Implementasi Sistem Keamanan File Menggunakan Algoritma Blowfish pada Jaringan LAN

Anggi Purwanto
Program Studi Teknik Telekomunikasi,
Fakultas Teknik Elektro dan Komunikasi
Institut Teknologi Telkom
Jl. Telekomunikasi Ters. Buah Batu,
Bandung, 40257
anggipurwanto87@gmail.com

Ledya Novamizanti
Program Studi Teknik Telekomunikasi,
Fakultas Teknik Elektro dan Komunikasi
Institut Teknologi Telkom
Jl. Telekomunikasi Ters. Buah Batu,
Bandung, 40257
ledyamizan@gmail.com

Ringkasan

Keamanan suatu informasi menjadi hal yang sangat penting saat ini. Banyak orang kemudian berusaha untuk mencari cara bagaimana mengamankan informasi dalam melakukan pertukaran informasi. Salah satu caranya adalah dengan metode enkripsi menggunakan algoritma simetri. Namun terdapat kendala dalam penggunaan kunci untuk tipe algoritma simetri, dimana kunci yang digunakan untuk enkripsi dan dekripsi harus sama, sedangkan jika kunci untuk dekripsi dikirimkan terpisah akan menyebabkan kunci dapat diketahui dengan mudah oleh penyadap. Pada penelitian ini dirancang suatu aplikasi enkripsi dan dekripsi file menggunakan algoritma Blowfish serta mengimplementasikannya pada jaringan LAN. Pengujian terhadap sistem akan dilakukan dengan mengukur kinerja dari algoritma Blowfish dari segi waktu enkripsi dan dekripsi, waktu pemecahan kunci, serta avalanche effect. Pada akhirnya sistem ini dapat mengatasi kelemahan pada konsep algoritma simetri dalam hal pengiriman data.

Kata Kunci: Kriptografi, Algoritma Blowfish, Enkripsi, Dekripsi, LAN.

1 Pendahuluan

Algoritma kriptografi dapat dibagi ke dalam kelompok algoritma simetris dan algoritma asimetris. Algoritma simetris merupakan algoritma kriptografi yang menggunakan kunci yang sama baik untuk proses enkripsi maupun dekripsi. Algoritma simetris dapat dikelompokkan menjadi dua kategori, yaitu cipher aliran (stream chiper) dan cipher blok (block chiper). Cipher aliran merupakan algoritma kriptografi yang beroperasi dalam bentuk bit tunggal. Sedangkan algoritma kriptografi kategori cipher blok beroperasi dalam bentuk blok bit.

Namun terdapat kendala dalam penggunaan kunci untuk tipe algoritma simetri, dimana kunci yang digunakan untuk enkripsi dan dekripsi harus sama, sedangkan jika kunci untuk dekripsi dikirimkan terpisah akan menyebabkan kunci dapat diketahui dengan mudah oleh penyadap. Pada penelitian sebelumnya, dilakukan penggabungan algoritma simetri dengan algoritma asimetri untuk keamanan dalam pengiriman data (Ariwibowo, 2005). Hal tersebut dilakukan agar kunci untuk proses dekripsi dapat dikirimkan dengan aman. Tetapi dengan penggabungan algoritma tersebut, maka akan berakibat proses yang dibutuhkan lebih lama.

Untuk itu pada penelitian ini dirancang suatu sistem kemanan file menggunakan Algoritma Blowfish pada jaringan LAN. Kunci yang digunakan untuk enkripsi dan dekripsi akan disamarkan dan disisipkan bersama dengan data yang telah dienkripsi, hal ini dilakukan agar informasi kunci tidak dapat diketahui dengan mudah oleh penyadap. setelah data yang telah dienkripsi dan dikirimkan sampai pada penerima, kunci yang telah disamarkan dan disisipkan akan diambil kembali dari data dan akan digunakan untuk proses dekripsi.

Pengujian terhadap sistem akan dilakukan dengan mengukur kinerja dari algoritma Blowfish dari segi waktu enkripsi dan dekripsi, waktu pemecahan kunci, serta avalanche effect. Sehingga, pada akhirnya sistem ini dapat mengatasi kelemahan pada konsep algoritma simetri dalam hal pengiriman data.

2 Landasan Teori

2.1 *Kriptografi*

Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasian data, keabsahan data, integritas data, serta autentifikasi data [1].

