BAB II

LANDASAN TEORI

2.1 Keamanan Data dan Informasi

Dalam era digital, komunikasi melalui jaringan komputer memegang peranan penting. Melalui komunikasi elektronik, seseorang dapat melakukan transaksi atau komunikasi dengan sangat cepat dan praktis. Hal ini merupakan pengaruh dari perkembangan yang sangat signifikan dalam teknologi informasi, dimana *bandwidth* internet yang semakin besar dengan biaya akses yang semakin murah. Konsekuensinya adalah resiko dalam keamanan informasi semakin meningkat.

Keamanan data adalah perlindungan data di dalam suatu sistem melawan terhadap otorisasi tidak sah, modifikasi, atau perusakan dan perlindungan sistem komputer terhadap penggunaan tidak sah atau modifikasi.

Ada empat aspek utama dalam keamanan data dan informasi yaitu:

1. *Privacy/Confidentiality* yaitu usaha menjaga data informasi yang bersifat pribadi dari orang yang tidak berhak mengakses.
2. *Integrity* yaitu usaha untuk menjaga data atau informasi tidak diubah oleh yang tidak berhak.
3. *Authentication* yaitu usaha atau metode untuk mengetahui keaslian dari informasi, misalnya apakah informasi yang dikirim dibuka oleh orang yang benar atau layanan dari server yang diberikan benar berasal dari server yang dimaksud.

4. *Availability* berhubungan dengan ketersediaan sistem dan data (informasi) ketika dibutuhkan.

Keamanan data dapat dibedakan menjadi dua kategori, yaitu keamanan fisik dan keamanan sistem. Keamanan fisik merupakan bentuk keamanan berupa fisik dari *server*, *terminal/client router* sampai dengan *cabling.* Sedangkan keamanan sistem adalah keamanan pada sistem pengoperasiannya atau lebih khususnya pada lingkup perangkat lunak, misalnya dengan penggunaan kriptografi dan steganografi. Dalam penelitian ini akan dibahas tentang penggunaan kombinasi steganografi dan kriptografi dalam memberikan keamanan pada data.

2.2 Steganografi

Pada Sub bab ini akan menjelaskan mengenai sejarah, pengertian, penggunaan serta media yang dipakai pada steganografi.

2.2.1 Sejarah steganografi

Steganografi berasal dari bahasa Yunani yaitu *Steganós* yang berarti menyembunyikan dan *Graptos* yang artinya tulisan sehingga secara keseluruhan artinya adalah tulisan yang disembunyikan.

Seperti kriptografi, penggunaan steganografi sebetulnya telah digunakan berabad-abad yang lalu bahkan sebelum istilah steganografi itu sendiri muncul. Berikut adalah contoh penggunaan steganografi di masa lalu:

1. Pada tahun 480 sebelum masehi, seseorang berkebangsaan Yunani yaitu Demaratus mengirimkan pesan kepada polis Sparta yang berisi peringatan mengenai penyerangan Xerxes yang ditunda. Teknik yang digunakan adalah

dengan menggunakan meja yang telah diukir kemudian diberi lapisan lilin untuk menutupi pesan tersebut, dengan begitu pesan dalam meja dapat disampaikan tanpa menimbulkan kecurigaan oleh para penjaga.

2. Pada sejarah Yunani kuno, masyarakatnya biasa menggunakan seorang pembawa pesan sebagai perantara pengiriman pesan. Pengirim pesan tersebut akan dicukur rambutnya, untuk kemudian dituliskan suatu pesan pada kepalanya yang sudah botak. Setelah pesan dituliskan, pembawa pesan harus menunggu hingga rambutnya tumbuh kembali sebelum dapat mengirimkan pesan kepada pihak penerima. Pihak penerima kemudian akan mencukur rambut pembawa pesan tersebut untuk melihat pesan yang tersembunyi.
3. Selama terjadinya Perang Dunia ke-2, tinta yang tidak tampak (*invisible ink*) telah digunakan untuk menulis informasi pada lembaran kertas sehingga saat kertas tersebut jatuh di tangan pihak lain hanya akan tampak seperti lembaran kertas kosong biasa. Cairan seperti urine, susu, vinegar, dan jus buah digunakan sebagai media penulisan sebab bila salah satu elemen tersebut dipanaskan, tulisan akan menggelap dan tampak melalui mata manusia.
4. Pada perang dunia II, Jerman menggunakan microdots untuk berkomunikasi. Penggnaan teknik ini biasa digunakan pada microfilm chip yang harus diperbesar sekitar 200 kali.
5. Pada perang dunia II, Amerika Serikat menggunakan suku Indian Navajo sebagai media untuk berkomunikasi.

2.2.2 Pengertian Steganografi

Steganografi (*steganography*) adalah ilmu dan seni menyembunyikan pesan rahasia (*hiding message*) sedemikian sehingga keberadaan (eksistensi) pesan tidak terdeteksi oleh indera manusia. Steganografi membutuhkan dua jenis berkas, yaitu berkas *cover* sebagai penampung pesan dan data rahasia yang akan disembunyikan.

Steganografi digital menggunakan media digital sebagai wadah *cover*, misalnya gambar, suara, teks, dan video dengan data rahasia yang disembunyikan juga dapat berupa gambar, suara, teks, dan video. Steganografi berbeda dengan kriptografi, di mana pihak ketiga dapat mendeteksi adanya data (*chipertext*), karena hasil dari kriptografi berupa data yang berbeda dari bentuk aslinya dan biasanya data seolah-olah berantakan, tetapi dapat dikembalikan ke bentuk semula.



Gambar 2.1 Ilustrasi kriptografi dan steganografi pada citra digital.

Steganografi membahas bagaimana sebuah pesan dapat disisipkan ke dalam sebuah berkas media sehingga pihak ketiga tidak menyadarinya. Steganografi memanfaatkan keterbatasan sistem indera manusia seperti mata dan telinga. Dengan

adanya keterbatasan inilah, metode steganografi ini dapat diterapkan pada berbagai media digital. Hasil keluaran dari steganografi ini memiliki bentuk persepsi yang mirip atau bahkan sama dengan bentuk aslinya, tentunya persepsi di sini hanya sebatas kemampuan indera manusia, tetapi tidak oleh komputer atau perangkat pengolah digital lainnya.

Penyembunyian data rahasia ke dalam media *cover* akan mengubah kualitas media tersebut. Kriteria yang harus diperhatikan dalam penyembunyian data diantaranya adalah:

1. *Fidelity*

   Kualitas berkas *cover* tidak jauh berubah setelah penyisipan data rahasia. berkas *cover* hasil steganografi masih terlihat/terdengar dengan baik. Pengamat tidak mengetahui kalau di dalam berkas tersebut terdapat data rahasia.

2. *Recovery*

   Data yang disembunyikan harus dapat diungkapkan/diekstrak kembali. Karena tujuan steganografi adalah penyembunyian informasi, maka sewaktu-waktu informasi di dalam berkas *cover* harus dapat diambil kembali untuk digunakan lebih lanjut.

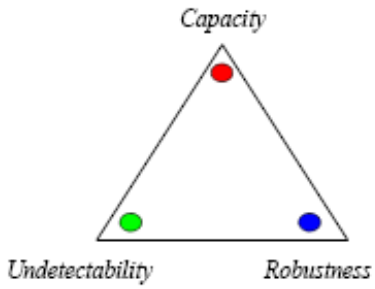
3. *Robustness*

   Robustness merupakan salah satu isu desain algoritma steganografi yang utama. Data rahasia yang disisipkan harus tahan terhadap pengolahan sinyal yang mungkin dilakukan termasuk konversi digital-analog dan analog-digital, *linear* dan *non-linear filtering,* kompresi dan perubahan ukuran (*scaling).*

4. *Security*

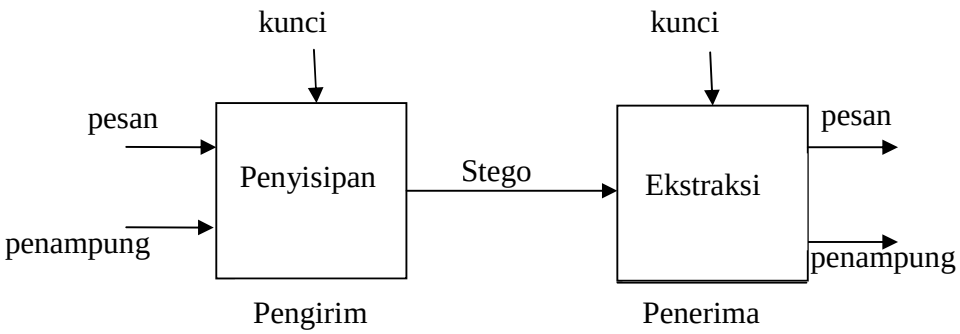
Data rahasia harus kebal terhadap deteksi pembajakan dan juga diharapkan bisa menyulitkan dari usaha steganalisis.

