BAB II

LANDASAN TEORI

2.1 Sistem Komputasi

Sistem yang dirancang adalah suatu program aplikasi untuk menyembunyikan sebuah pesan yang berupa teks di dalam sebuah gambar dengan menggunakan metode algoritma F5. Sekuriti merupakan faktor yang sangat penting dalam menyembunyikan sebuah pesan. Serangan yang dilakukan untuk mengetahui isi dari pesan tersebut mempunyai banyak bentuk misalnya memodifikasi informasi, mengakses informasi secara illegal, dan lainnya. *Vulnerabilitas* yaitu kelemahan di dalam sistem sekuriti yang dapat dieksploitasi sehingga menyebabkan kehilangan atau kerugian. Ancaman terhadap keamanan computer dapat disebabkan oleh bencana alam, kesalahan manusia, dan kerusakan perangkat keras atau perangkat lunak.

Ada empat macam ancaman dari sistem komputasi menurut Pfleeger(1997, p22) yaitu :

- Interupsi : dapat menyebabkan suatu asset penting dapat menjadi hilang, tidak dapat digunakan, atau tidak berguna sama sekali.
- Intersepsi : suatu cara di mana beberapa pihak yang tidak berhak dapat memperoleh hak untuk mempunyai akses ke sebuah informasi.
- Modifikasi : suatu cara di mana pihak luar itu tidak hanya dapat memperoleh hak akses, tetapi dapat merusak dengan cara mengubah pesan/informasi tersebut.

- Fabrikasi atau produktif pemalsuan : salah satu factor ancaman yang cukup membahayakan dan pihak luar yang tidak berhak dapat memalsukan sebuah informasi atau program yang akan digunakan.

Karakteristik sekuriti computer dikatakan aman berkaitan dengan tiga tujuan berikut ini yaitu :

- Kerahasiaan, berarti sebuah aset dari sistem computer yang dapat diakses hanya oleh pihak yang berhak (legal).
- Integritas, berarti sebuah aset hanya bisa dimodifikasi oleh pihak-pihak yang berhak atau dengan cara yang legal.
- Ketersediaan (availability), berarti sebuah aset dapat diakses oleh orang yang berhak saja di mana tidak dibatasi hanya mengakses suatu bagian saja.

Kegunaan dari sekuriti ini sangat penting terutama dalam mengirimkan sebuah pesan. Ada dua macam teknik untuk mengirimkan pesan yaitu dengan cara kriptografi dan steganografi. Kedua macam teknik itu akan dibahas di bawah ini.

2.2 Kriptografi, Kriptoanalisis, Steganografi

Kriptografi : merupakan ilmu dari penulisan rahasia yaitu dengan cara mengenkripsi dan mendekripsi sebuah informasi sehingga tidak dapat dibaca oleh pihak yang tidak berhak. Kriptografi ini melibatkan dua proses yaitu enkripsi dan dekripsi. Proses enkripsi yaitu seorang pengirim menyandikan pesannya dengan sebuah metode atau algoritma yang ada. Kemudian penerima pesan akan melakukan proses dekripsi dengan menggunakan metode atau algoritma sesuai dengan yang digunakan oleh pengirim pesan. Setelah melakukan proses tersebut maka penerima pesan dapat melihat pesan tersebut.

Sebagian jenis-jenis kriptografi yang pernah digunakan yaitu(Grabbe 2004, p1) :

- *Monoalphabetic ciphers* yaitu sebuah metode yang pertama kali dikenalkan oleh Julius Caesar yang bekerja dengan cara mensubstitusikan karakter.
- Enkripsi RSA (Rivest-Shamir-Adelman) yaitu sebuah algoritma yang menggunakan dua buah kunci, d dan e, untuk dekripsi dan enkripsi.
- DES (Data Encryption Standard) yaitu algoritma yang sekarang ini banyak digunakan dalam pengenkripsian. DES merupakan blok cipher yang bekerja pada blok dari teks asli (64-bit) dan mengembalikan blok teks cipher dengan ukuran yang sama. Kemudian masing-masing blok dibagi menjadi dua blok, masing-masing 32 bit .