Algoritma Kriptografi

Berdasarkan kunci yang digunakan dikenal dua buah algoritma kriptografi, yaitu:

1. Algoritma Simetris / Konvensional

   Algoritma simetris (Symmetric Algorithm) adalah suatu algoritma dimana kunci enkripsi yang digunakan sama dengan kunci dekripsi sehingga algoritma ini disebut juga sebagai single-key-algorithm. Sebelum melakukan pengiriman pesan, pengirim dan penerima harus memilih suatu kunci tertentu yang sama untuk dipakai bersama, dan kunci ini haruslah rahasia bagi pihak yang tidak berkepentingan sehingga algoritma ini disebut juga algoritma kunci rahasia (secret-key algorithm). Penerapan algoritma akan menghasilkan output yang berbeda sesuai dengan kunci yang dipakai.

2. Algoritma Asimetris / Kunci Publik

   Pada algoritma asimetris kunci kriptografi dibuat sepasang, satu kunci untuk enkripsi dan satu kunci untuk dekripsi. Kunci untuk enkripsi diumumkan kepada publik oleh karena itu tidak rahasia sehingga dinamakan kunci publik. Sedangkan kunci untuk dekripsi bersifat rahasia sehingga dinamakan kunci privat.

Mode Operasi Enkripsi Blok Cipher

Mode-mode operasi enkripsi digunakan dengan tujuan untuk mengatasi keamanan cara penyandian dan juga mempermudah penyandian dalam algoritma blok cipher. Ada dua mode yang umum digunakan, yaitu:

1. Mode ECB (Electronic Code Book)

   Pada mode ini, setiap blok plainteks dienkrip menjadi satu blok cipher tanpa mempengaruhi blok pesan yang lain. Mode ini adalah cara yang paling sederhana. Kerusakan satu blok data tidak mempengaruhi blok lain. Sifat dasar mode ECB adalah bahwa blok plainteks yang sama akan dikodekan menjadi cipher yang sama.

2. Mode CBC (Chiperblock Chaining) Pada mode ini, blok plainteks yang sama akan dienkrip kedalam blok cipher yang berbeda. Karena cipher tidak hanya bergantung pada blok plainteks yang berhubungan melainkan bergantung pada cipher sebelumnya

2.2 Algoritma Blowfish

Enkripsi Algoritma Blowfish

Blowfish termasuk dalam enkripsi block Cipher 64-bit dengan panjang kunci yang bervariasi antara 32-bit sampai 448-bit. Algoritma Blowfish terdiri atas dua bagian yaitu Pembangkitan sub-kunci (Key-Expansion) dan Enkripsi Data. Enkripsi Data terdiri dari iterasi fungsi sederhana (Feistel Network) sebanyak 16 kali putaran. Semua operasi adalah penambahan (addition) dan XOR pada variabel 32-bit.

Pada algoritma Blowfish, digunakan banyak sub-key. Kunci-kunci ini harus dihitung atau dibangkitkan terlebih dahulu sebelum dilakukan enkripsi atau dekripsi data. Pada jaringan feistel, Blowfish memiliki 16 iterasi, masukannya adalah 64-bit elemen data atau sebut saja "X". Untuk melakukan proses enkripsi langkah-langkahnya adalah sebagai berikut:

1. Bagi X menjadi dua bagian yang masing-masing terdiri dari 32-bit yaitu X_L dan X_R.

2. For i = 1 to 16 :

   (a) $X_L = X_L$ Xor P(i)

   (b) $X_R = F(X_L)$ Xor X_R

   (c) Tukar X_L dan X_R

3. Setelah iterasi keenambelas, tukar X_L dan X_R lagi untuk membatalkan pertukaran terakhir.

4. Lalu lakukan :

   (a) $X_R = X_R$ Xor P17

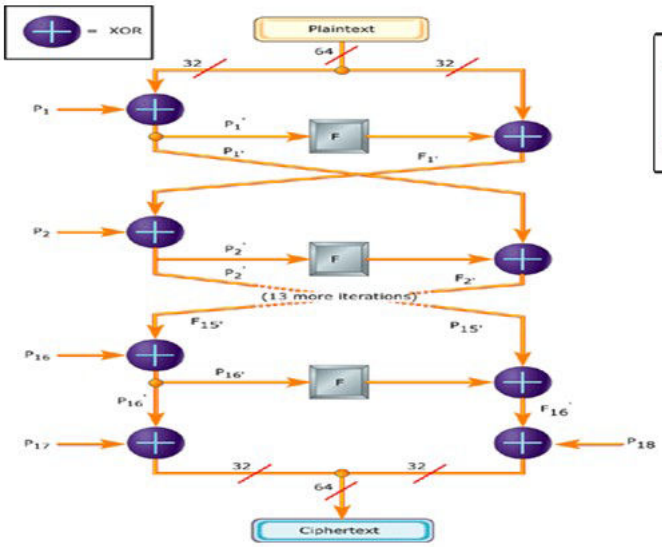
   (b) $X_L = X_L$ Xor P18

5. Terakhir, gabungkan kembali X_L dan X_R untuk mendapatkan cipherteks.

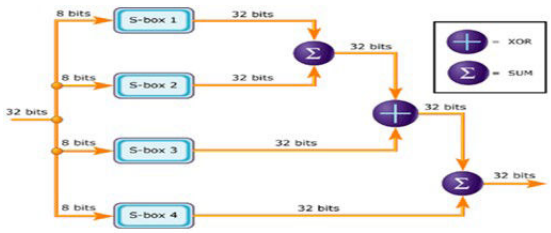
Berikut adalah gambaran proses algoritma Blowfish:

Fungsi F adalah dengan cara membagi X_L menjadi empat bagian 8-bit yaitu: a,b,c dan d seperti gambar 2 maka:

$F(X_L) = (((S_1,a + S_2,b)$ XOR $S_3,c) + S_4,d)$



Gambar 1: Jaringan Feistel Algoritma Blowfish



Gambar 2: Skema Fungsi F

Dekripsi Algoritma Blowfish

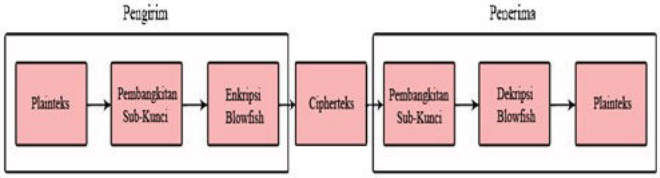
Algoritma Blowfish memiliki keunikan dalam hal proses dekripsi, yaitu proses dekripsi dilakukan dengan urutan yang sama persis dengan proses enkripsi, hanya saja pada proses dekripsi P1, P2, . . . , P18 digunakan dalam urutan yang terbalik.

Pembangkitan Sub-Kunci (Sub-Key)

Cara menghitung atau membangkitkan sub-key:

1. Inisialisasi P-array yang pertama dan juga empat S-box, berurutan, dengan string yang telah pasti. String tersebut terdiri dari digit-digit heksadesimal.

2. XOR P1 dengan 32-bit pertama dari kunci, XOR P2 dengan 32-bit kedua dari kunci, dan seterusnya untuk seluruh bit dari kunci (sampai P18). Ulangi siklus seluruh bit kunci secara berurutan sampai seluruh P-array telah di-XOR-kan dengan bit-bit kunci.



Gambar 3: Blok Diagram Pemodelan Sistem

3. Enkripsikan string yang seluruhnya nol sebanyak 64-bit dengan algoritma Blowfish menggunakan subkey baru dari langkah 2, gantikan seluruh elemen dari P-Box dan kemudian keempat S-box secara berurutan dengan hasil keluaran algoritma Blowfish yang terus menerus berubah.

3 Metode Penelitian

3.1 Pemodelan Sistem

Pada gambar 3, akan dilakukan proses pengiriman data dari satu komputer ke komputer lain. Pada sisi pengirim, data yang dikirimkan akan di enkripsi kemudian dikirim melalui jaringan, setelah sampai di penerima data akan didekripsi kembali. Proses pengiriman ini bisa berlangsung dua arah, dalam arti baik sisi pengirim maupun sisi penerima dapat melakukan pengiriman data serta proses enkripsi dan dekripsi.