Dalam menyisipkan informasi ada beberapa faktor yang saling berkompetisi satu sama lain, artinya faktor-faktor tersebut tidak dapat dioptimalkan dalam satu waktu, yaitu kapasitas (*capacity*), anti deteksi (*undetectability)* dan kekokohan (*robustness*).



Gambar 2.2 Faktor-faktor yang Saling Berkompetisi dalam Steganografi

Dalam steganografi, proses penyembunyian pesan ke dalam media *cover* disebut penyisipan, sedangkan proses sebaliknya disebut ekstraksi.



Gambar 2.3 Cara kerja steganografi secara umum

Keterangan

Penyisipan : Penggabungan berkas *cover* dengan pesan rahasia

Ekstraksi : Pengekstrakan pesan rahasia dari berkas *cover*

Cover : Berkas yang menjadi media penyisipan pesan yang akan disembunyikan

Pesan : Pesan (data rahasia) yang akan disisipkan

Stego : Berkas data yang sudah disisipi pesan tersembunyi

Kunci : Parameter pengontrol proses penyembunyian

2.2.3 Penggunaan Steganografi

Steganografi sebagai suatu teknik penyembunyian informasi pada data digital lainnya dapat dimanfaatkan untuk berbagai tujuan seperti:

1. *Tamper-proofing*

   Steganografi digunakan sebagai alat untuk mengidentifikasikan atau alat indikator yang menunjukkan data host telah mengalami perubahan dari aslinya.

2. *Feature location*

   Steganografi digunakan sebagai alat untuk mengidentifikasikan isi dari data digital pada lokasi-lokasi tertentu, seperti contohnya penamaan objek tertentu dari beberapa objek yang lain pada suatu citra digital.

3. *Annotation/caption*

   Steganografi hanya digunakan sebagai keterangan tentang data digital itu sendiri.

4. *Copyright-Labeling*

   Steganografi dapat digunakan sebagai metoda untuk penyembunyian label hak cipta pada data digital sebagai bukti otentik kepemilikan karya digital tersebut.

2.2.4 Media Steganografi

Steganografi menggunakan sebuah berkas pembawa data rahasia yang disebut dengan *cover*, tujuannya sebagai kamuflase dari pesan yang sebenarnya. Banyak format berkas digital yang dapat dijadikan media untuk menyembunyikan pesan. Pada jaman modern seperti saat ini, steganografi biasanya dilakukan dengan melibatkan berkas-berkas seperti teks, gambar, audio dan video.

2.2.4.1 Steganografi pada Teks

Teknik steganografi yang menggunakan teks sebagai cover adalah hal yang menantang. Ini dikarenakan berkas teks memiliki ukuran data yang kecil untuk bisa digantikan dengan berkas rahasia. Dan kekurangan lainnya adalah teks yang

mengandung teknik steganografi ini dengan mudah dapat diubah oleh pihak yang tidak diinginkan dengan cara mengubah teks itu sendiri maupun mengubah format dari teksnya (misal .TXT menjadi .PDF).

Ada beberapa metode yang digunakan pada media teks ini yaitu, Line-Shift Encoding, Word-shift Encoding dan Feature Coding. Ketiganya merupakan metode encoding yang membutuhkan berkas asli dan juga format aslinya untuk dapat *didecode* atau diekstrak kembali.

2.2.4.2 Steganografi pada Gambar

Steganografi pada gambar adalah metode yang paling banyak digunakan secara luas di dunia digital saat ini. Hal ini dikarenakan keterbatasan kemampuan dari visual atau *Human Visual System* (HVS). Format gambar yang biasanya digunakan adalah format bitmap (bmp), gif, pcx, jpeg dan format gambar lainnya.

Hampir semua plain teks, cipher teks, gambar dan media lainnya dapat *diencode* ke dalam aliran bit untuk disembunyikan di dalam gambar digital. Perkembangan dari metode ini sangat pesat, didukung dengan semakin canggihnya komputer grafik yang *powerfull*, dan perangkat lunak steganografi yang sekarang sudah banyak tersebar luas di internet. Pendekatan yang paling sering dilakukan pada media jenis ini adalah *Least Significant Bit Insertion*, *Masking and Filtering* dan *Algorithm and Transformation*. Sebagai catatan, masih ada banyak lagi teknik-teknik yang digunakan untuk *encoding* pada media gambar ini.

2.2.4.3 Steganografi pada Audio

Penyembunyian data pada audio merupakan teknik yang paling menantang pada steganografi ini. Hal ini disebabkan Human Auditory System (HAS) memiliki jangkauan yang dinamis. HAS memiliki kemampuan mendengar lebih dari satu sampai 1 miliar. Dan jangkauan frekuensi lebih dari satu hingga seribu. Auditory System ini juga sangat peka pada gangguan suara (noise) yang halus sekalipun. Sedikit saja terdapat gangguan pada sebuah berkas audio maka dengan mudah akan terdeteksi. Satu-satunya kelemahan yang dimiliki HAS dalam membedakan suara adalah kenyataan bahwa suara keras bisa menenggelamkan suara pelan. Terdapat dua konsep yang harus dipertimbangkan sebelum memilih metoda mana yang akan dipakai. Yaitu format digital audio dan media transmisi dari audio.

Terdapat 4 (empat) teknik yang sering digunakan dalam menyembunyikan pesan dalam format audio ini yaitu :

a. Low Bit Encoding

Metode steganografi yang paling umum pada tipe berkas audio dan gambar adalah *least significant bit* atau disebut juga *Low Bit Encoding*. Metode ini berasal dari angka yang paling kurang signifikan dari jumlah bit dalam 1 byte. Bit yang memiliki signifikansi paling tinggi adalah numerik yang memiliki nilai tertinggi (misal $2^7 = 28$) , artinya yang paling tidak signifikannya adalah yang memiliki nilai terendah (misal $2^0 = 1$).

b. Phase Coding

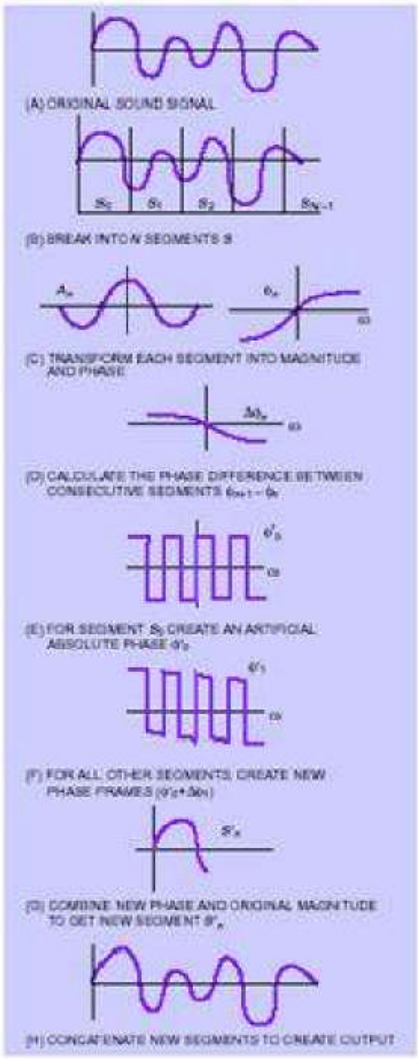
Phase coding bekerja berdasarkan karakteristik sistem pendengaran manusia yang mengabaikan suara yang lebih lemah jika dua suara itu datang bersamaan. Ide dasar dibalik teknik *phase coding* adalah menyembunyikan data dengan cara menukarkan fase asli segmen inisial dari sinyal suara dengan fase absolut dari sinyal dari berkas yang telah tersisipi dengan tetap menjaga fase relatif antara segmen sinyal menggunakan beda fase segmen dari sinyal asli. Ketika beda fase antara sinyal asli dan sinyal yang dimodifikasi besarnya kecil, maka perbedaan suara yang dihasilkan tidak terdeteksi oleh pendengaran manusia.

