah, "APLIKASI KRIPTOGRAFI FILE MENGGUNAKAN," *Seminar Nasional Informatika 2010 (semnasIF 2010)*, pp. 78-86, 2010.

[3] P. C. Mandal, "Superiority of Blowfish Algorithm," *International Journal*, vol. 2, no. 9, pp. 196-201, 2012.

[4] Sukrisno and E. Utami, "IMPLEMENTASI STEGANOGRAFI TEKNIK EOF DENGAN GABUNGAN ENKRIPSI RIJNDAEL , SHIFT CIPHER DAN FUNGSI HASH MD5," *Seminar Nasional Teknologi 2007 (SNT 2007)*, no. November, pp. 1-16, 2007.

[5] P. A. Nani, "PENERAPAN ENKRIPSI ALGORITMA BLOWFISH PADA PROSES STEGANOGRAFI METODE EOF," pp. 1-6, 2011.

[6] T. Andriyanto and D. L. C. Pardede, "STUDI DAN PERBANDINGAN ALGORITMA IDEA DAN ALGORITMA BLOWFISH."

[7] E. Utami, S. Erikawaty, and A. Tambunan, "Pendahuluan Cryptosystem," *Jurnal Dasi*, vol. 11, no. 2, pp. 33-44, 2010.

[8] Ratih, "STUDI DAN IMPLEMENTASI ALGORITMA BLOWFISH UNTUK APLIKSI ENKRIPSI DAN DEKRIPSI FILE," pp. 1-15.