BAB III

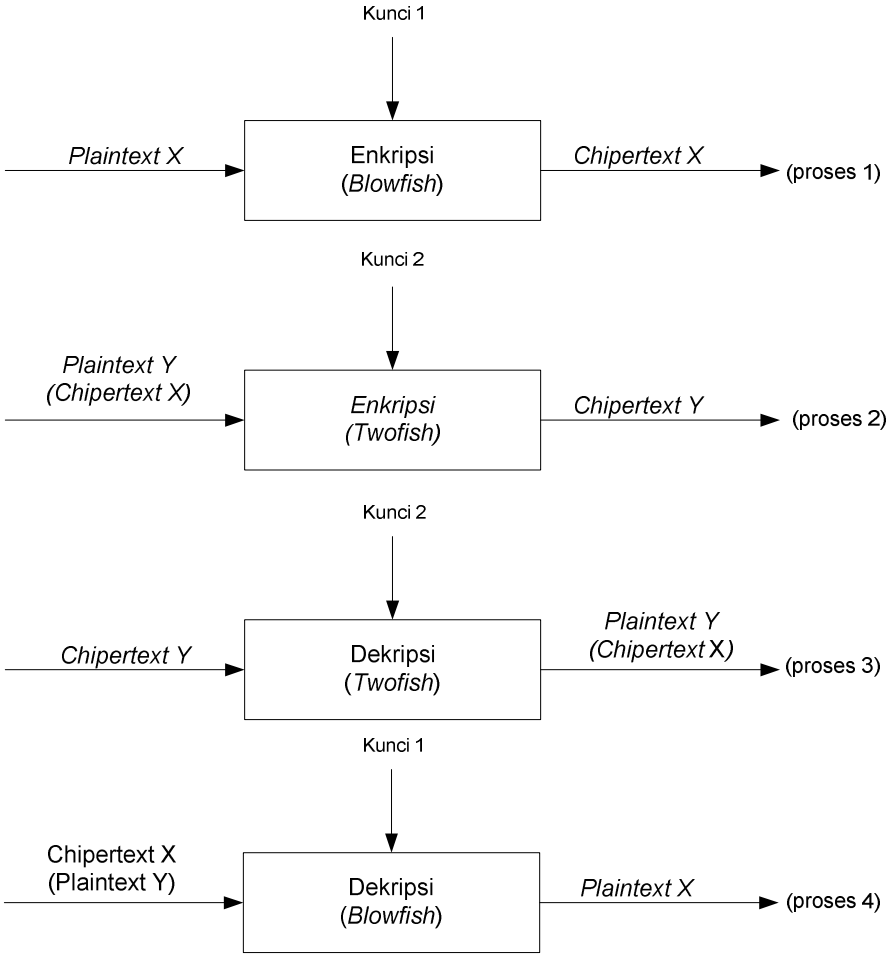
ANALISIS DAN PERANCANGAN

3.1 Analisis

Analisis adalah penguraian dari suatu pembahasan, dalam hal ini pembahasan mengenai aplikasi keamanan pengiriman data pada jaringan *Local Area Network* (LAN), yang berguna untuk mengetahui apa saja yang dapat dijadikan isi dari sebuah aplikasi dan perangkat yang dapat digunakan dalam perancangan aplikasi yang akan dibuat.

3.1.1 Analisis permasalahan

Dalam pembahasan kriptografi yang sedang penulis bahas yaitu mengenai mengamankan pengiriman sebuah data pada jaringan *Local Area Network* (LAN) dengan menggunakan algoritma kriptografi *Blowfish dan Twofish,* atau kombinasi kedua algoritma tersebut dalam aplikasi kriptografi. Berikut dibawah ini analisa rancangan dari permasalahan yang sedang di bahas yang dijelaskan pada gambar 3.1 :



Gambar 3.1 analisis permasalahan

Dari gambar 3.1 diatas dapat dijelaskan sebagai berikut :

1. Proses 1 (Enkripsi *Blowfish*) :
    a. *Plaintext* X, merupakan *file* yang akan dienkripsi
    b. Enkripsi *plaintext* X, menggunakan enkripsi *Blowfish* dan memasukkan kunci 1 (kunci yang digunakan pada saat enkripsi *Blowfish*)
    c. *Chipertext* X, merupakan hasil enkripsi dari *plaintext* X