Kriptoanalisis

Kriptoanalisis (cryptanalyst): seseorang yang mempelajari enkripsi dan pesan yang terenkripsi dengan tujuan untuk menemukan isi yang tersembunyi dari sebuah pesan (Pfleeger1997, p22). Kriptoanalisis mencoba untuk memecahkan algoritma dari jenis-jenis kriptografi yang ada. Proses untuk memecahkan algoritma ini disebut kriptoanalisis. Hasil dari proses kriptoanalisis yaitu beruapa isi atau arti dari pesan yang tersembunyi baik tersembunyi maupun tidak tersembunyi di dalam suatu media perantara.

Seorang kriptoanalisis dikatakan mahir bila ia dapat memecahkan algoritma atau metode dari sebuah algoritma yang dianggap cukup sulit bagi masyarakat umum. Kriptoanalisis dapat melakukan hal-hal berikut untuk memecahkan pesan misalnya dengan cara mencoba untuk memecahkan sebuah pesan yang singkat, mencoba untuk

mengenali bentuk dari pesan yang terenkripsi, dan mencoba untuk menemukan kelemahan umum dalam sebuah algoritma enkripsi yang digunakan.

Steganografi

Penyembunyian pesan di dalam sebuah media disebut dengan steganografi (*steganography*). Kata steganografi (*steganography*) berasal dari bahasa Yunani *steganos*, yang artinya 'tersembunyi/terselubung', dan *graphein*, 'menulis' sehingga kurang lebih artinya "menulis (tulisan) terselubung".

Teknik ini meliputi banyak metode komunikasi untuk menyembunyikan pesan rahasia. Metode ini termasuk tinta yang tidak tampak, pengaturan kata, *microdots*, jalur tersembunyi, tanda tangan digital, dan komunikasi spectrum lebar (Sukmawan 2004, p1).

Cara bekerja dari metode steganografi berbeda dengan cara bekerja dari metode kriptografi. Metode kriptografi bekerja dengan cara mengacak sebuah pesan asli dengan sebuah metode atau algoritma enkripsi dari kriptografi sehingga pesan asli tersebut tidak dapat dibaca atau arti dari pesan tersebut menjadi tidak jelas. Lain halnya dengan metode steganografi, metode ini bekerja dengan cara menyembunyikan

sebuah pesan asli melalui sebuah media perantara sehingga pesan asli tersebut tidak dapat dilihat atau disadari oleh pihak yang tidak berhak.

Metode steganografi yang umum dilakukan berdasarkan dua prinsip yaitu informasi dapat diubah menjadi tingkat tertentu dengan mempertahankan kegunaan atau fungsi awalnya dan mata manusia tidak dapat terlalu membedakan perubahan yang terjadi pada media perantara. Metode steganografi bekerja dengan tiga buah media perantara yaitu teks (*text*), gambar (*image*), dan suara (*audio*).

Steganografi di dalam teks bekerja dengan cara menyisipkan baris dan menyisipkan kata-kata. Jumlah penyisipan bervariasi bergantung pada pesan yang akan disembunyikan. Steganografi di dalam audio mempunyai teknik low bit encoding dan phase coding. Low bit encoding melibatkan penyandian pesan dalam LSB (Least Significant Bus) dari file pembawa (*carrier file*). Phase coding membawakan pergeseran phase pembawa suara (*carrier sound*) bergantung pada pesan. Metode yang terakhir yaitu metode steganografi yang akan dibahas sekarang ini yaitu metode steganografi di dalam gambar mempunyai teknik seperti LSB (Least Significant Bus) dan penyembunyian (masking).

Metode steganografi yang dibuat harus memperhatikan empat sisi yang penting yaitu (Cacciaguerra dan Ferretti 2004, p6) :

- Kuat (*robustness*) yaitu metode ini harus dapat mengatasi segala macam serangan (*attack*) yang dilakukan misalnya serangan visual (visual attack). Metode steganografi yang menyembunyikan sebuah pesan di dalam sebuah pesan yang berupa teks mempunyai kelemahan dalam serangan visual karena perubahan pesan yang berupa teks akan lebih mudah terlihat oleh pembaca. Hal ini merupakan salah sastu kelemahan dari metode steganografi yang umum terjadi.
- Tidak terdeteksi (*undetectability*) yaitu lebih mementingkan ke arah perubahan komunikasi yang aman, maksudnya yaitu perubahan yang terjadi pada gambar yang dikarenakan penyembunyian sebuah pesan harus diminimalisasikan. Gangguan (*noise*) harus dibuat seminimal mungkin karena pihak-pihak luar yang ingin mengambil informasi dapat mengetahui bila gangguan (*noise*), misalnya

berupa bintik-bintik tersebut bila gangguan pada gambar terdapat dalam jumlah yang cukup banyak.