Metode *Phase coding* adalah metode yang mensubstitusi bagian inisialisasi dari sebuah berkas audio dengan sebuah phase lain yang berisikan data yang akan disembunyikan. Metode ini memanfaatkan kelemahan sistem pendengaran manusia untuk merasakan fase absolut yang mengabaikan suara yang lebih lemah jika dua suara itu datang bersamaan.

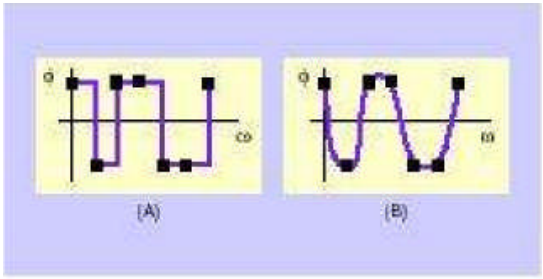
Terdapat beberapa parameter yang berpengaruh dalam teknik steganografi *phase coding.* Parameter tersebut antara lain fase relatif dan amplitudo sinyal suara, fase absolut data, serta beda fase relatif antara segmen sinyal suara.

Phase coding didasarkan pada kenyataan bahwa pendengaran manusia lebih peka terhadap beda fase relatif dalam sinyal audio dibandingkan dengan fase absolut yang ditambahkan. Jika hubungan fase antar setiap komponen frekuensi diubah secara dramatis, akan terjadi dispersi fase yang tampak dengan jelas. Akan tetapi, selama modifikasi fase cukup kecil (tergantung pada pengamat) dan

dengan menjaga beda fase relatif antara segmen-segmen sinyal suara, maka modifikasi fase yang dilakukan tidak akan terdengar.Amplitudo sinyal suara digunakan bersama dengan nilai fase sinyal suara untuk mengubah kembali sinyal suara dari domain frekuensi menjadi domain waktu sehingga dapat didengarkan.



Gambar 2.4 Steganografi Phase Coding



Gambar 2.5 Sinyal steganografi phase coding

c. Spread Spectrum

Encoding dari spread spectrum pada audio ini melibatkan keseluruhan dari spektrum frekuensinya. Kemudian mentransmit audio tersebut melalui frekuensi yang berbeda-beda tergantung dari metode spread spectrum yang digunakan.

Direct Sequence Spread Spectrum (DSSS) adalah salah satu metode spread spectrum yang digunakan dalam menyebarkan sinyal dengan melipatgandakan sumber sinyal dengan beberapa bagian pseudorandom yang dikenal dengan nama chip. Sampling rate dari sumber sinyal ini kemudian akan digunakan sebagai rate dari chip dalam proses coding nya. Teknik spread spectrum ini adalah teknik paling aman untuk mengirimkan pesan didalam audio, tetapi teknik ini juga menimbulkan noise secara acak pada audio, dan mengakibatkan beberapa data asli dari audio tersebut hilang.

d. Echo Data Hiding

Teknik echo data hiding dilakukan dengan cara menambahkan *echo* kedalam audio *cover* sebagai representasi dari data yang dimasukkan. Data disembunyikan dengan tiga parameter echo yang berbeda-beda: Initial Amplitude, Decay rate dan

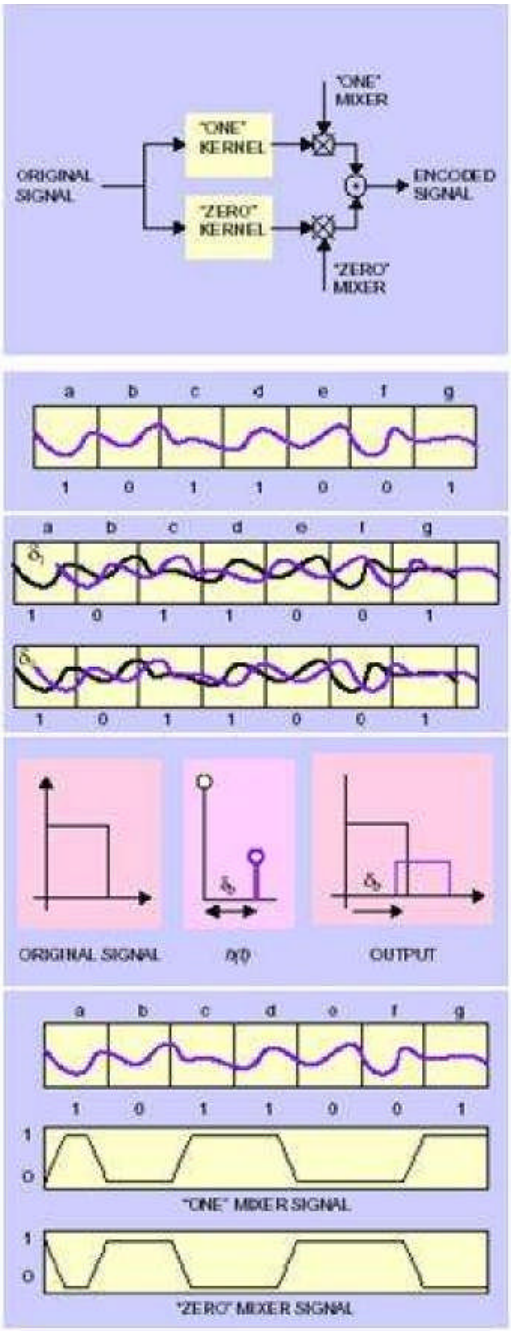
offset, atau delay. Ketika offset antara data audio asli dengan echo nya berkurang, maka dua sinyal akan berbaur. Pada saat-saat tertentu. Pendengaran manusia tidak dapat membedakan antara dua sinyal dan echo biasanya hanya dianggap resonansi saja. Penggunaan metode ini bergantung pada beberapa faktor seperti kualitas dari rekaman aslinya, tipe audio, dan tentu saja pendengarnya.

Echo data hiding merupakan salah satu bentuk dari metode penyisipan data pada suatu sinyal audio. Hal ini dilakukan dengan harapan agar berkas audio yang telah disisipi hanya akan mengalami penurunan kualitas suara seminimal mungkin. Metode ini telah banyak digunakan unluk hal kepemilikan, keterangan dan jaminan terhadap integritas data. Oleh karena itu. data (pesan yang disisipkan) sebaiknya tidak dapat dengan mudah dihilangkan oleh transformasi umum pada stego audio (sinyal audio yang telah disisipi), seperti filtering, resampling, block editing atau kompresi data yang bersifat lossy,

Penyisipan pesan pada sinyal audio menimbulkan berbagai keraguan berkaitan dengan perbedaan jangkauan dan kemampuan dari sistem pendengaran manusia yang lebih bersifat dinamis dibandingkan dengan indera yang lain. Sistem pendengaran manusia dapat merasakan suara dengan jangkauan amplitudo satu milyar banding satu dan jangkauan frekuensi seribu banding satu. Selain itu kepekaan sistem pendengaran manusia terhadap tambahan noise juga tajam. Gangguan file suara dapat dideteksi serendah satu bagian dari sepuluh juta (80 db dibawah level ambient). Jangkauan yang lebar dan dinamis dari sistem pendengaran manusia inilah yang menyebabkan adanya beberapa "lubang" dimana data dapat disembunyikan. Hal

ini juga sering menyebabkan terjadinya lebar jangkauan yang sedikit berbeda. Oleh sebab itu, sinyal suara dengan amplitudo kuat cenderung menutupi sinyal suara dengan amplitudo lemah. Sistem pendengaran manusia hanya dapat merasakan fase relatif sehingga tidak dapat merasakan fase absolut. Hal ini menyebabkan adanya penyimpangan-penyimpangan yang umumnya tidak terdeteksi oleh pendengar dalam beberapa kasus.