2. Proses 2 (Enkripsi *Twofish*) :
    a. *Plaintext* Y, merupakan hasil enkripsi (*chipertext* X) dari *plaintext* X
    b. Enkripsi *plaintext* Y, menggunakan enkripsi *Twofish* dan memasukkan kunci 2 (kunci yang digunakan pada saat enkripsi *Twofish*)
    c. *Chipertext* Y, merupakan hasil enkripsi dari *plaintext* Y (*chipertext* X)
3. Proses 3 (Dekripsi *Twofish*) :
    a. *Chipertext* Y, merupakan hasil enkripsi *Twofish* yang akan didekripsi
    b. Dekripsi *Twofish*, menggunakan dekripsi *Twofish* dengan menggunakan kunci 2 (kunci yang digunakan pada saat enkripsi *Twofish*)
    c. *Plaintext* Y (*chipertext* X), merupakan hasil dekripsi dari *chipertext* Y
4. Proses 4 (Dekripsi *Blowfish*) :
    a. *Chipertext* X*(plaintext* Y*)*, merupakan hasil enkripsi *Blowfish* yang akan didekripsi
    b. Dekripsi *Blowfish*, menggunakan dekripsi *Blowfish* dengan menggunakan kunci1 (kunci yang digunakan pada saat enkripsi *Blowfish*)
    c. *Plaintext* X, merupakan hasil dari dekripsi *chipertext* X *(paintext* Y*)*

3.1.2 Kebutuhan Perangkat

Berikut dibawah ini perangkat yang dapat mendukung dalam perancangan aplikasi yang akan dirancang, yaitu :

3.1.2.1 Perangkat Keras

Perangkat keras yang dibutuhkan dalam pembuatan aplikasi kriptografi adalah sebagai berikut :

1. Pentium III 800 Mhz
2. RAM 128 MB
3. *Hardisk* 10 GB
4. Sebagai alat koneksi jaringan memakai Kabel UTP (*Unshielded Twisted Pair*) dan konektor RG 45

3.1.2.2 Perangkat Lunak

Perangkat lunak atau *software* yang digunakan dalam program aplikasi kriptografi adalah :

1. *Visual Basic* 6.0 (perancangan aplikasi)
2. *Windows* XP (sistem operasi)

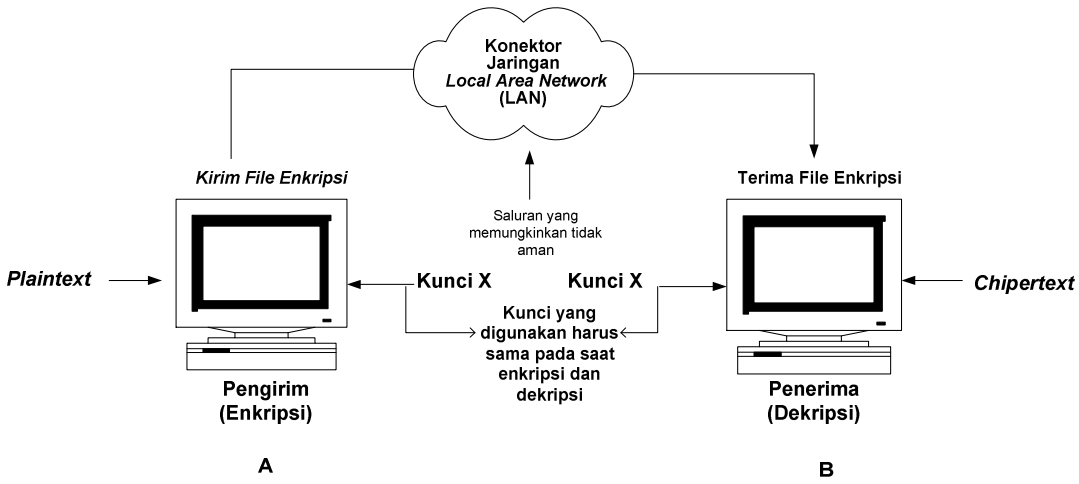
3.1.3 Analisis Data

Analisis data merupakan tahap dimana dilakukannya analisis terhadap data-data apa saja yang diolah dalam sistem atau prosedur sebuah aplikasi, dalam hal ini data yang akan di enkripsi pada aplikasi kriptografi adalah berupa *file Txt, Doc,* Jpg, Mp3 dan Mpeg.

3.1.4 Analisis pengiriman data dan keamanan Jaringan

Pada permasalahan yang sedang dibahas yaitu mengenai aplikasi keamanan pengiriman data menggunakan jaringan *Local Area Network* (LAN). Jadi proses *transfer* data dari komputer satu ke komputer lain menggunakan sistem jaringan *Local Area Network* (LAN). Dalam hal ini dengan adanya keamanan jaringan komputer dengan menggunakan algoritma kriptografi *Blowfish* dan *Twofish* dapat melindungi pada saat transaksi pengiriman data pada jaringan *Local Area Network* (LAN). Keamanan data pada suatu jaringan *Local Area Network* (LAN) adalah sesuatu yang sangat penting, karena untuk menghindari data yang kita kirim di gunakan oleh orang – orang yang tidak berhak atau berkepentingan.