- Tidak terlihat (*invisibility*) yaitu penyembunyian lebih didasarkan pada sistem pandangan manusia atau sistem pendengaran manusia. Informasi atau pesan yang disamarkan tidak boleh mempengaruhi perubahan dari media perantara. Hal ini akan lebih terlihat bila media perantara yang digunakan berupa suara (audio) karena perubahan pada suara akan lebih terdengar oleh manusia, maka pihak luar akan lebih mudah mengetahui bahwa suara tersebut berisikan suatu pesan yang tersembunyi.
- Keamanan (*security*) lebih ditujukan pada algoritma yang digunakan. Suatu algoritma untuk penyembunyian pesan dikatakan aman bila isi dari pesan yang disembunyikan tidak dapat diketahui oleh pendeteksian dengan cara apapun.

2.3 JPEG (Proses kompresi dan dekompresi JPEG)

JPEG telah dikenal sebagai salah satu bentuk yang paling popular/terkenal khususnya untuk aplikasi internet dan skema efisiensi *coding* (*efficient coding scheme*) untuk banyak tingkatan (*multilevel*) dari sebuah gambar baik monokrom maupun berwarna. JPEG menggunakan teknik kompresi yaitu *lossy compression* di mana dapat menghasilkan rasio kompresi yang sangat tinggi (1:40). Salah satu JPEG memberikan *encoding framework* yang fleksibel di mana empat model *encoding* dari JPEG(Dave 2002) yaitu :

- *Sequential encoding* yaitu citra diproses dari arah kiri ke kanan atau atas ke bawah berdasarkan pada DCT. Node ini merupakan yang paling umum digunakan.

- *Progressive encoding* yaitu citra di-*encode* dalam beberapa tahap sehingga proses dekompresi dapat dilakukan mel alui *streaming*. Cocok untuk transmisi citra melalui jalur data dengan *bandwidth* yang tidak besar.
- *Lossless coding* yaitu gambar hasil dekompresi dari model ini sama dengan citra aslinya sehingga tidak terjadi penurunan kualitas.
- Hierarki *coding* yaitu citra di-*encode* dalam beberapa resolusi yang berbeda, sehingga pada saat dekompresi dapat dipilih resolusi yang mana yang akan ditampilkan (resolusi rendah/sedang/tinggi).

Ada empat proses dari skema encoding JPEG yaitu (Rangan 2004, p1) :

1. Persiapan gambar yaitu menyiapkan gambar yang akan di-encode
2. DCT (Discrete Cosine Transform) merupakan bentuk *transform encoding* untuk mengurangi ukuran dari bit yang dibutuhkan untuk merepresentasikan tiap 8x8 blok. Pada DCT ini akan dihasilkan 64 koefisien.
3. Kuantisasi yang tidak seragam diaplikasikan pada koefisien DCT (resolusi yang lebih tinggi diberikan ke DC dan koefisien frekuensi yang rendah).
4. *Entropy Encoding* digunakan lebih jauh untuk mengurangi jumlah dari ruangan penyimpanan yang dibutuhkan untuk menyimpan gambar JPEG. *Run length encoding* dapat digunakan untuk keadaan yang lama (*long sequences*) untuk nilai nol yang diproduksi oleh DCT.