Seperti semua metode - metode steganografi yang baik, echo data hiding juga menyisipkan informasi pada data *cover* dengan penurunan kualitas seminimal mungkin. Tujuan dari penurunan yang minimal adalah agar pendengar melewatkan atau tidak dapat merasakan perubahan yang telah terjadi pada sinyal audio tersebut. Perubahan yang digunakan pada metode ini mirip pada gema yang ditemukan pada ruangan karena adanya tembok.



Gambar 2.6 Steganografi Echo data hiding

2.3 Kriptografi

Kriptografi berasal dari bahasa Yunani, *cryptos* yang berarti rahasia dan *graphein* yang berarti tulisan. Kriptografi adalah sebuah cara dalam mengamankan dan mengirim data dalam bentuk yang hanya diketahui oleh pihak yang berhak membukanya. Kriptografi merupakan ilmu dan seni dalam memproteksi informasi dengan mengubahnya ke dalam bentuk himpunan karakter acak yang tidak dapat dibaca. Kriptografi adalah sebuah cara yang efektif dalam mengamankan informasi-informasi penting baik yang tersimpan dalam media penyimpanan maupun yang ditransmisikan melalui jaringan komunikasi.

Walaupun tujuan utama dari kriptografi dan segala mekanisme yang dirancang adalah untuk mengamankan informasi dari pihak yang tidak berhak, sebagian besar algoritma dapat dipecahkan dan informasi dapat didapatkan jika penyerang mempunyai cukup waktu, kesungguhan dan sumber daya. Sehingga arahan realistis dari kriptografi dewasa ini adalah untuk merancang mekanisme algoritma yang kuat dan tahan terhadap serangan para penyerang.

2.3.1 Sejarah Kriptografi

Kriptografi telah digunakan sejak 4000 tahun lalu. Di Romawi, dikisahkan suatu saat Julius Caesar mengirimkan pesan rahasia kepada jenderalnya yang sedang berada di medan perang. Ia mengacak pesan rahasia tersebut hingga menjadi pesan yang tidak dapat dipahami siapapun selain jenderalnya. Yang dilakukan Julius Caesar adalah menggeser 3 urutan alfabet ke kanan, sehingga a menjadi d, b menjadi e dan seterusnya. Dari ilustrasi tersebut, beberapa istilah kriptografi dipergunakan. Proses

mengacak pesan disebut enkripsi. Pada saat jenderal menyusun kembali pesan acak tersebut menjadi pesan seperti sediakala disebut dekripsi. Pesan awal yang belum diacak disebut *plaintext,* dan pesan yang telah diacak disebut *ciphertext.* Orang Mesir menggunakan kriptografi dengan huruf-huruf *hieroglyph.*



Gambar 2.7 Penggunaan *Hierogliph* di Mesir

Metode kriptografi pun berkembang sesuai dengan perkembangan zaman dan kebutuhan. Pada abad ke-15 Leo Battista Alberti menemukan metode roda kode (*wheel cipher)* yang dipergunakan sebagai alat enkripsi. Pada perang dunia ke-2, militer Jerman menggunakan mesin *cipher* substitusi polialfabetik yang disebut Enigma sebagai sistem pengkodean utama.

Enigma menggunakan rotor mekanis untuk pengkodean dan pendekodean. Seorang Belanda, Hugo Koch mengembangkan mesin ini pada 1919, dan diproduksi untuk pasar komersial pada 1923 oleh Arthur Scherbius. Scherbius mendapatkan hak paten pada mesin Enigma untuk perusahaan Berlin Chiffriermasschinen Aktiengesellschaft. Pakar kriptanalisis Polandia, Marian Rejewski, bekerja bersama Perancis dari 1928 sampai 1938, berhasil memecahkan pengkabelan sistem 3 rotor

yang digunakan Jerman saat itu dan menciptakan berkas kartu yang dapat mengantisipasi 6 kali 17.576 kemungkinan posisi rotor.

Jerman mengubah indikator sistem dan jumlah rotor menjadi 6 pada 1938, sehingga meningkatkan kesulitan untuk memecahkan *cipher* Enigma. Dalam kerjanya pada 1938, Polandia dan Perancis mengkonstruksi mesin prototipe yang disebut “The Bombe” untuk memecahkan *cipher* Enigma. Namanya diturunkan dari bunyi detikan yang dihasilkan oleh mesin. Usaha memecahkan *cipher* Enigma diambil alih oleh Inggris di Bletchley Park Inggris dan dipimpin oleh banyak ilmuwan terkemuka termasuk Alan Turing. Prototipe Bombe Turing muncul pada 1940, dan Bombe berkecepatan tinggi dikembangkan oleh Inggris dan Amerika pada 1943.

Perkembangan komputer dan sistem komunikasi pada tahun 60-an berdampak pada permintaan dari sektor-sektor swasta sebagai sarana untuk melindungi informasi dalam bentuk digital dan untuk menyediakan layanan keamanan. Dimulai dari usaha Feistel pada IBM di awal tahun 70-an dan mencapai puncaknya pada 1977 dengan pengangkatan DES (*Data Encryption Standard*) sebagai standar pemrosesan informasi federal US untuk mengenkripsi informasi. DES merupakan mekanisme kriptografi yang paling dikenal sepanjang sejarah.

Di Amerika Serikat, sejak tanggal 4 November 1952 telah didirikan *National Security Agency* (NSA) yang diresmikan oleh Presiden Harry Truman. Keputusan pendirian NSA diambil setelah memperhatikan kesuksesan dalam perang dunia kedua di Pasifik yang merupakan kontribusi dari keberhasilan dalam memecahkan kode rahasia Jerman dan Jepang. NSA (www.nsa.gov) merupakan organisasi kriptografi

nasional Amerika Serikat, dimana para pembuat kode dan pemecah kode berhimpun. Misi utama NSA adalah untuk melindungi sistem keamanan nasional Amerika Serikat dan untuk mengetahui informasi intelijen dari pihak asing.



Gambar 2.8 Logo NSA (*National Security Agency*) dan CSS (*Central Security Service*)

Pengembangan paling mengejutkan dalam sejarah kriptografi terjadi pada 1976 saat Diffie dan Hellman mempublikasikan *New Directions in Cryptography*. Tulisan ini memperkenalkan konsep revolusioner kriptografi kunci publik dan juga memberikan metode baru dan jenius untuk pertukaran kunci, keamanan yang berdasar pada kekuatan masalah logaritma diskrit. Pada 1978 Rivest, Shamir dan Adleman menemukan rancangan enkripsi kunci publik dan tanda tangan, yang sekarang disebut RSA. Rancangan RSA berdasar pada masalah matematika yang sulit untuk kriptografi menggiatkan kembali usaha untuk menemukan metode yang lebih efisien untuk pemfaktoran. Kelas lain yang merupakan rancangan kunci publik praktis ditemukan oleh ElGamal pada 1985. Rancangan ini juga berdasar pada masalah logaritma diskret.

Selama bertahun-tahun kriptografi menjadi bidang khusus yang hanya dipelajari oleh kalangan militer, seperti agen keamanan nasional Amerika (*National Security Agency)*, Uni Soviet, Inggris, Perancis, Israel dan negara-negara lain yang rela membelanjakan miliaran dolar untuk mengamankan komunikasi mereka dari pihak luar. Mereka juga selalu mempelajari kode-kode rahasia dari negara lain. Namun dalam 30 tahun terakhir kriptografi tidak hanya dimonopoli kalangan militer saja. Kriptografi meluas di kalangan sipil, yaitu mereka yang menginginkan pesan dan komunikasi mereka aman dari pihak lain. Bahkan kriptografi kini merambah ke dunia sastra dengan dijadikan tema dalam novel maupun film.

2.3.2 Tujuan Kriptografi

Tujuan kriptografi adalah melindungi data dari ancaman yang disengaja atau tidak disengaja dengan mengubah suatu data informasi menjadi sebuah sandi yang hanya akan dimengerti oleh pihak pengirim dan penerima pesan.