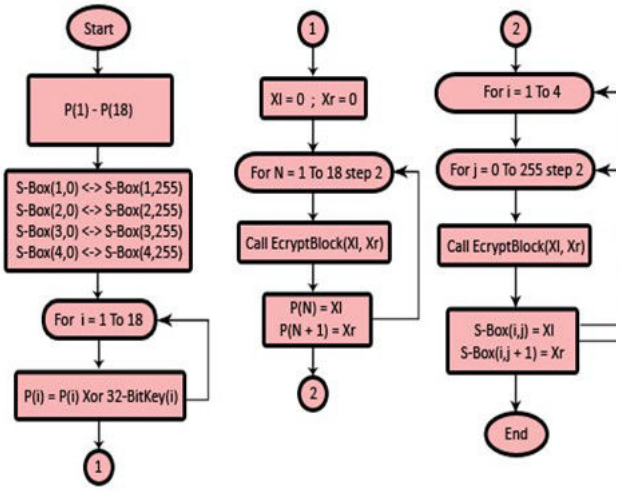
3.2 Proses-Proses Simulasi

Proses Pembacaan Data

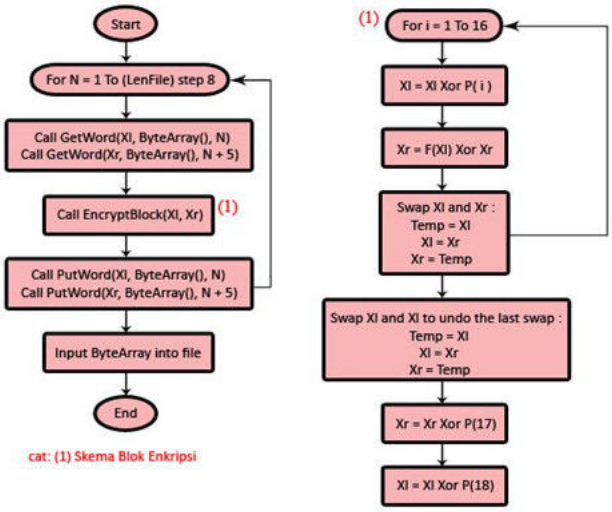
Dalam simulasi sistem ini input yang akan digunakan adalah informasi file berupa teks, citra, audio dan video. Proses ini meliputi proses pembacaan ukuran file sebelum dan sesudah enkripsi atau dekripsi, proses penambahan padding untuk data yang tidak berkelipatan 64-Bit agar menjadi kelipatan 64-Bit,hal ini dikarenakan algoritma blowfish memproses data 64-Bit blok, kemudian padding akan dibuang kembali saat proses dekripsi.

Proses Pembangkitan Sub-Kunci

Pada Algoritma Blowfish digunakan banyak Sub-Kunci. Kunci-kunci ini harus dibangkitkan terlebih dahulu sebelum dilakukan enkripsi atau dekripsi data. Berikut adalah gambaran proses pembangkitan Sub-Kunci



Gambar 4: Proses Pembangkitan Sub-Kunci 4



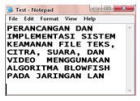
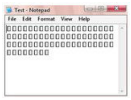
Gambar 5: Skema Proses Enkripsi Blowfish

Proses Enkripsi Dan Dekripsi

Pada gambar 5, Algoritma Blowfish memiliki 16 iterasi, masukannya adalah 64-bit elemen data atau sebut saja "X". Langkah-langkah untuk melakukan proses enkripsi adalah sebagai berikut:

1. Bagi X menjadi dua bagian yang masing-masing terdiri dari 32-Bit yaitu Xl dan Xr.
2. Untuk iterasi i = 1 sampai 16 lakukan :
   (a) Xl = Xl Xor P(i)
   (b) Xr = F(Xl) Xor Xr
   (c) Tukar Xl dan Xr
3. Setelah iterasi ke-enambelas, tukar Xl dan Xr lagi untuk membatalkan proses pertukaran terakhir.
4. Lalu lakukan :
   (a) Xr = Xr Xor P(17)
   (b) Xl = Xl Xor P(18)
5. Terakhir, gabungkan kembali Xl dan Xr dalam 64-Bit blok data untuk mendapatkan cipherteks.
6. Ulangi langkah diatas sampai seluruh blok dari data terenkripsi, kemudian masukan cipherteks kedalam file tujuan.

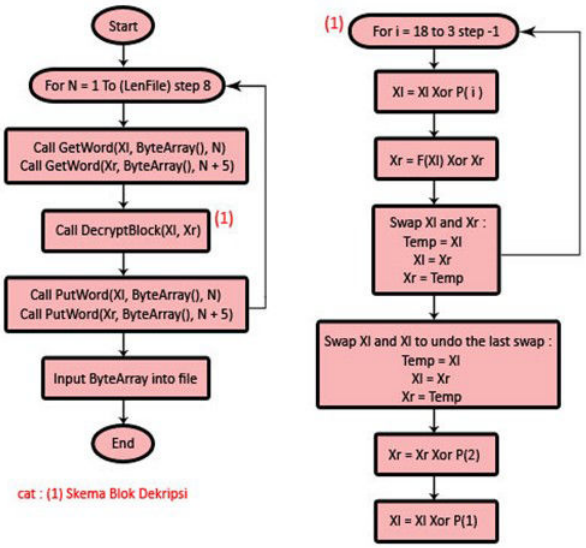
Berikut ini adalah contoh hasil dari proses sebelum dan sesudah enkripsi :

Jenis File	SEBELUM	SESUDAH
Teks		
Citra		
Audio		

Tabel 1: Contoh Hasil Enkripsi Pada File

Proses Dekripsi Blowfish

Proses dekripsi pada gambar 6, hampir sama dengan proses enkripsi hanya saja Sub-Kunci P(1) sampai P(18) digunakan dalam urutan terbalik yaitu P(1) menjadi P(18), P(2) menjadi P(17) dan seterusnya. Didalam proses dekripsi cipherteks diubah kembali kedalam bentuk plainteks atau kondisi semula sebelum dienkripsi.