Proses kompresi JPEG

Secara umum metode JPEG ini mengikuti proses *transform coding*. *Transform coding* merupakan salah satu metode *lossy compression* yang umum digunakan. Metode ini merupakan gabungan dari beberapa teknik kompresi. Langkah-langkah kompresi dari JPEG yaitu :

- Membagi citra menjadi *subimage* berukuran 8x8 *pixel*.
- Normalisasi yaitu proses mengurangi nilai setiap pixel dengan $2^{n-1} - 1$ (untuk citra dengan *greylevel* 2^n).
- *Forward Transform* yaitu proses transformasi yang digunakan yaitu *Discrete Cosine Transform* (DCT).
- Kuantisasi yaitu proses mereduksi ukuran data pada proses *forward transform* dengan cara menghilangkan informasi yang tidak penting bagi proses persepsi mata terhadap citra. Kuantisasi dilakukan dengan cara membagi setiap elemen matriks dengan matriks tertentu yang sudah didefinisikan sebelumnya, kemudian membulatkan hasilnya ke atas.

Karakteristik-karakteristik dari AC yaitu :

- "Mengambil nilai" dalam jangkauan (*range*) antara -1023 sampai dengan 1023.
- Terdiri atas 10 ukuran kategori.
- Hanya koefisien *non-zero* yang harus di-*encoded*-kan.
- Diproses menurut aturan zig-zag
- Lebih efisien dengan *run length encoding* dari koefisien AC.
- Direpresentasikan sebagai *run*/ukuran dan amplitudo.
- Bila *run*>15, kemungkinan beberapa symbol digunakan.

Proses dekompresi JPEG

- Simbol *decoding*
- Membentuk vector menjadi matriks.
- Dekuantisasi dengan mengalikan tiap elemen matriks dari proses sebelumnya dengan matriks kuantisasi
- Inverse DCT
- Denormalisasi yaitu menambahkan tiap elemen matriks dengan $2^{n-1} - 1$
- Menggabungkan subimage kembali menjadi citra utuh.

2.4 Discrete Cosine Transform (DCT)

Discrete Cosine Transform (DCT) membantu untuk memisahkan gambar menjadi bagian-bagian atau spectral sub-bands dari perbedaan penting (sesuai dengan kualitas visual dari gambar). DCT mirip dengan *discrete Fourier Transform* yaitu mentransformasikan sebuah sinyal atau gambar dari spatial domain menuju domain frekuensi. DCT ini menggunakan matriks 8x8, karena pembagian yang paling efisien.

Persamaan umum untuk 1D (N jumlah data) DCT didefinisikan sebagai berikut :

$$F(u) = (2/N) \sum_{i=0}^{½N-1} A(i).\cos[\pi.u/2.N(2i+1)]\, f(i)$$

dan invers dari 1D DCT yaitu $F^{-1}(u)$:

$$A(i) = 1/\sqrt{2} \text{ for } \xi = 0 \text{ dan } 1 \text{ untuk otherwise}$$

Persamaan umum untuk 2D (gambar NxM) DCT didefinisikan dengan persaman berikut :

$$F(u,v) = (2/N)^{1/2}\,(2/M)^{1/2} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} A(i).A(j).\cos[\pi.u/2N\,(2i+1)]\cos[\pi.v/2M\,(2j+1)].f(i,j)$$

dan invers dari 2D DCT yaitu $F^{-1}(u,v)$:

$$A(\xi) = 1/\sqrt{2} \text{ for } \xi = 0 \text{ dan } 1 \text{ untuk otherwise}$$

ξ = Koefisien JPEG

Operasi dasar dari DCT sebagai berikut :

- Masukkan gambar yaitu NxM
- f(i,j) yaitu intensitas dari piksel pada baris i dan kolom j.
- F(u,v) yaitu koefisien DC pada baris k1 dan kolom k2 dari matriks DCT.
- Secara umum untuk gambar, banyak dari energi sinyal tidak benar (*lies*) saat frekuensi rendah. Hal ini terlihat pada ujung kiri atas dari DCT.
- Kompresi dicapai bila nilai kanan bawah merepresentasikan frekuensi yang lebih tinggi.
- Masukan dari DCT yaitu *array integer* dari 8x8. Array ini mengandung tingkatan piksel dari *gray scale*.
- 8 bit piksel mempunyai tingkatan dari 0 sampai dengan 255.
- F[0,0] menjelaskan koefisien AC dan DC.
- *Array* keluaran dari koefisien DCT mengandung *integer* yang mempunyai jangkauan -1024 sampai 1023. Hal tersebut memudahkan perhitungan implementasi dan lebih efisien untuk DCT sebagai sekelompok dari fungsi dasar

(*basis functions*) yang memberikan masukan ukuran *array* (8x8) untuk dapat dihitung kembali dan disimpan.

