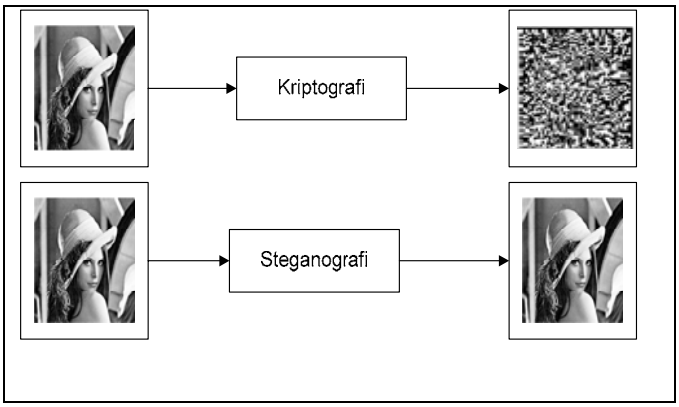
BAB II
LANDASAN TEORI

2.1 Definisi Steganografi

Steganografi (*steganography*) adalah ilmu dan seni menyembunyikan pesan rahasia (*hiding message*) sedemikian sehingga keberadaan (eksistensi) pesan tidak terdeteksi oleh indera manusia [5]. Kata steganorafi berasal dari Bahaya Yunani yang berarti "tulisan tersembunyi" (*covered writing*). Steganografi membutuhkan dua properti: wadah penampung dan data rahasia yang akan disembunyikan. Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, suara, teks, dan video. Data rahasia yang disembunyikan juga dapat berupa citra, suara, teks, atau 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 datanya seolah-olah berantakan, tetapi dapat dikembalikan ke bentuk semula. Ilustrasi mengenai perbedaan kriptografi dan steganografi dapat dilihat pada gambar 2-1.

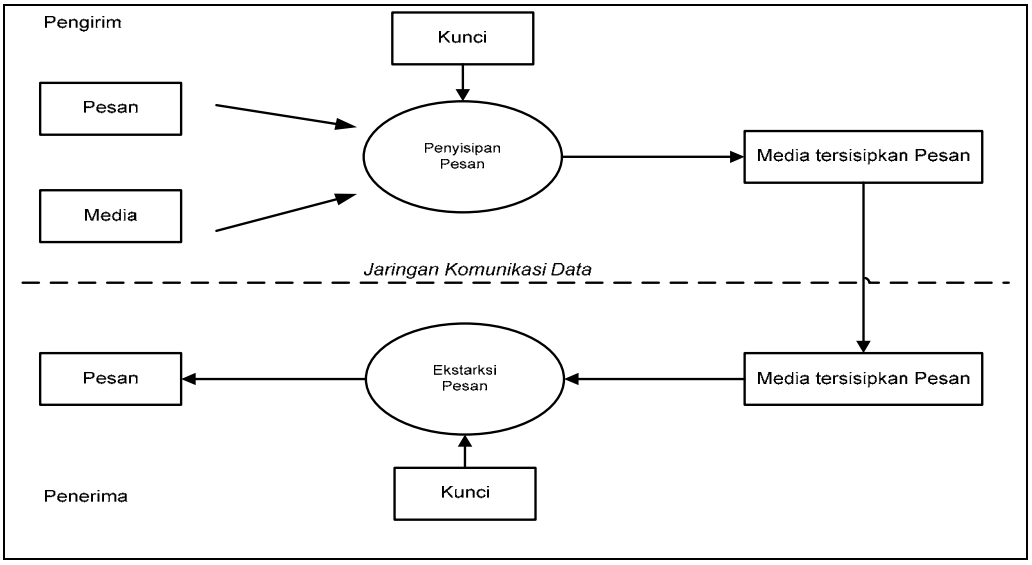


Gambar 2.1. Ilustrasi kriptografi dan steganografi pada citra digital.

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, metoda steganografi ini dapat diterapkan pada berbagai media digital. Hasil keluaran dari steganografi ini memiliki bentuk persepsi yang sama dengan bentuk aslinya, tentunya persepsi di sini sebatas oleh kemampuan indera manusia, tetapi tidak oleh komputer atau perangkat pengolah digital lainnya. Ilustrasi mengenai proses steganografi dapat dilihat pada gambar 2-2.