Ditinjau dari masalah yang dihadapi, maka perlu dibangun suatu aplikasi keamanan pengiriman data dengan menggunakan algoritma kriptografi *Blowfish* dan *Twofish* pada saat mengirimkan data melalui jaringan *Local Area Network* (LAN). Berikut dibawah ini merupakan gambaran dari pengiriman data melalui jaringan *Local Area Network* (LAN) dan permasalahan yang memungkinkan terjadi pada media jaringan *Local Area Network* (LAN):



Gambar 3.2 pengiriman data melalui jaringan *Local Area Network* (LAN)

Dari gambar 3.2 diatas dapat dijelaskan sebagai berikut **:**

Gambar 3.2 diatas adalah menjelaskan bagaimana melakukan pengiriman sebuah data dari komputer A dan B melalui media jaringan *Local Area Network* (LAN), dalam gambar 3.2 diatas menjelaskan bagaimana melakukan pengiriman data melalui media jaringan *Local Area Network* (LAN). Untuk melakukannya diperlukan sebuah data yang akan dikirim dengan dienkripsi serta memasukkan kunci terlebih dahulu, kemudian dikirimkan menurut alamat tujuan yang akan dikirim, dan penerima dari data yang terenkripsi tersebut dapat mendekripsi data tersebut jika mengetahui kunci yang dipakai, maka dari itu penggunaan kunci sebelumnya harus disepakati dan diketahui oleh pengirim dan penerima.

3.1.5 Analisis Pengguna

Latar belakang pengguna yang akan memakai program aplikasi yang akan dibuat, terbagi menjadi dua pengguna yaitu pengirim pesan dan penerima pesan, serta memiliki kriteria pengguna sebagai berikut :

1. Memiliki kemampuan dasar dalam pengoperasian komputer
2. Mengenal sistem kriptografi
3. Mengenal jaringan *Local Area Network* (LAN)

3.2 Langkah – Langkah Penyelesaian

Dari permasalahan diatas, maka penulis mencoba untuk membuat sebuah aplikasi yang berguna untuk mengamankan sebuah data pada saat pengiriman data melalui media jaringan *Local Area Network* (LAN) dengan menggunakan kombinasi algoritma kriptografi *Blowfish* dan *Twofish*. Langkah – langkah simulasi dalam penyelesaian masalah diatas yaitu :

3.2.1 Pengirim *File* / Data

Orang yang mengirim *file*, akan melakukan langkah – langkah sebagai berikut :

1. memilih suatu *file* yang akan dienkripsi dan dikirim
2. Memasukkan kunci 1 dan kunci 2yang akan menjadi kunci pada saat proses enkripsi
3. Memasukkan IP *Address* tujuan pengiriman

4. Mengkoneksikan jaringan
5. Mengirim *file* enkripsi kombinasi *blowfish* dan *twofish*

3.2.2 Penerima *File* / Data

Berikut langkah – langkah yang akan dilakukan oleh penerima pesan atau *file*, yaitu :

1. Mengkoneksikan jaringan
2. Mengambi *file* enkripsi (*blowfish* dan *twofish*) yang akan di dekripsi
3. Memasukkan kunci 2 dan kunci 1 untuk melakukan proses dekripsi
4. Melakukan dekripsi

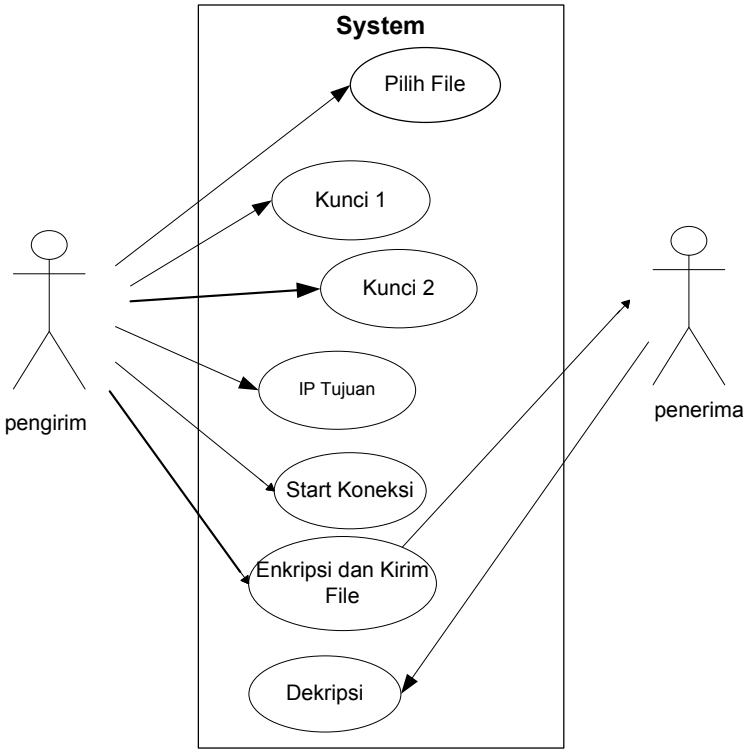
3.3 Perancangan Sistem

Perancangan sistem merupakan bagian dari metodologi pengembangan suatu perangkat lunak yang dilakukan setelah melalui tahapan analisis. Perancangan bertujuan untuk memberikan gambaran secara terperinci. Perancangan sistem merupakan tahap lanjutan dari analisis sistem, dimana pada perancangan sistem digambarkan rancangan sistem yang akan dibangun sebelum dilakukan pengkodean kedalam suatu bahasa pemrograman.