2.5 Sejarah Metoda F5

Sejarah metode F5 ini sebenarnya berawal dari metode dengan nama F3. Metode ini diciptakan oleh Andreas Westfeld. Metode ini bekerja dengan beberapa tahap yaitu :

1. Metode ini selain menulis ulang (*overwriting*) bits juga mengurangi nilai absolute dari koefisien supaya LSBnya tidak sama, kecuali koefisien-koefisien dengan nilai nol, di mana tidak dapat dikurangi lagi nilai absolutnya. Oleh karena itu, koefisien nol (*zero*) tidak digunakan dalam steganografi. LSB dari koefisien *non-zero* sesuai dengan pesan rahasia setelah proses penyimpanan, tetapi metode ini tidak menulis ulang (*overwrite*) bits karena pengujian Chi-Square dapat dengan mudah mengetahui perubahan yang terjadi. Jadi F3 ini menggunakan koefisien-koefisien yang bernilai 1.
2. Beberapa bits yang disimpan akan terkena proses penyusutan (*shrinkage*). Penyusutan akan bertambah setiap F3 melakukan pengurangan terhadap nilai absolute dari 1 dan -1 yang menghasilkan nilai 0. Penerima tidak dapat membedakan koefisien nol di mana tidak berguna dalam teknik steganografi, dari nilai 0 yang dihasilkan oleh penyusutan. Metode ini melewati seluruh koefisien nol, maka pengirim berulang kali menyimpan bit yang berubah ketika pengirim mengetahui bahwa ia menghasilkan nilai nol.

Jadi penyusutan hanya terjadi bila kita menyimpan bit nol. Pengulangan dari bit nol mengubah rasio dari nilai steganografi. Oleh karena itu, proses penyimpanan F3 menghasilkan koefisien genap daripada koefisien ganjil. Interpretasi steganografi dari koefisien dengan nilai 1 atau -1 yaitu 1 karena LSBnya yaitu 1. Untuk hal ini, fungsi penyimpanan akan tetap menyimpan nilai koefisien tersebut ketika menyimpan angka 1.

Bila kita mengabaikan penyusutan, maka akan banyak muncul jumlah koefisien genap. Hal ini menyebabkan kerugian bagi penerima yaitu penerima hanya mendapatkan sebagian dari pesan.

2.6 Metode yang Digunakan (F3, F4, dan F5)

Metode F3:

Metode F3 mempunyai 2 kelemahan yaitu :

1. Penyusutan ekslusif nilai nol pada steganografi, maka F3 secara efektif menyimpan nilai nol daripada satu dan menghasilkan statistic ganjil yang mudah dideteksi dalam histogram.
2. Histogram dari *file* JPEG mengandung lebih banyak koefisien ganjil daripada koefisien genap (tanpa nol).

Metode F4:

Metode ini menghilangkan dua kelemahan ini dengan cara memetakan koefisien negative ke nilai steganografi yang terbalik; koefisien genap yang negative merepresentasikan satu, koefisien ganjil yang negative menghasilkan nol; sedangkan koefisien genap yang positif mereprentasikan nol, koefisien ganjil yang positif menghasilkan satu.

Metode F5:

Tidak seperti media *stream* (seperti *video conferences*), gambar hanya menyediakan kapasitas steganografi yang terbatas. Pada umumnya, sebuah penyimpanan pesan tidak memerlukan kapasitas seluruhnya (bila pesan tersebut sudah sesuai dengan media penyimpanannya). Oleh karena itu, ada bagian dari *file* yang tidak digunakan. Untuk mencegah serangan, fungsi penyimpanan harus menggunakan media perantara secara teratur. Pada metode F4, terlihat lebih maju dalam operasi penyimpanan (*embedding operation*). Akan tetapi, metode ini mempunyai kelemahan terutama dalam serangan intensitas perubahan pada tingkat yang tinggi (*high change density*). Jadi untuk menanggulangi hal tersebut, maka proses penyebaran dari penyimpanan harus sama di seluruh tempat pada gambar.