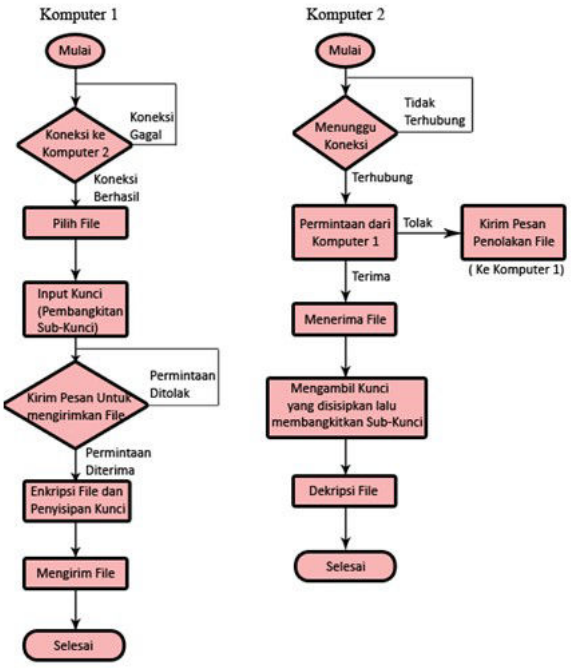


cat : (1) Skema Blok Dekripsi

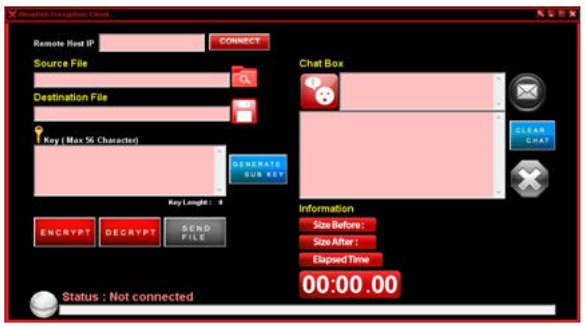
Gambar 6: Skema Proses Dekripsi Blowfish

Proses Pengiriman Data

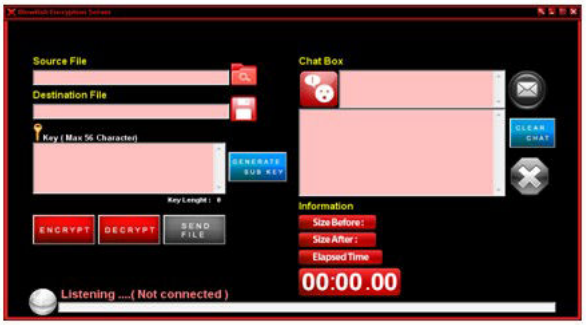
Dalam simulasi ini dilakukan proses pengiriman data dari komputer pengirim ke komputer tujuan yang berada dalam satu jaringan LAN. Pada sisi pengirim data yang akan dikirim dienkripsi terlebih dahulu baru dikirimkan ke komputer tujuan. Saat data diterima di komputer tujuan, data didekripsi kembali sehingga menjadi data semula.



Gambar 7: Skema Proses Pengiriman Data



Gambar 8: Tampilan Aplikasi Dalam Mode Client



Gambar 9: Tampilan Aplikasi Dalam Mode Server

4 Hasil dan Diskusi

4.1 Graphical User Interfaces (GUI)

Tampilan Aplikasi dalam Mode Client

Pada gambar 8 diatas terlihat aplikasi dalam mode Client, dimana ditambahkan fasilitas chatting agar user dapat saling berkomunikasi

Tampilan Aplikasi dalam Mode Server

Tampilan pada mode server tidak terlalu berbeda dengan tampilan pada mode client. Pada mode server tidak terdapat text box untuk meng-input IP Address dan tombol connect. Selain dari pada itu semua tampilan memiliki bentuk dan fungsi yang sama.

4.2 Pengujian Sistem

Parameter Pengujian Sistem

1. Pengiriman Data, yaitu untuk menganalisa proses pertukaran data dari komputer pengirim ke komputer penerima dilakukan dengan bantuan software Socket Sniff yang berfungsi untuk menyadap port aplikasi yang digunakan dan melihat paket data yang keluar atau masuk dari port tersebut.