Dalam perancangan *system* ini menggunakan *tool* UML (*Unified Modelling Language*) sebagai rancangan dari sebuah model *system* yang akan dibangun. Perancangan UML (*Unified Modelling Language*) ini mencakupi perancangan *use case, sequence* diagram, *class* diagram dan *activity* diagram.

1. *Use case* diagram

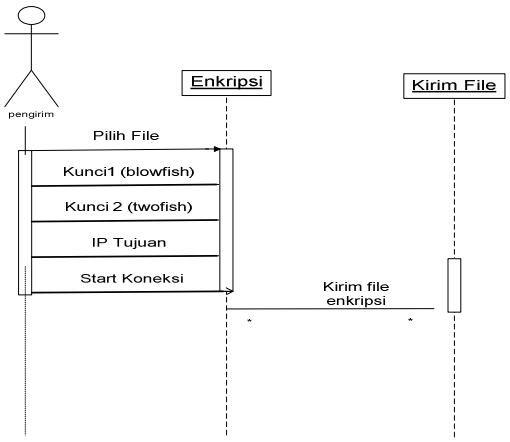
   Dalam *use case* menggambarkan interaksi seorang aktor dari sebuah *system* yang akan dibangun. Gambar 3.3 dibawah ini merupakan diagram rancangan *use case* :



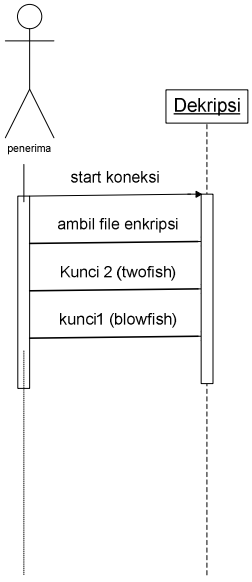
Gambar 3.3 Diagram *Use Case*

2. *Sequence* Diagram

   Diagram yang menggambarkan langkah – langkah yang dilakukan oleh *actor* maupun system. Gambar 3.4 dan 3.5 dibawah ini merupakan rancangan dari *sequence* diagram :



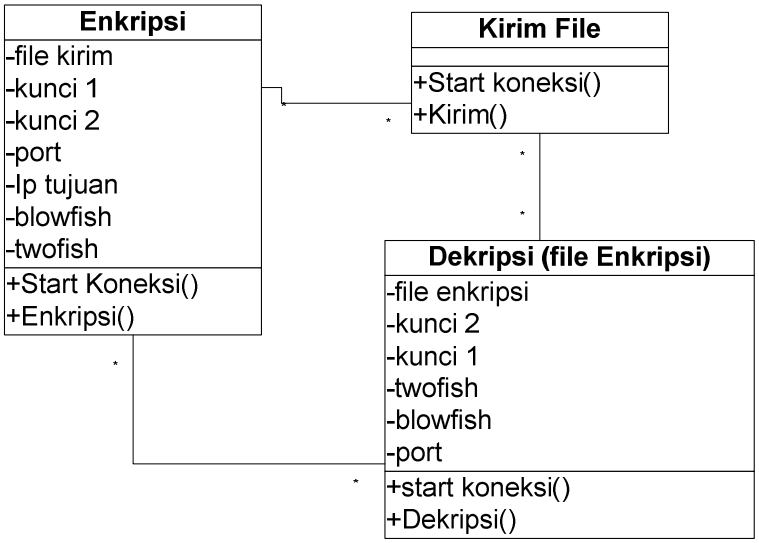
Gambar 3.4 Diagram *Sequence* enkripsi dan kirim



Gambar 3.5 Diagram *sequence* dekripsi *file*

3. *Class* Diagram

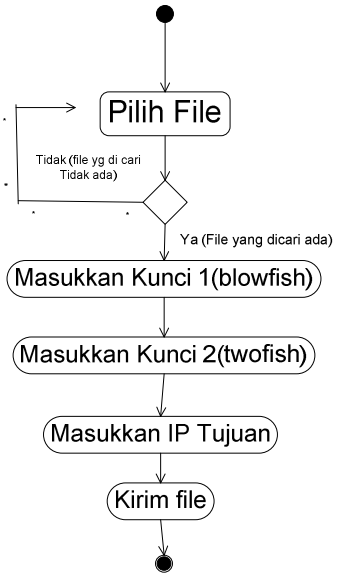
Menggambarkan keadaan suatu *system* (*atribbut*), dan memberikan pelayanan untuk menyelesaikan keadaan tersebut (metoda). Gambar 3.6 dibawah ini merupakan gambaran dari sebuah *class* diagram :