Andreas memikirkan untuk melakukan penyempurnaan dengan menambahkan fungsi dan fasilitas pada metode F4. Metode yang baru itu disebut metode F5. Keunggulan metode F5 ini yaitu :

1. *Permutative straddling* merupakan salah satu proses memencarkan (*scatter*) pesan melewati seluruh medium perantara. Banyak dari proses penyebaran tersebut mempunyai kompleksitas waktu yang buruk. Proses itu bergerak lambat bila mencoba untuk menghabiskan seluruh kapasitas dari steganografi. *Straddling* itu mudah bila kapasitas dari medium perantara diketahui secara pasti, tetapi penyusutan untuk F4 tidak dapat dikira-kira karena F4 bergantung pada bit mana yang disimpan untuk posisi tertentu.

   Proses dari mekanisme *straddling* yaitu pertama, mengacak semua koefisien dengan permutasi. Kemudian F5 menyimpan ke keadaan permutasi.

Penyusutan tidak mengubah jumlah dari koefisien-koefisien (hanya nilainya saja). Permutasi yang dilakukan bergantung pada kunci yang diambil dari kata kunci yang digunakan. F5 mengirimkan koefisien steganografi yang berubah pada keadaan asli menuju Huffman *coder*. Dengan kunci yang benar, penerima dapat mengulangi permutasinya. Permutasi mempunyai kompleksitas waktu yang linear $O(n)$.

2. Matriks encoding yaitu sebuah teknik untuk meningkatkan efisiensi penyimpanan. Bila banyak dari kapasitas steganogram tidak digunakan, maka matriks encoding ini mengurangi jumlah dari perubahan yang terjadi.

Metode F5 ini mempunyai langkah-langkah untuk implementasi dalam proses penyimpanan sebuah pesan teks dalam sebuah gambar sebagai berikut yaitu :

1. Mulai kompresi dari JPEG. Berhenti setelah kuantisasi dari koefisien-koefisien.
2. Inisialisasi sebuah *random number generator* dengan kunci yang diambil dari kata kunci yang digunakan.
3. Lakukan permutasi (dua ukuran : *random generator* dan jumlah koefisien, termasuk koefisien nol).
4. Tentukan ukuran dari k dari kapasitas medium perantara dan panjang dari pesan rahasia.
5. Hitung panjang *code word* $n = 2^k - 1$. *Code word* adalah banyaknya byte yang dapat dipakai untuk menyisipkan pesan dokumen.
6. Simpan pesan rahasia dengan (1,n,k) matriks *encoding*.

b. Hash *buffer* tersebut (*generate* nilai hash dengan k *bit-places*).

c. Tambahkan k bits berikutnya dari pesan pada nilai hash (bit per bit, xor).

d. Bila jumlahnya sama dengan 0, *buffer* dibiarkan tetap sedangkan bila jumlah indeks *buffer* 1 ... n, nilai absolute dari tiap elemen harus dikurangi.

e. Pengujian untuk penyusutan, misalnya apakah menghasilkan nilai nol. Bila ya, sesuaikan *buffer* (hapus nilai 0 dengan membaca sekali lagi koefisien *non-zero*, misalnya ulangi langkah 6a dari awal dengan koefisien yang sama). Bila penyusutan tidak terjadi, maju ke koefisien baru di belakang *buffer* yang sekarang ini. Bila masih ada data pesan, lanjutkan ke langkah 6a.

7. Lanjutkan dengan kompresi JPEG (Huffman coding, dan lainnya).

Kelebihan-kelebihan metode F5 ini dibandingkan dengan teknik steganografi yang lainnya yaitu mempunyai kapasitas steganografi yang tinggi (*high steganographic capacity*), efisiensi yang tinggi melalui *matrix encoding*, mencegah serangan visual (*visual attack*), menahan (*resistant*) terhadap serangan statistic (*chi square*), menggunakan file tipe JPEG sebagai medium perantara (umum digunakan di email), dan *source code* secara umum tersedia.