2. Waktu Proses, yaitu pengukuran waktu proses

(a) Sebelum

(b) Sesudah

Gambar 10: Data yang dikirimkan sebelum dan sesudah enkripsi

enkripsi dan dekripsi dilakukan dengan beberapa percobaan. Kemudian seluruh hasil percobaan dijumlahkan lalu dibagi dengan banyaknya percobaan (n).

$$rata - rata\ waktu = \frac{\sum_{i=0}^{n} waktu}{n}$$

3. Waktu Pemecahan Kunci, yaitu mengukur seberapa lama waktu yang dibutuhkan untuk mencoba-coba semua kemungkinan kunci.

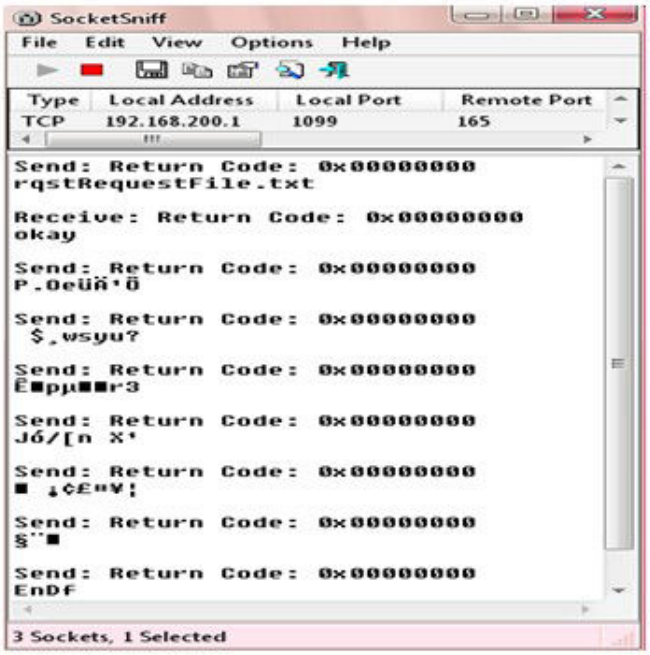
$$brute\ force\ attack = \frac{0{,}5\ x\ waktu\ enkripsi\ x\ jumlah\ kemungkinan\ k}{364x24x3600}$$

4. Avalanche Effect, merupakan salah satu cara untuk menentukan baik atau tidaknya suatu algoritma kriptografi, dimana akan diketahui seberapa besar perubahan bit yang terjadi pada cipherteks akibat proses enkripsi. Semakin besar avalanche effect akan semakin baik algoritma kriptografi tersebut. Cara menghitung avalanche effect sebagai berikut :

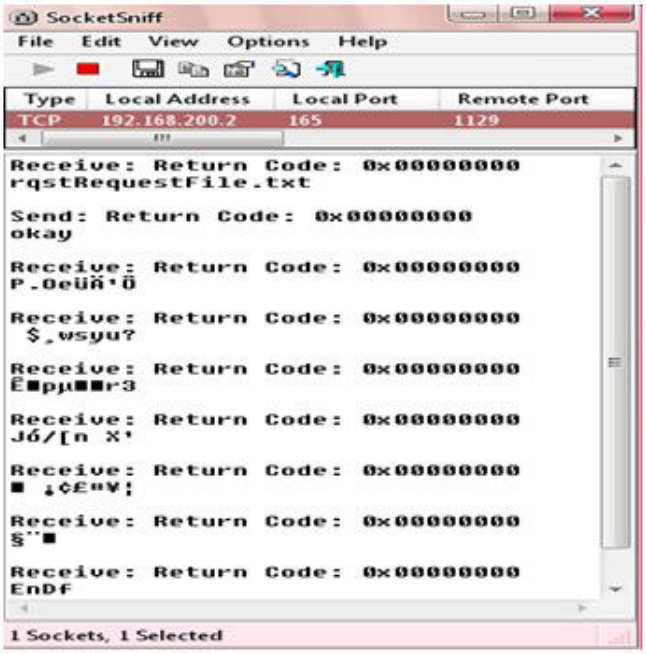
$$Avalanche\ Effect = \left(\frac{Besar\ peruibahan\ Bit}{Jumlah\ Keseluruhan\ Bit}\ x100\%\right)$$

4.3 Analisa Data Hasil Pengujian Sistem

Analisa Proses Pengiriman Data



Gambar 11: Pengiriman Data Pada Sisi Pengirim



Gambar 12: Proses Penerimaan Data Pada Sisi Penerima

No	Ukuran Kunci	Waktu Enkripsi (s)				Waktu Dekripsi (s)			
		500 KB	1000 KB	1500 KB	2000 KB	500 KB	1000 KB	1500 KB	2000 KB
1	64 Bit	0.052	0.102	0.262	0.312	0.053	0.102	0.264	0.313
2	200 Bit	0.052	0.103	0.263	0.312	0.053	0.103	0.264	0.313
3	320 Bit	0.053	0.103	0.262	0.312	0.054	0.104	0.264	0.314