2.3.3 Komponen Kriptografi

Pada dasarnya kriptografi terdiri dari beberapa komponen sebagai berikut:

1. Algoritma, merupakan himpunan aturan matematis yang digunakan dalam enkripsi dan dekripsi.
2. Enkripsi, adalah transformasi data ke dalam bentuk yang tidak dapat terbaca tanpa sebuah kunci tertentu.
3. Dekripsi, merupakan kebalikan dari enkripsi, yaitu transformasi data terenkripsi kembali ke bentuknya semula.

4. Kunci, digunakan pada saat melakukan enkripsi dan dekripsi. Pada kriptografi modern, keamanan enkripsi tergantung pada kunci, dan tidak tergantung kepada algoritmanya apakah dilihat orang lain atau tidak.
5. Pesan asli *(Plaintext),* disebut juga dengan *clear-text,* merupakan teks asli yang akan diproses menggunakan algoritma kriptografi tertentu untuk menjadi *ciphertext.*
6. *Ciphertext,* merupakan pesan yang telah melalui proses enkripsi yang merupakan himpunan karakter acak.
7. Kriptologi, merupakan studi tentang kriptografi dan kriptanalisis.
8. Kriptanalis (*Cryptanalysis),* merupakan aksi memecahkan mekanisme kriptografi dengan cara menganalisisnya untuk menemukan kelemahan dari suatu algoritma kriptografi sehingga akhirnya dapat ditemukan kunci atau teks asli.

2.3.4 Teknik Kriptografi.

Pada umumnya terdapat dua teknik yang digunakan dalam kriptografi yaitu kunci simetrik dan kunci asimetrik.

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.

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, yakni 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 dibuat adalah harus sebanyak jumlah pihak yang berkorespondensi.

2.3.5 Algoritma Rijndael

Sejak tahun 1976, *Data Encryption Standard* (DES) dipilih sebagai standar kriptografi yang dipakai pada pemerintahan Amerika Serikat. Namun pada tahun 1990, panjang kunci DES dianggap terlalu pendek, dan pada tahun 1998 DES berhasil dipecahkan dalam waktu 96 hari, kemudian di tahun 1999 dapat dipecahkan dalam waktu 22 hari.

Karena alasan tersebut maka kemudian diadakan kompetisi oleh NIST (*National Institute of Standard and Technology)* untuk mencari pengganti DES. NIST mengundang peserta dari seluruh dunia untuk berpartisipasi dengan mengajukan algoritma baru untuk menggantikan DES. Pada tahun 1997 ada 21 pelamar dan 6 dari mereka gugur karena tidak masuk dalam kriteria pemilihan. Kemudian dipilih 5 kandidat untuk seleksi akhir, yaitu Mars (IBM Amerika), RSA (RSA Corp. Amerika), Rijndael (Belgia), Serpent (Israel, Norwegia dan Inggris) dan Twofish (Counterpane Amerika). Pada tahap ini NIST memberikan kriteria sebagai berikut.

1. Aspek keamanan, NIST memberikan kesempatan dalam 3 tahun kepada komunitas kriptografer untuk menganalisis dan mengevaluasi kandidat-kandidat tersebut, sehingga dapat diketahui mana algoritma yang kuat dan yang lemah.
2. Implementasi software, mempunyai kategori kecepatan eksekusi, kemampuan antar-platform dan variasi dari kecepatan dengan ukuran kunci.
3. Ruang lingkup yang tidak terbatas, algoritma dapat digunakan pada banyak aplikasi seperti *smart card, Random Acces Memory* (RAM) dan sebagainya.
4. Implementasi perangkat keras, algoritma mampu mengoptimalkan kecepatan atau ukuran, tidak membutuhkan memori yang besar untuk menjalankannya, sehingga sumber daya tidak banyak terpakai.
5. Algoritma untuk enkripsi dan dekripsi berbeda.
6. Ketahanan terhadap serangan, tidak ada jenis serangan yang dapat memecahkan algoritma yang ada. Dari segi kunci tidak mudah dipecahkan.

7. Kemampuan kunci, mengacu kepada kemampuan kecepatan pertukaran kunci dan memiliki sumber yang sedikit.
8. Potensial untuk instruksi paralel, mengacu kepada kemampuan prosesor masa depan.

Melalui seleksi yang sangat ketat dan persyaratan yang tidak mudah maka pada 2 Oktober 2000 terpilih algoritma Rijndael yang dibuat oleh Joan Daemen dan Vincent Rijmen sebagai pemenang.

2.3.6 Unit Data pada Algoritma Rijndael

Rijndael menggunakan 5 unit pengukuran data yaitu bit, byte, word, blok dan *state.* Bit adalah sebuah *binary digit* dengan nilai 0 atau 1 yang merupakan unit pengukuran terkecil. Sebuah byte terdiri dari 8 bit. 1 word terdiri dari 4 byte (32 bit). Rijndael mengenkripsi dan mendekripsi blok data. Untuk menyesuaikan dengan AES, maka sebuah blok pada Rijndael terdiri dari 128 bit. Rijndael menggunakan beberapa putaran dimana setiap putaran terdiri dari beberapa tahapan. Blok data ditransformasi dari satu tahapan ke tahapan selanjutnya. Sebelum dan sesudah menjalani tiap tahapan, blok data disebut sebagai *state.*

2.3.7 Transformasi pada Algoritma Rijndael

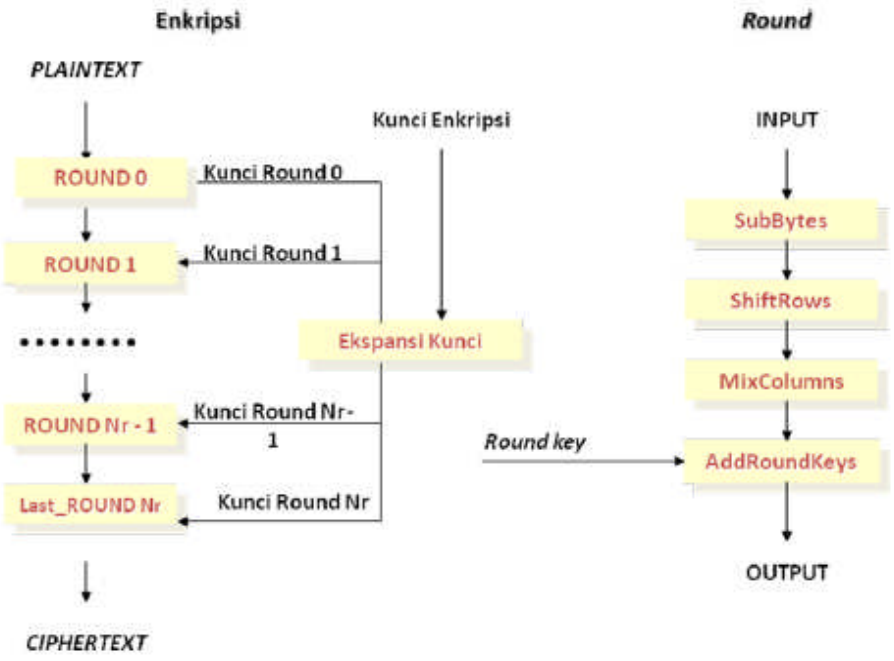
Algoritma Rijndael menggunakan substitusi, permutasi dan sejumlah putaran yang dikenakan pada tiap blok yang akan dienkripsi/dekripsi. Untuk setiap putarannya, Rijndael menggunakan kunci yang berbeda. Kunci setiap putaran disebut *round key.* Pada Rijndael dimungkinkan untuk menggunakan panjang kunci yang berbeda tergantung pada tingkat keamanan yang dibutuhkan untuk suatu aplikasi.

Rijndael dapat dikategorikan sebagai *iterated block cipher* dengan panjang kunci yang dapat dipilih secara independen sebagai 128, 192, atau 256 bit. Dalam desimal ada kira-kira 3.4 x 1038 kemungkinan kunci 128 bit, 6.2 x 1057 kemungkinan kunci 192 bit, dan 1.1 x 1077 kemungkinan kunci 256 bit. AES menspesifikasikan tiga ukuran kunci, 128, 192, dan 256 bit dengan ukuran blok tetap 128 bit. (Forouzan, 2008).