2.7 Steganalisis

Steganalisis yaitu suatu cara yang dilakukan untuk menentukan pesan yang asli dan media perantara yang digunakan sebagai stego (Jeong-Jae 2004, p1), sedangkan orang yang memecahkan teknik dari steganografi yang digunakan disebut steganalisis.

Serangan visual (*visual attack*)

Pada umumnya, mustahil untuk mendeteksi steganografi secara langsung dengan melihat keseluruhan gambar pada tampilan di layer atau monitor, tetapi hal ini dapat diatasi bila LSB dari sebagian gambar dimodifikasi. Yang dimaksud memodifikasi sebagian gambar yaitu meningkatkan LSB dengan cara mengatur semua bit dalam byte yang merepresentasikan dari sebagian warna untuk nilai dari LSB. Jadi misalnya intensitas maksimum dilambangkan dengan satu (1) dan intensitas minimum atau tanpa intensitas dilambangkan nol (0), contoh hitam. Oleh karena itu, dapat diketahui gambar mana yang menggunakan algoritma steganografi atau tidak.

Analisis statistik

Serangan visual dapat bekerja hanya pada file bitmap. Secara umum, serangan visual tidak akan bekerja pada file yang bertipe JPEG. Hal ini terjadi karena modifikasi LSB untuk JPEG terjadi pada domain frekuensi. Analisis statistic mempunyai dua buah metode yaitu :

Metode χ^2

Sebuah metode yang berhasil mendeteksi steganografi yaitu berdasarkan pada pengujian χ^2. pengujian statistic dalam steganalisis untuk steganografi pertama kali dilakukan oleh Andreas Westfield dan Andreas Pfitzmann (Moerland 2004, p10).

Ketika LSB digunakan untuk menyembunyikan informasi, bit ke-2 sampai ke-7 masih sama. Sekarang ada dua byte yang mempunyai spesifikasi bit ke-2 sampai ke-7, satunya dengan LSB=0 dan yang lain LSB=1. Dua byte ini disebut *pairs of value* atau PoV. Ini berarti apapun yang kita lakukan pada LSB, jumlah total perubahan dari dua anggota sebuah PoV akan tetap sama. *Image* dapat dikatakan terdistorsi jika persentase kesalahan lebih besar dari 10%.

Analisis dari gambar *true color*

Kunci observasi untuk serangan ini yaitu memperhitungkan jumlah dari warna yang unik di dalam sebuah gambar di mana ciri pentingnya lebih kecil dari jumlah piksel di dalm sebuah gambar. Hal ini akan terlihat pada gambar yang *true color*.

Penyeleksian bit (*bit selection*)

Prosedur yang paling nyata yaitu memulai untuk penyimpanan (*embedding*) byte pertama dari data gambar, kemudian secara sekuensial berjalan melalui gambar dengan menggunakan setiap *redundant bit* yang ditemukan. Kemudian berhenti setelah pesan tersebut telah selesai dikelilingi atau akhir dari data gambar telah dicapai.

2.8 *System Development Life Cycle* (SDLC) dan *State Transition Diagram* (STD)

Metode siklus hidup pengembangan sistem atau sering disebut dengan *System Development Life Cycle* (SDLC) merupakan suatu tahapan-tahapan metode untuk merancang sebuah program aplikasi perangkat lunak. Nama lain dari metode SDLC yaitu metode *waterfall*. Metode ini disebut *waterfall* karena model dari langkah-langkah yang dilakukan mirip dengan air terjun (bertingkat). Jadi proses harus dilakukan secara bertingkat untuk menghasilkan suatu program aplikasi yang baik.

Perancangan aplikasi perangkat lunak dengan metode SDLC dilakukan dalam enam tahap. Tahapan-tahapan yang harus dilakukan terdiri dari perencanaan, analisis, desain, pengkodean, pengujian, dan pemeliharaan.

Berikut ini dijelaskan setiap tahapan SDLC tersebut, yaitu :

1. **Perencanaan**

   Perencanaan adalah suatu kegiatan untuk menentukan program aplikasi yang akan dirancang, tempat program aplikasi akan dirancang dan dijalankan, dan siapa yang akan merancang program aplikasi tersebut.