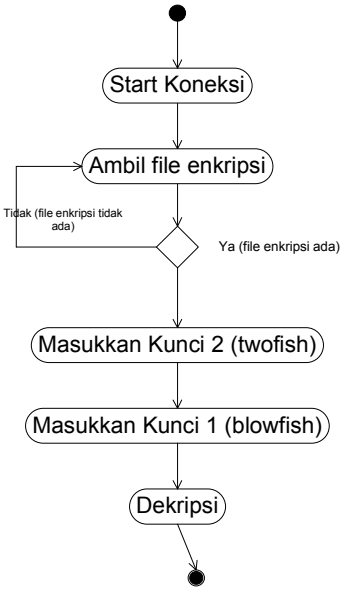
Gambar 3.6 *Class* diagram

4. *Activity* Diagram

Menggambarkan alir dari system yang sedang dirancang. Gambar 3.7 dan 3.8 dibawah ini merupakan bentuk dari *activity* diagram yang akan dirancang :



Gambar 3.7 *Activity* diagram pengiriman *file* enkripsi



Gambar 3.8 *Activity* diagram penerimaan *file* enkripsi (Dekripsi)

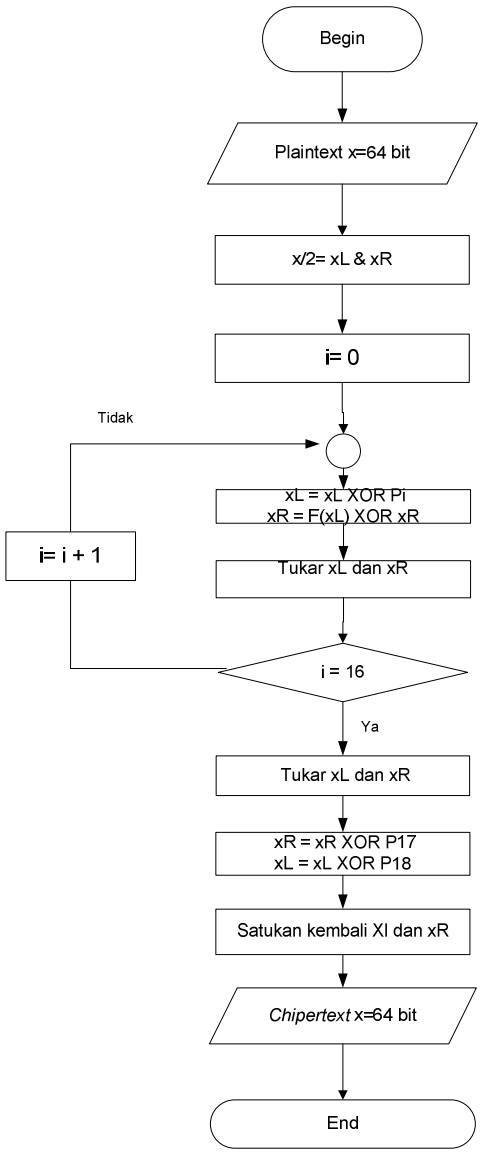
3.4 Perancangan Proses

Tujuan perancangan proses adalah tahap untuk memperbaiki atau meningkatkan efisiensi kerja sistem. Tahap ini sangat penting dalam menentukan baik atau tidaknya hasil perancangan sistem yang diperoleh. Tahap perancangan proses dapat digambarkan sebagai perancangan untuk membangun suatu sistem dan mengkonfigurasikan komponen-komponen perangkat lunak dan perangkat kerasnya sehingga menghasilkan sistem yang lebih baik bagi *user*. Proses yang dirancang diuraikan menjadi beberapa bagian yang dapat membentuk sistem tersebut menjadi satu komponen.

3.4.1 *Flowchart* Proses Enkripsi dan dekripsi *Blowfish*

A. *Flowchart* Enkripsi *Blowfish*

Pada gambar 3.9 dibawah ini merupakan gambar dari proses enkripsi *Blowfish* :



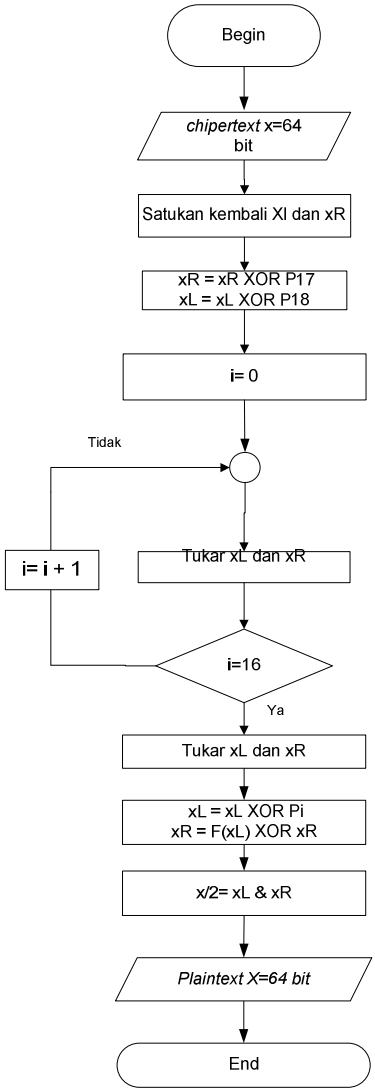
Gambar 3.9 *Flowchart* Proses Enkripsi *Blowfish*

dari gambar 3.9 diatas berhubungan dengan gambar 2.9 struktur algoritma *Blowfish* pada BAB II, dapat diperoleh keterangan sebagai berikut :