Tabel 2.1 Parameter Rijndael

	Kunci 128	Kunci 192	Kunci 256
Panjang Kunci	4 word (16 byte)	6 word (24 byte)	8 word (32 byte)
Panjang *Plaintext*	4 word (16 byte)	4 word (16 byte)	4 word (16 byte)
Jumlah Putaran	10	12	14
Ukuran *Round Key*	4 word (16 byte)	4 word (16 byte)	4 word (16 byte)
Ukuran Kunci Ekspansi	44 word (176 byte)	52 word (208 byte)	60 word (240 byte)

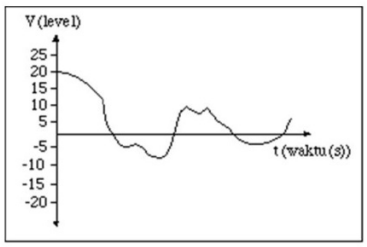
Pada intinya, tiap blok masukan (*array state*) dikenakan empat fungsi utama, yaitu SubBytes, ShiftRows, MixColumns dan AddRoundKey (Daemen *et al*, 1999). *SubBytes* adalah transformasi substitusi pada setiap byte menggunakan tabel kotak S. *ShiftRows* adalah transformasi dengan menggeser baris ke-r dalam *array state* sebanyak r byte ke kiri. *MixColumns* adalah mengacak *array state* dengan cara melakukan perkalian matriks yang merupakan transformasi dari perkalian polinom antara tiap kolom dengan polinom empat suku pada GF(28), dimana GF adalah singkatan dari Galois Field. Dan terakhir adalah *AddRoundKey* yaitu melakukan *XOR* antara *array state* sekarang dengan *round key*.



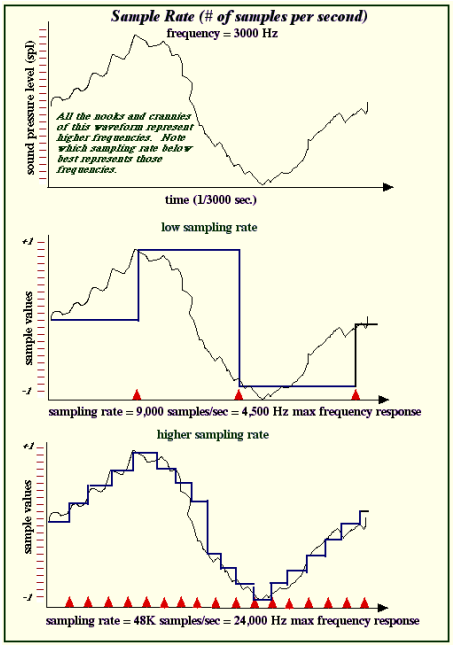
Gambar 2.9 Skema Enkripsi Rijndael

2.4 Berkas Audio

Suara (audio) atau bunyi merupakan sebuah sinyal analog yang dihasilkan oleh getaran yang membentuk sebuah gelombang longitudinal dengan frekuensi 20 hz sampai 20.000 hz (berdasarkan tingkat pendengaran manusia). Gelombang suara yang masih berbentuk sinyal analog dapat diubah ke dalam bit-bit digital menjadi sinyal digital, melalui proses sampling dan digitalisasi.



Gambar 2.10 Suara dalam bentuk sinyal analog



Gambar 2.11 Contoh Proses Sampling dari sinyal analog ke sinyal digital

2.4.1 WAV

Berkas audio ini diciptakan oleh Microsoft. Berkas ini menjadi standar berkas audio dalam *Personal Computer*(PC) baik untuk sistem operasi, *game*, maupun berkas suara lain yang kualitas suaranya setara dengan CD. Berkas ini memiliki ukuran yang cukup besar, karena suara yang disimpan dalam format ini tidak mengalami proses kompresi *(raw data).* Meski buatan Microsoft, bukan berarti berkas ini tidak dapat dijalankan oleh sistem operasi selain Windows. Setidaknya, Linux dan Macintosh juga dapat menjalankan berkas ini.

Berkas wave (WAV) adalah format berkas audio yang diciptakan oleh Microsoft, dan telah menjadi standar format audio bagi PC. Berkas WAV dikenali sebagai sebuah berkas dengan ekstensi *.wav.

Berkas WAV memiliki format 1-N saluran dan 16 bit per *sample,* serta memiliki *sampling rate* 44.1 KHz. Berkas WAV digunakan di dalam PC sebagai suatu medium pertukaran antara komputer dengan platform yang berbeda.

Sound sample juga disimpan sebagai data mentah dengan 1-N saluran dalam berkas yang sama. Saluran tersebut harus diselipkan dengan cara yang sama seperti dalam berkas *AIFF-C.* Sebagai tambahan untuk data mentah audio yang tidak terkompresi, format berkas WAV menyimpan informasi mengenai jumlah track (mono atau stereo), *sample rate,* dan *bit depth.*

Keuntungan dengan adanya berkas WAV adalah kita dapat mengedit berkas tersebut karena berkas tersebut masih dalam kondisi tidak terkompresi, tetapi ketika berkas telah dimampatkan (kompresi), kita akan mengalami kesulitan dalam melakukan penyisipan.

Berkas WAV sendiri terdiri dari tiga chunk informasi, yaitu RIFF *chunk* yang mengindentifikasikan bahwa file tersebut adalah file WAV, FORMAT *chunk*, yang mengidentifikasikan parameter-parameter seperti data rate,dan DATA *chunk* yang merupakan data yang sebenarnya. Beberapa *chunk* dapat dipecah sebagai berikut:

a. RIFF *chunk* (12 *bytes*)

Tabel 2.2 RIFF *Chunk* pada WAVE

Byte Number	
0 – 3	"RIFF" (ASCII Char)
4 – 7	Total Length of Package to Follow
8 – 11	"WAV" (ASCII Char)

b. FORMAT *chunk* (24 *bytes*)

Tabel 2.3 FORMAT *Chunk* pada WAV

Byte Number	
0 – 3	"fmt_" (ASCII Char)
4 – 7	Length of FORMAT chunk
8 – 9	Always 0x01 / PCM
10 – 11	channel number
12 -15	Sample Rate
16 – 19	Bytes per second
20 -21	Bytes per sample
22 -23	Bits per sample

c. DATA *chunk*

Tabel 2.4 DATA *Chunk* pada WAV

Byte Number	
0 – 3	"data" (ASCII Char)
4 – 7	Length of Data to Follow
8 – end	Data (samples)

2.4.2 MPEG Audio Layer 3 (MP3)

Asal-usul MP3 dimulai dari penelitian IIS-FHG (Institut Integriette Schaltungen Fraunhofer Gesellschaft), sebuah lembaga penelitian terapan di Munich, Jerman dalam penelitian *coding audio perceptual.* Penelitian mengenai pemampatan berkas audio ini dipimpin langsung oleh Karl Heinz Brandenburg, dan menghasilkan

sebuah algoritma MPEG-1 Layer 3 yang kemudian dikenal sebagai MP3. Penelitian tersebut menghasilkan suatu algoritma yang menjadi standar sebagai ISO-MPEG Audio Layer-3 (MP3), yang merupakan berkas dengan teknik *lossy compression.*

Dalam dunia kompresi digital dikenal dengan dua macam metode yakni *lossless compression* dan *lossy compression*. Pada *lossless compression* terjadi penahanan (tidak menghilangkan) semua informasi yang ada dan dekodernya mampu merekonstruksi sinyal yang telah terkompresi berubah menjadi bentuk berkas asli. Kompresi *lossless* menghasilkan kompresi data yang hampir mirip dengan kualitas audio aslinya, hasil kompresi ini juga menghasilkan ukuran file yang lebih kecil (yakni sekitar 50-75% saja) dan tentunya masih bisa dikembalikan ke bentuk aslinya tanpa menghilangkan sesuatu apapun informasi yang terkandung didalamnya. Format-format *lossless compression* bisa dalam bentuk FLAC (*Free Lossless Audio Codec*) dan Monkey Audio (*APE*).