2. **Analisis**

   Analisis adalah suatu kegiatan untuk menentukan tentang topik dari permasalahan yang sedang dihadapi dan bagaimana cara pemecahan atau solusi masalah tersebut.

3. **Desain**

   Desain adalah suatu kegiatan untuk menentukan konsep dasar rancangan dari suatu program yang akan dibuat sehingga diharapkan dengan desain yang baik, maka pengguna akan merasa nyaman dala menggunakan program aplikasi yang dirancang tersebut.

4. **Pengkodean**

   Pengkodean adalah suatu kegiatan yang berguna untuk mengimplementasikan konsep dasar dari tahap sebelumnya (desain) ke dalam bahasa pemrograman.

5. **Pengujian**

   Pengujian adalah suatu kegiatan untuk mencari kelemahan dan kesalahan yang terjadi pada program aplikasi dan kemudian memperbaiki kesalahan atau kelemahan

tersebut. Ada beberapa metode penguji untuk menguji fungsi-fungsi dari suatu program aplikasi. Metode-metode tersebut adalah sebagai berikut :

a. **Metode pengujian *White-box***

Metode ini menerapkan pengujian terhadap struktur logika program dan detail prosedural. Pengujian dilakukan terhadap setiap baris kode program untuk meyakinkan bahwa semua operasi internal bekerja sesuai dengan spesifikasi dan semua komponen internal telah dicoba.0

b. **Metode pengujian *Black-box***

Metode ini merupakan pengujian interface dari perangkat lunak oleh pemakai untuk mengetahui spesifikasi dari suatu fungsi dalam program aplikasi. Pengujian dilakukan dengan memberi *input* pada program aplikasi, kemudian diproses, dan hasil keluarannya dibandingkan apakah telah sesuai dengan kebutuhan fungsional yang diinginkan pemakai.

c. **Metode pengujian *Gray-box***

Metode ini merupakan gabungan dari metode pengujian *white-box* dan metode pengujian *black-box* yaitu memvalidasi *interface* perangkat lunak dan pemilihan beberapa logika internal.

6. **Pemeliharaan**

Pemeliharaan adalah suatu kegiatan yang berguna untuk memastikan bahwa program aplikasi akan berjalan dengan baik sehingga diperlukan pemeliharaan secara berkala.

***State Transition Diagram* (STD)**

State Transition Diagram merupakan sebuah *modeling tool* yang digunakan untuk mendeskripsikan sistem yang memiliki ketergantungan terhadap waktu. STD merupakan suatu kumpulan keadaan atau atribut yang mencirikan suatu keadaan pada waktu tertentu.

Komponen-komponen utama STD adalah:

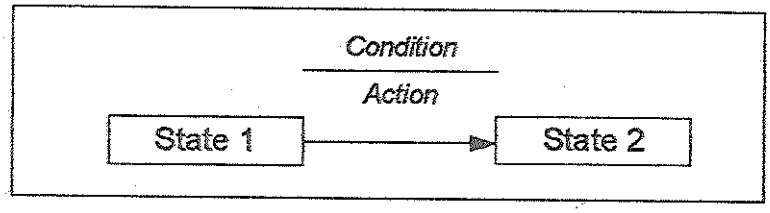
1. *State*, disimbolkan dengan 

   State merepresentasikan reaksi yang ditampilkan ketika suatu tindakan dilakukan. Ada dua jenis *state* yaitu: *state* awal dan *state* akhir. *State* akhir dapat berupa beberapa *state*, sedangkan *state* awal tidak boleh lebih dari satu.

2. *Arrow*, disimbolkan dengan

   Arrow sering disebut juga dengan transisi state yang diberi label dengan ekspresi aturan, label tersebut menunjukkan kejadian yang menyebabkan transisi terjadi.

3. *Condition* dan *Action*, disimbolkan dengan



Untuk melengkapi STD diperlukan 2 hal lagi yaitu *condition* dan *action*. *Condition* adalah suatu *event* pada lingkungan eksternal yang dapat dideteksi oleh sistem, sedangkan *action* adalah yang dilakukan oleh sistem bila terjadi perubahan *state* atau merupakan reaksi terhadap kondisi. Aksi akan menghasilkan keluaran atau tampilan.