1. Memulai proses enkripsi (*plaintext*) dengan x=64 bit
2. X dibagi menjadi 2, XL (x *left* =32 bit) dan XR (x *right* =32 bit)
3. i = 0 merupakan inisial perputaran yang dimulai dari 0
4. Melanjutkan proses selanjutnya, yaitu operasi XL = XL xor Pi dan XR = F(XL) xor XR
5. Menukar hasil dari XL dan XR (XL= XR dan XR=XL)
6. Melakukan perulangan sebanyak 16 kali perulangan, dan melakukan kembali penukaran XL dan XR (i = i+ 1)
7. Pada proses ke-17 lakukan operasi untuk XR = XR xor P17 dan XL = XL xor P18.
8. Menggabungkan kembali XL dan XR (sehingga menjadi x=64 bit)
9. Menghasilkan *chipertext* dengan x=64 bit
10. Selesai

B. *Flowchart* Dekripsi *Blowfish*

Pada gambar 3.10 dibawah ini merupakan gambar dari proses dekripsi *Blowfish* :



Gambar 3.10 *Flowchart* proses dekripsi *Blowfish*

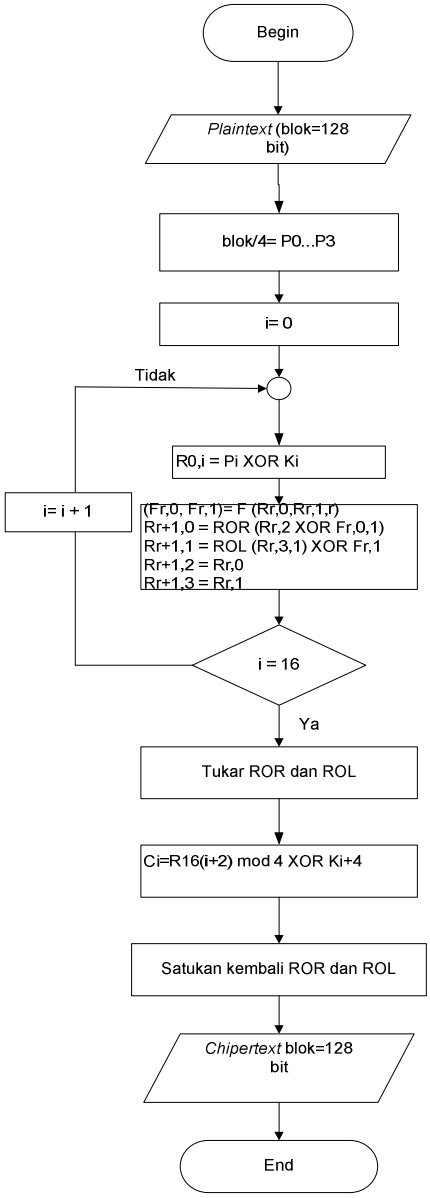
dari gambar 3.10 diatas dapat diperoleh keterangan sebagai berikut :

1. Memulai proses dekripsi (*chipertext)* x= 64 bit (xL= 32 *bit* dan xR=32 *bit*)
2. Menggabungkan XL dan XR (sehingga menjadi x=64 bit)
3. Pada proses ke-17 lakukan operasi untuk XR = XR xor P17 dan XL = XL xor P18.
4. Menukar hasil nilai XL dan XR (XL= XR dan XR=XL)
5. i = 0 merupakan inisial perputaran yang dimulai dari 0
6. Melakukan perulangan sebanyak 16 kali perulangan, dan melakukan kembali penukaran XL dan XR (i= i + 1)
7. Melanjutkan proses selanjutnya, yaitu operasi XL = XL xor Pi dan XR = F(XL) xor XR
8. Menggabungkan kembali XL dan XR (sehingga menjadi x=64 bit)
9. Menghasilkan *Plaintext* dengan x=64 bit
10. selesai

3.4.2 Flowchart proses enkripsi dan dekripsi *Twofish*

A. Enkripsi *Twofish*

Pada gambar 3.11 dibawah ini merupakan gambar dari proses enkripsi *Twofish* :