Pada metode dengan *lossy compression*, dimana menghasilkan kompresi dengan ukuran yang jauh lebih kecil dibandingkan berkas asli. Kompresi jenis *lossy* tentunya lebih terkenal dalam dunia portable audio karena ukuran filenya jauh lebih kecil dibanding dengan penggunaan jenis *lossless compression.* Hal itu terjadi karena, pemampatan data dilakukan dengan cara mengurangi atau menghilangkan informasi-informasi yang dianggap tidak penting atau yang kurang *audible* bagi telinga. Hasil kompresi dengan teknik *lossy* tidak dapat dibalikkan ke dalam bentuk semula. Sebagai contoh berkas audio yang menggunakan teknik *lossy* adalah MP3. Sebuah lagu (WAV) dengan durasi 3 menit dapat menyita alokasi *hard-disk* sebesar 30 MB.

Lagu yang sama dengan format MP3 hanya membutuhkan ruang sebesar 3 MB dengan penurunan kualitas suara yang minimum.

Dalam upaya menghasilkan MP3, Brandenburg menganalisis bagaimana otak dan telinga manusia menangkap suara. Teknik yang digunakan berhasil memanipulasi telinga dengan membuang bagian yang kurang penting pada suatu file musik. Sebagai contoh, apabila terdapat dua nada yang mirip, atau apabila nada tinggi dan rendah muncul secara bersamaan, otak hanya akan memproses salah satunya. Sehingga algoritma MP3 akan memilih sinyal yang lebih penting dan membuang sisanya. Hasilnya adalah file MP3 yang memiliki ukuran file audio orisinal hingga 10 kali lebih kecil. Berkas ini sering digunakan di internet karena ukurannya yang cukup kecil dibandingkan ukuran berkas audio yang tidak terkompresi. Teknologi ini kemudian distandarisasi pada tahun 1991.

Dalam hal audio steganografi, ada beberapa keuntungan dalam menggunakan MP3. Pertama, keberadaannya yang bersifat kosmopolit dan populer sebagai sarana hiburan di kalangan pengguna komputer. Karena keberadaannya yang sangat umum, diharapkan dapat meminimalisir kecurigaan akan adanya pesan rahasia di dalamnya. Selain itu, MP3 dapat didistribusikan dengan mudah dan hampir tanpa biaya walaupun sebenarnya hak paten dari MP3 telah dimiliki dan penyebaran MP3 seharusnya dikenakan biaya. Namun pemilik hak paten dari MP3 telah memberikan pernyataan bahwa penggunaan MP3 untuk perorangan tidak dikenai biaya. Keuntungan lainnya adalah kemudahan dalam mengakses MP3, dimana banyak *software* yang dapat menghasilkan berkas MP3. Kesuksesan MP3 dimulai pada 1998,

ketika WinAmp, sebuah mesin pemutar MP3 yang dibuat oleh sepasang mahasiswa bernama Justin Frankel dan Dmitry Boldyrev, ditawarkan secara cuma-cuma di internet. Dalam waktu singkat, pengguna musik di seluruh dunia terhubung dalam satu jaringan pusat bernama MP3, dan saling menawarkan musik-musik yang memiliki hak cipta secara gratis. Dalam rentang waktu yang tidak terlalu lama, banyak *programmer* lain yang tidak mau ketinggalan, mereka menciptakan berbagai perangkat lunak pendukung untuk para pengguna MP3. *Encoder, ripper,* dan *player* terbaru dirilis setiap minggunya, dan pertumbuhannya bergerak semakin kencang. Mesin-mesin pencari pun membuat proses pencarian file MP3 tertentu yang dikehendaki menjadi semakin cepat. Selain itu, *player portabl*e seperti Rio dan iPod membuat MP3 dapat didengarkan sambil berjalan.

2.5 Metode Pengembangan Perangkat Lunak

Pada sub bab ini menjelaskan tentang beberapa metode pengembangan perangkat lunak.

1. Metode *Waterfall*

Metode pengembangan perangkat lunak secara *Waterfall* yang meliputi: Teknik analisis data dalam pembuatan perangkat lunak menggunakan paradigma perangkat lunak secara *waterfall*, yang meliputi beberapa proses diantaranya:

a. Kebutuhan Sistem / Informasi

Merupakan bagian dari sistem yang terbesar dalam pengerjaan suatu proyek, dimulai dengan menetapkan berbagai kebutuhan dari semua elemen yang

diperlukan sistem dan mengalokasikannya ke dalam pembentukan perangkat lunak.

b. Analisis

Merupakan tahap menganalisis hal-hal yang diperlukan dalam pelaksanaan proyek pembuatan perangkat lunak, seperti analisis kebutuhan fungsional dan nonfungsional.

c. Perancangan

Tahap perancangan *interface* yang mudah dimengerti user yang mengacu pada data – data analisis.

d. Pengkodean

Tahap penerjemahan data yang telah dirancang ke dalam bahasa pemrograman tertentu.

e. Pengujian

Merupakan tahap pengujian terhadap perangkat lunak yang dibangun. Pada tahap ini menggunakan dua metode pengujian, yaitu pengujian alpha dan pengujian beta. Pengujian alpha dilakukan dengan metode black box untuk menguji fungsional sistem seperti kesalahan sintak, terminasi dan kesalahan interface apakah sudah sesuai dengan fungsi yang dibangun. Sedangkan pengujian beta dilakukan dengan menyebarkan kuesioner.

f. Pemeliharaan

Tahap akhir dimana suatu perangkat lunak yang sudah selesai dapat mengalami perubahan–perubahan atau penambahan sesuai dengan permintaan user.

2. Model *Prototype*

   Model *prototype* dibangun dari mengumpulkan berbagai kebutuhan, kemudian tim pengembang akan bertemu dengan pelanggan untuk menentukan tujuan dari perangkat lunak, dan mengidentifikasi kebutuhan-kebutuhan yang telah diketahui oleh pelanggan, dan batasan-batasan apa saja yang dapat dikategorikan sebagai tugas utama. Hasilnya akan dibangun rancangan sementara yang mewakili berbagai aspek dari perangkat lunak yang kelak akan digunakan oleh pelanggan/pengguna (seperti bentuk pendekatan input yang digunakan dan bentuk output). Idealnya model *prototype* melayani sebuah mekanisme untuk mengidentifikasi kebutuhan perangkat lunak. Dimana jika nantinya sebuah model *prototype* berhasil dibuat, seorang developer harus berusaha mendayagunakan tools yang ada (semisal, report generator, windows manager) dapat bekerja dengan baik (cepat).

3. Model RAD

   Rapid Aplication Development (RAD) adalah sebuah model proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan yang sangat pendek. Model RAD ini merupakan sebuah adaptasi “kecepatan tinggi” dari model sequensial linier dimana perkembangan cepat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen.

4. Model Evolusioner

   Model Evolusioner adalah model iterative. Model itu ditandai dengan tingkah laku yang memungkinkan perekayasa perangkat lunak mengembangkan versi perangkat yang lebih lengkap sedikit demi sedikit.

2.6 Pemodelan Data Terstruktur

Alat-alat pemodelan sistem informasi sangat dibutuhkan dalam proses analisis dan perancangan sistem. Alat-alat pemodelan sistem informasi terdiridari:

a. Bagan Alir Dokumen (*Document Flowmap*)

   Bagan alir dokumen atau disebut juga bagan alir formulir merupakan bagan alir yang menunjukan arus dari laporan dan formulir termasuk tembusan-tembusannya.

b. *Entity-Relasionship Diagram* (ERD)

   ERD adalah diagram yang memperlihatkan entitas-entitas yang terlibat dalam suatu sistem serta hubungan-hubungan (relasi antar entitas. Komponen-komponen pembentuk model ERD yaitu:

   1. Entitas (*Entity*)

      Entitas merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain. Entitas dapat berupa orang, tempat, benda, peristiwa atau konsep yang bisa memberikan atau mengandung informasi.

2. Atribut (*Attributes/Properties*)

   Setiap entitas pasti memiliki atribut yang mendeskripsikan karakteristik dari entitas tersebut.