Tabel 2: RATA-RATA WAKTU ENKRIPSI DAN DEKRIPSI BLOWFISH UNTUK FILE TEKS

No	Ukuran Kunci	Waktu Enkripsi (s)				Waktu Dekripsi (s)			
		500 KB	1000 KB	1500 KB	2000 KB	500 KB	1000 KB	1500 KB	2000 KB
1	64 Bit	0.052	0.102	0.262	0.312	0.053	0.102	0.265	0.313
2	200 Bit	0.052	0.102	0.262	0.312	0.053	0.103	0.265	0.313
3	320 Bit	0.052	0.102	0.262	0.312	0.054	0.104	0.266	0.314

Tabel 3: RATA-RATA WAKTU ENKRIPSI DAN DEKRIPSI BLOWFISH UNTUK FILE CITRA

No	Ukuran Kunci	Waktu Enkripsi (s)				Waktu Dekripsi (s)			
		500 KB	1000 KB	1500 KB	2000 KB	500 KB	1000 KB	1500 KB	2000 KB
1	64 Bit	0.051	0.102	0.261	0.312	0.053	0.103	0.265	0.313
2	200 Bit	0.050	0.103	0.260	0.312	0.053	0.104	0.263	0.313
3	320 Bit	0.051	0.103	0.262	0.312	0.053	0.104	0.262	0.314

Tabel 4: . RATA-RATA WAKTU ENKRIPSI DAN DEKRIPSI BLOWFISH UNTUK FILE AUDIO

No	Ukuran Kunci	Waktu Enkripsi (s)				Waktu Dekripsi (s)			
		1000 KB	1500 KB	2000 KB	3000KB	1000 KB	1500 KB	2000 KB	3000KB
1	64 Bit	0.102	0.262	0.312	0.412	0.106	0.265	0.313	0.416
2	200 Bit	0.102	0.262	0.312	0.412	0.106	0.265	0.313	0.417
3	320 Bit	0.102	0.262	0.312	0.412	0.107	0.266	0.314	0.415

Tabel 5: . RATA-RATA WAKTU ENKRIPSI DAN DEKRIPSI BLOWFISH UNTUK FILE VIDEO

Dari hasil analisa pengiriman data diatas, membuktikan bahwa data yang dikirim benar-benar terenkripsi saat berada dalam jaringan. Serta kunci yang telah disisipkan pada cipherteks tidak terlihat lagi karena kunci sebelum disisipkan disamarkan terlebih dahulu, sehingga jika ada yang menyadap data dan membaca data tersebut, informasi kunci tidak akan terlihat jelas bahwa didalam data tersebut terdapat kunci yang disisipkan.

Analisa Pengukuran Waktu Proses

Berikut adalah hasil analisa data dari pengukuran waktu proses enkripsi dan dekripsi dengan kunci bervariasi yaitu 8 karakter (64 Bit), 25 karakter (200 Bit), dan 40 Karakter (320 Bit).

Dari hasil pengukuran diatas terlihat bahwa rata-rata waktu enkripsi lebih cepat dibandingkan waktu dekripsinya. Hal ini disebabkan karena adanya penambahan bit tambahan atau padding pada saat enkripsi, sehingga menyebabkan proses dekripsi membutuhkan waktu tambahan pula. Kemudian dapat dilihat juga bahwa semakin besar ukuran file maka semakin besar pula waktu enkripsi dan dekripsinya.

Analisa Pengukuran Waktu Pemecahan Kunci

Algoritma blowfish adalah algoritma simetri dengan kunci bervariasi dengan batas maksimum kunci adalah 56 karakter atau 448-Bit. Hasil perhitungan waktu pemecahan kunci menggunakan brute force attack untuk file berukuran 3000 KB :

$BruteForceAttack = \frac{0,5x0,412x2^{448}}{365x24x3600} = 4,7479x10^{126}tahun$

Analisa Perbandingan Algoritma Blowfish Dan DES

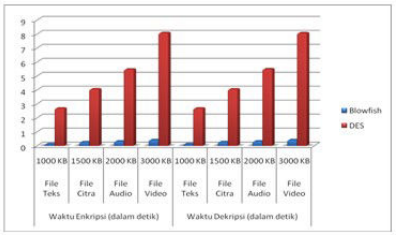
Hasil analisa dari perbandingan Algoritma Blowfish terhadap Algoritma DES dari segi waktu enkripsi, waktu dekripsi, waktu pemecahan kunci dan avalanche effect :

- Waktu Enkripsi dan Dekripsi
- Waktu Pemecahan Kunci

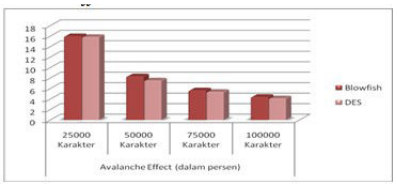
  Blowfish, memiliki 2448 key space.