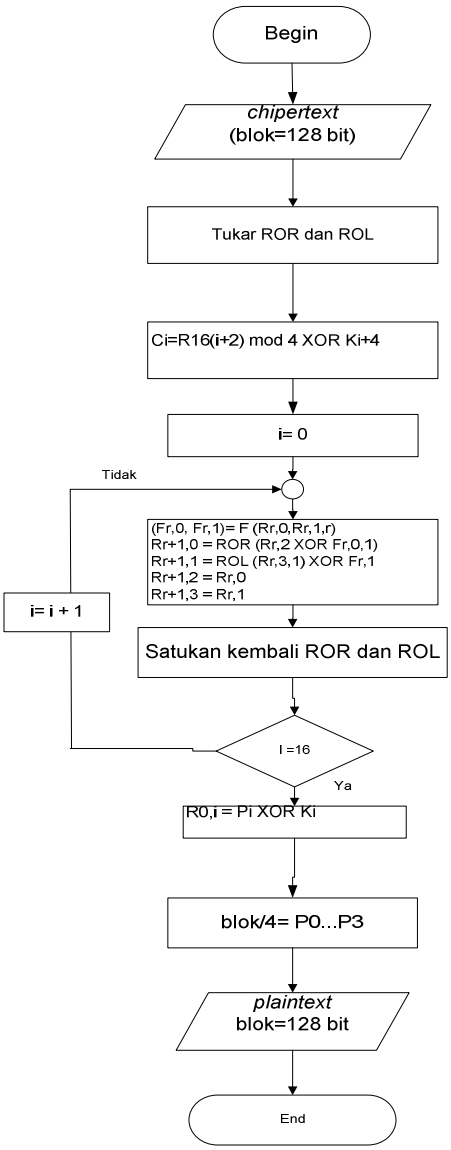
Gambar 3.11 *Flowchart* Enkripsi *Twofish*

Dari gambar 3.11 diatas dapat diperoleh keterangan sebagai berikut :

1. memulai proses enkripsi (*plaintext*) dengan ukuran blok 128 bit
2. blok *plaintext* dibagi menjadi 4 bagian
3. *input whitening* ke 4 bagian tersebut di XOR kan dengan 4 buah kata dari kunci yang diekspansi
4. diulang hingga 16 putaran (i= i + 1), pada setiap putaran kedua kata yang pertama sebagai masukan dari fungsi F, kata ke tiga dilakukan operasi XOR dan dirotasikan kekanan sebanyak 1 bit,kata ke empat dirotasikan kekanan 1 bit dan dilakukan rotasi XOR pada keluaran kedua fungsi F
5. menukarkan hasil ROR dan ROL
6. *output whitening* hasil keluaran dan melakukan operasi XOR dengan 4 buah kata dari kunci yang diekspansi
7. menggabungkan hasil ROR dan ROL, membatalkan proses penukaran pada pertukaran terakhir.
8. Menghasilkan *chipertext*
9. Selesai

B. Dekripsi *Twofish*

Pada gambar 3.12 dibawah ini merupakan gambar dari proses dekripsi *Twofish* :



Gambar 3.12 *Flowchart* Dekripsi *Twofish*

dari gambar 3.12 diatas dapat diperoleh keterangan sebagai berikut :

1. memulai proses dekripsi (*chipertext*) dengan ukuran blok 128 bit
2. menukarkan hasil ROR dan ROL
3. *whitening* hasil keluaran dan melakukan operasi XOR dengan 4 buah kata dari kunci yang diekspansi
4. menggabungkan hasil ROR dan ROL, membatalkan proses penukaran pada pertukaran terakhir.
5. diulang hingga 16 putaran (i=i+ 0), pada setiap putaran kedua kata yang pertama sebagai masukan dari fungsi F, kata ke tiga dilakukan operasi XOR dan dirotasikan kekanan sebanyak 1 bit,kata ke empat dirotasikan kekanan 1 bit dan dilakukan rotasi XOR pada keluaran kedua fungsi F
6. *whitening* ke 4 bagian tersebut di XOR kan dengan 4 buah kata dari kunci yang diekspansi
7. blok *plaintext* dibagi menjadi 4 bagian
8. menghasilkan plaintext blok = 128 bit
9. selesai

3.5 Perancangan *Interface* Aplikasi Kriptografi

3.5.1 *Interface* Aplikasi Enkripsi dan kirim

Gambar 3.13 dibawah ini merupakan rancangan dari tampilan atau *interface* dari aplikasi enkripsi yang akan dibuat adalah :