3. Relasi (*Relationship*)

   Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

c. Diagram Konteks (*Context Diagram*)

Diagram konteks merupakan diagram alir pada tingkat paling atas yang merupakan penggambaran yang berfungsi untuk memperlihatkan interkasi/hubungan langsung antara sistem dengan lingkungannya. Diagram konteks menggambarkan sebuah sistem berupa sebuah proses yang berhubungan dengan satu atau beberapa entitas.

d. *Data Flow Diagram* (DFD)

DFD adalah suatu alat pemodelan yang digunakan untuk memodelkan fungsi dari sistem, menggambarkan secara rinci mengenai sistem sebagai jaringan kerja antar fungsi yang berhubungan satu sama lain dengan menunjukkan dari dan kemana data mengalir serta penyimpanannya.

Beberapa symbol digunakan di DFD:

1. Kesatuan luar (*external entity*) atau batas sistem (*boundary*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan masukan atau menerima keluaran dari sistem.

2. Arus data (*data flow*) ini mengalir di antara proses (*process*), simpanan data (*data store*) dan kesatuan luar (*external entity*). Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.
3. Proses (*process*) merupakan kegiatan yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.
4. Simpanan data (*data store*) merupakan simpanan dari data yang dapat berupa suatu file atau database di sistem komputer, suatu arsip atau catatan manual, suatu kotak tempat data di meja seseorang, suatu table acuan manual, dan suatu agenda atau buku.
5. Spesisikasi Proses (*Process Specification (PSPEC)*)

   Spesifikasi proses digunakan untuk menggambarkan semua proses model aliran yang nampak pada tingkat akhir penyaringan. Kandungan dari spesifikasi proses dapat termasuk teksnaratif, gambaran bahasa desain program (*Program Design Language*) dari algoritma proses, persamaan matematika, tabel, digram, atau bagan.
6. Kamus Data (*Data Dictionary*)

   Kamus data adalah catalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data, analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap.Kamus data harus memuat hal-hal berikut ini:

a. Nama arus data.
b. Alias atau nama lain dari data dapat dituliskan bila nama lain ini ada. Alias perlu ditulis karena data yang sama mempunyai nama yang berbeda untuk orang atau departemen satu dengan yang lainnya.
c. Bentuk data, dapat berupa dokumen dasar atau formulir,dokumen hasil cetakan komputer, laporan cetak, tampilan di layar monitor, variabel, parameter dan *field*.
d. Arus data menunjukkan dari mana data mengalir dan kemana data akan menuju.
e. Penjelasan dapat diisi dengan keterangan-keterangan tentang arus data tersebut.
f. Periode menunjukkan kapan terjadinya arus data
g. Volume digunakan untuk mengidentifikasikan besarnya simpanan luar yang akan digunakan, kapasitas dan jumlah dari alat input, alat pemroses dan alat output.
h. Struktur data menunjukkan arus data yang dicatat di kamus data berisi item data apa saja.

e. Skema Relasi

Skema relasi adalah untuk presentasi atribut-atribut dari entitas yang terdapat dalam sistem dan hubungan antar entitas. Pada model ERD, skema relasi merupakan turunan dari ERD.

2.7 Perangkat Lunak Pendukung

Pada sub bab ini akan menjelaskan tentang perangkat lunak pendukung dalam penyelesaian tugas akhir ini.

2.7.1 MATLAB (*Matrix Laboratory*)

MATLAB adalah sebuah bahasa dengan (*high-performance*) kinerja tinggi untuk komputasi masalah teknik. Matlab mengintegrasikan komputasi, visualisasi, dan pemrograman dalam suatu model yang sangat mudah untuk pakai dimana masalah-masalah dan penyelesaiannya diekspresikan dalam notasi matematika yang familiar. Penggunaan Matlab meliputi bidang–bidang:

a. Matematika dan Komputasi
b. Pembentukan Algorithm
c. Akusisi Data,pemrosesan sinyal digital
d. Pemodelan, simulasi, dan pembuatan prototipe
e. Analisa data, explorasi, dan visualisasi
f. Grafik Keilmuan dan bidang Rekayasa

MATLAB merupakan suatu sistem interaktif yang memiliki elemen data dalam suatu array sehingga tidak lagi kita dipusingkan dengan masalah dimensi. Hal ini memungkinkan kita untuk memecahkan banyak masalah teknis yang terkait dengan komputasi, kususnya yang berhubungan dengan matrix dan formulasi vektor, yang mana masalah tersebut merupakan momok apabila kita harus menyelesaikannya dengan menggunakan bahasa level rendah seperti Pascall, C dan Basic.

Nama MATLAB merupakan singkatan dari *Matrix Laboratory*. MATLAB pada awalnya ditulis untuk memudahkan akses perangkat lunak matrik yang telah dibentuk oleh LINPACK dan EISPACK. Saat ini perangkat MATLAB telah menggabung dengan LAPACK dan BLAS *library*, yang merupakan satu kesatuan dari sebuah seni tersendiri dalam perangkat lunak untuk komputasi matrix. Dalam lingkungan perguruan tinggi teknik, Matlab merupakan perangkat standar untuk memperkenalkan dan mengembangkan penyajian materi matematika, rekayasa dan kelimuan. Di industri, MATLAB merupakan perangkat pilihan untuk penelitian dengan produktifitas yang tingi, pengembangan dan analisanya.

Fitur-fitur MATLAB sudah banyak dikembangkan, dan lebih kita kenal dengan nama *toolbox*. Sangat penting bagi seorang pengguna Matlab, toolbox mana yang mandukung untuk *learn* dan *apply* technologi yang sedang dipelajarinya. *Toolbox* ini merupakan kumpulan dari fungsi-fungsi MATLAB (M-*files*) yang telah dikembangkan ke suatu lingkungan kerja MATLAB untuk memecahkan masalah dalam kelas *particular*. Area-area yang sudah bisa dipecahkan dengan *toolbox* saat ini meliputi pengolahan sinyal, system kontrol, neural networks, fuzzy logic, wavelets, dan lain-lain.

Sebagai sebuah system, MATLAB tersusun dari 5 bagian utama:

1. ***Development Environment*.**

   Development Environment merupakan sekumpulan perangkat dan fasilitas yang membantuanda untuk menggunakan fungsi-fungsi dan file-file MATLAB. Beberapa perangkat ini merupakan sebuah graphical user interfaces (GUI).

Termasuk didalamnya adalah MATLAB *desktop* dan *Command Window*, *command history*, sebuah editor dan *debugger*, dan *browsers* untuk melihat *help*, *workspace*, *files*, dan *search path.*

2. **MATLAB *Mathematical Function Library.***

   Mathematical Function Library merupakan sekumpulan algoritma komputasi mulai dari fungsi-fungsi dasar seperti: sum, sin, cos, dan *complex arithmetic*, sampai dengan fungsi-fungsi yang lebih kompleks seperti matrix invers, matrix *eigenvalues, Bessel functions,* dan *fast Fourier transforms*.

3. **MATLAB Language.**

   MATLAB *Language* merupakan suatu high-level matrix/array language dengan control flow statements, functions, data structures, input/output, dan fitur-fitur object-oriented programming. Ini memungkinkan bagi kita untuk melakukan kedua hal baik pemrograman dalam lingkup sederhana untuk mendapatkan hasil yang cepat, dan "pemrograman dalam lingkup yang lebih besar" untuk memperoleh hasil-hasil dan aplikasi yang komplek.

4. ***Graphics.***

   MATLAB memiliki fasilitas untuk menampilkan vector dan matriks sebagai suatu grafik. Didalamnya melibatkan *high-level functions* (fungsi-fungsi level tinggi) untuk visualisasi data dua dikensi dan data tiga dimensi, image processing, animation, dan presentation graphics. Ini juga melibatkan fungsi level rendah yang memungkinkan bagi anda untuk membiasakan diri untuk memunculkan grafik mulai dari benutk yang sederhana sampai dengan tingkatan graphical user

interfaces pada aplikasi MATLAB anda.

5. **MATLAB Application Program Interface (API).**

   ***Application Program Interface* (API).** Merupakan suatu *library* yang memungkinkan program yang telah anda tulis dalam bahasa C dan Fortran mampu berinterakasi dengan MATLAB. Ini melibatkan fasilitas untuk pemanggilan routines dari MATLAB (*dynamic linking*), pemanggilan MATLAB sebagai sebuah *computational engine*, dan untuk membaca dan menuliskan *MAT-files.*