$BruteForceAttack = \frac{0,5x0,412x2^{448}}{365x24x3600} = 4,7479x10^{126}tahun$

DES, memiliki 256 key space.



Gambar 13: Perbandingan Waktu Enkripsi Dan Waktu Dekripsi Untuk Algoritma Blowfish dan DES



Gambar 14: Perbandingan Avalanche Effect Untuk Algoritma Blowfish dan DES

$$BruteForceAttack = \frac{0{,}5x8{,}076x2^{56}}{365x24x3600} = 9,2266x10^{126}tahun$$

- Avalanche Effect

Dari hasil analisa didapat bahwa waktu enkripsi dan dekripsi menggunakan algoritma blowfish jauh lebih cepat dari pada menggunakan algoritma DES, hal ini dimungkinkan karena proses pada algoritma DES lebih banyak dibandingkan proses pada algoritma Blowfish. Kemudian waktu pemecahan kunci pada algoritma Blowfish membutuhkan waktu yang lebih lama dari DES, serta avalanche effect dari algoritma Blowfish lebih besar dari pada algoritma DES walaupun dengan perbedaan yang kecil sehingga dapat disimpulkan algoritma Blowfish lebih baik untuk diterapkan pada sistem keamanan file dibandingkan menggunakan algoritma DES baik itu dilihat dari waktu enkripsi, waktu dekripsi, waktu pemecahan kunci dan avalanche effect.

5 Kesimpulan

Berdasarkan hasil studi dan percobaan dapat disimpulkan sebagai berikut :

1. Sistem yang telah dirancang mampu mengirimkan data terenkripsi melalui jaringan LAN dengan aman. Hal ini dapat dilihat dari informasi data dan kunci yang melintas di jaringan dalam keadaan terenkripsi serta waktu pemecahan kunci yang sangat lama yaitu 4,7479x10126 tahun untuk panjang kunci 448 Bit.
2. Sistem yang telah dirancang mampu mengatasi kelemahan pada konsep algoritma simetri dalam hal pengiriman data, dimana kunci untuk enkripsi dan dekripsi menggunakan kunci yang sama. Hal itu menyebabkan pihak penerima harus mengetahui kunci untuk dekripsi, serta pengiriman kunci untuk dekripsi dari pihak pengirim tidak dapat menggunakan algoritma simetri lagi karena harus dibuka dengan kunci yang sama dan begitu seterusnya. Dengan begitu sistem yang telah dirancang ini dapat menjawab persoalan tersebut.
3. Waktu proses enkripsi dan dekripsi dari algoritma Blowfish dipengaruhi oleh besar ukuran file. Semakin besar ukuran file maka akan semakin lama waktu prosesnya. Sedangkan format file dan panjang kunci yang digunakan tidak mempengaruhi waktu proses enkripsi dan dekripsi, karena algoritma blowfish memproses bit-bit yang ada pada file dan pembangkitan sub kunci dilakukan sebelum proses enkripsi atau dekripsi dilakukan.
4. Algoritma Blowfish lebih baik untuk diterapkan pada sistem keamanan file dibandingkan menggunakan algoritma DES. Hal ini dapat dilihat dari hasil pengujian yaitu waktu enkripsi untuk file berukuran 3000 KB adalah 0,412 detik untuk Blowfish dan 8,076 detik untuk DES, sedangkan waktu dekripsinya adalah 0,414 detik untuk Blowfish dan 8,076 detik untuk DES. Waktu pemecahan kunci untuk Blowfish adalah 4,7479x10126 tahun dan waktu pemecahan kunci untuk DES adalah 9,2266x109 tahun. Selanjutnya, besarnya avalanche effect untuk data dengan panjang 25000 karakter adalah 16,01 % untuk Blowfish dan 15,86% untuk DES.

Pustaka

[1] Rinaldi Munir. Data encryption standart (des). Technical report, Program Studi Teknik Informatika Institut Teknologi Bandung., 2005.

[2] Man Young. Rhee. *Cryptography and Secure Communications*. McGraw- Hill., 1994.

[3] Bruce. Schneider. Description of a new variable-length key, 64-bit block cipher (blowfish), fast software encryption. In *Cambridge Security Workshop Proceedings*, 1994.

[4] Bruce Schneier. The blowfish encryption algorithm. In – *One Year Later. Dr. Dobb's Journal.*, 1995.

[5] Bruce Schneier. *Aplied Cryptography 2nd*. 1996.

[6] W Stalling. *Cryptography and Network Security, Principle and Practice 2rd Edition,*. Pearson Education, Inc., 1998.