Gambar 3.13 Rancangan *Interface* aplikasi Enkripsi dan kirim

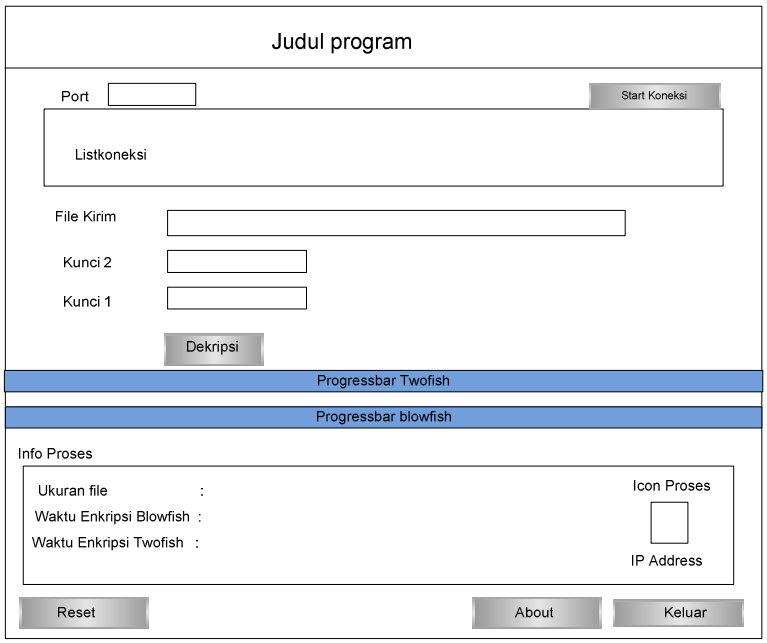
Dibawah ini merupakan keterangan dari gambar 3.13 diatas, yaitu :

1. *Frame* atas, merupakan judul atau nama program aplikasi yang akan dibuat
2. Menampilkan alamat dan nama *file*
3. Kunci 1 (enkripsi *blowfish*)
4. Kunci 2 (enkripsi *twofish*)
5. *Port*, jalur yang digunakan
6. IP *address* tujuan
7. *Button*_pilih file, berfungsi untuk memanggil proses memilih *file* yang akan diproses
8. *Button_start* koneksi, berfungsi untuk memanggil proses koneksi jaringan
9. *List* koneksi, untuk mengetahui koneksi jaringan
10. *Button* kirim, memanggil fungsi proses enkripsi *blowfish* dan *twofish*
11. *Progress blowfish*, merupakan *progress* dari enkripsi *blowfish*
12. *Progress twofish*, merupakan *progress* dari enkripsi *twofish*
13. *Progress* kirim, merupakan *progress* dari kirim enkripsi
14. *Button_Reset*, berfungsi untuk memanggil proses reset semua yang telah di inputkan
15. *Button_About*, berfungsi untuk memanggil form about yang berisi mengenai petunjuk program
16. *Button_Exit*, berfungsi untuk keluar dari program
17. Ukuran *file*, merupakan ukuran awal dari *file* yang akan dienkripsi / dekripsi

18. Waktu proses enkripsi *blowfish*, untuk menampilkan waktu proses enkripsi *blowfish*
19. Waktu proses enkripsi *twofish*, menampilkan waktu proses enkripsi *twofish*
20. Waktu kirim, untuk mengetahui waktu pengiriman *file*
21. *Icon* proses, untuk mengetahui *icon file* yang akan diproses
22. IP *Address*, untuk menampilkan IP *Address* dari computer

3.5.2 *Interface* Aplikasi Dekripsi

Gambar 3.14 dibawah ini merupakan rancangan dari tampilan atau *interface* dari aplikasi dekripsi yang akan dibuat adalah :



Gambar 3.14 *Interface* Aplikasi Dekripsi

Dibawah ini merupakan keterangan dari gambar 3.13 diatas, yaitu :

1. *Frame* atas, merupakan judul atau nama program aplikasi yang akan dibuat
2. *List* koneksi, untuk mengetahui koneksi jaringan
3. *Port*, jalur yang digunakan
4. *Button_start* koneksi, berfungsi untuk memanggil proses koneksi jaringan
5. Menampilkan alamat dan nama *file*
6. Kunci 1 (enkripsi *blowfish*)
7. Kunci 2 (enkripsi *twofish*)
8. *Button*_Ambil file, berfungsi untuk memanggil proses memilih *file* yang akan diproses
9. *Button* dekripsi, memanggil fungsi proses dekripsi *twofish* dan *blowfish*
10. *Progress twofish*, merupakan *progress* dari enkripsi *twofish*
11. *Progress blowfish*, merupakan *progress* dari enkripsi *blowfish*
12. *Button_Reset*, berfungsi untuk memanggil proses reset semua yang telah di inputkan
13. *Button_About*, berfungsi untuk memanggil form about yang berisi mengenai petunjuk program
14. *Button_Exit*, berfungsi untuk keluar dari program
15. Ukuran *file*, merupakan ukuran awal dari *file* yang akan dienkripsi / dekripsi

16. Waktu proses dekripsi *twofish*, untuk menampilkan waktu proses enkripsi *twofish*
17. Waktu proses enkripsi *blowfish*, menampilkan waktu proses enkripsi *blowfish*
18. *Icon* proses, untuk mengetahui *icon file* yang akan diproses
19. IP *Address*, untuk menampilkan IP *Address* dari computer