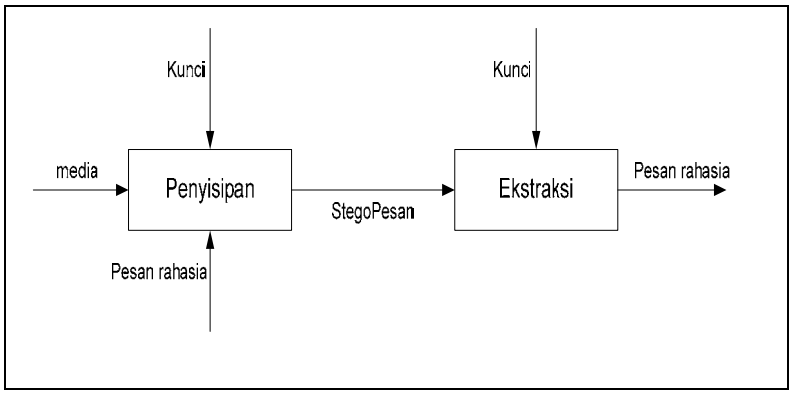
Penyembunyian data rahasia ke dalam media digital mengubah kualitas media tersebut. Kriteria yang harus diperhatikan dalam penyembunyian data diantaranya adalah [5]:

1. *Fidelity*. Mutu media penampung tidak jauh berubah. Setelah penambahan data rahasia pengamat tidak mengetahui kalau di dalam media tersebut terdapat data rahasia.
2. *Recovery*. Data yang disembunyikan harus dapat diungkapkan kembali (*recovery*). karena tujuan steganografi adalah *data hiding*, maka sewaktu-waktu data rahasia didalam media penampung harus dapat diambil kembali untuk digunakan lebih lanjut.



Gambar 2.2 Diagram Sistem Steganografi

Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, suara, teks, dan video. Sedangkan data rahasia yang disembunyikan dapat berupa berkas apapun. Media yang telah disisipi data disebut stegomessage. Proses penyembunyian data ke dalam media disebut penyisipan (*embedding*), sedangkan proses sebaliknya disebut ekstraksi. Proses tersebut dapat dilihat pada gambar 2-3. Penambahan kunci yang bersifat opsional dimaksudkan untuk lebih meningkatkan keamanan [10].



Gambar 2.3 Proses penyisipan dan ekstraksi dalam steganografi

Sejarah Steganografi

Steganografi sudah dikenal oleh bangsa Yunani. Herodatus, penguasa Yunani, mengirim pesan rahasia dengan menggunakan kepala budak atau prajurit sebagai media. Dalam hal ini, rambut budak dibotaki, lalu pesan rahasia ditulis pada kulit kepala budak. Ketika rambut budak tumbuh, budak tersebut diutus untuk membawa pesan rahasia di balik rambutnya.

Bangsa Romawi mengenal steganografi dengan menggunakan tinta tak-tampak (*invisible ink*) untuk menuliskan pesan. Tinta tersebut dibuat dari campuran sari buah, susu, dan cuka. Jika tinta digunakan untuk menulis maka tulisannya tidak tampak. Tulisan di atas kertas dapat dibaca dengan cara memanaskan kertas tersebut. Saat ini di negara-negara yang melakukan penyensoran informasi, steganografi sering digunakan untuk menyembunyikan pesan-pesan melalui gambar (*images*), video, atau suara (*audio*) [10].

Aplikasi Steganografi

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

- *Tamper-proofing* dimana steganografi digunakan sebagai alat untuk mengidentifikasikan atau alat indikator yang menunjukkan data *host* telah mengalami perubahan dari aslinya.
- *Feature location* dimana 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.

- *Annotation/caption* dimana steganografi hanya digunakan sebagai keterangan tentang data digital itu sendiri.
- *Copyright-Labeling* dimana steganografi dapat digunakan sebagai metoda untuk penyembunyian label hak cipta pada data digital sebagai bukti otentik kepemilikan karya digital tersebut.

Media Steganografi

Steganografi menggunakan sebuah berkas 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, audio, dan gambar.

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 di*decode* atau diekstrak kembali

Steganografi pada Gambar

Steganografi pada gambar adalah metode yang paling banyak digunakan secara luas didunia 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 di*encode* kedalam aliran bit untuk disembunyikan didalam gambar digital. Perkembangan dari metoda ini sangat pesat, didukung dengan semakin canggihnya komputer grafik yang *powerful*, dan software 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.

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.

Metode-metode Steganografi pada Audio

Terdapat empat *encoding* yang paling terkenal dalam menyembunyikan data pada format ini, yaitu : *low-bit encoding, phase-coding, spread spectrum* dan *echo data hiding*.

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, $2^7 = 128$) , artinya yang paling tidak signifikannya adalah yang memiliki nilai terrendah (misal, $2^0 = 1$). Sebagai contoh, kita akan menyisipkan karakter ‘G’ pada 8 byte berkas *carrier* atau bit dari media pembawa. *Low Bit Encoding* ditandai dengan garis bawah.

10010101 00001101 11001001 10010110

00001111 11001011 10011111 00010000

01000111 adalah bentuk huruf G dalam biner. kedelapan bit ini bisa dituliskan kedalam *Low Bit Encoding* dari tiap-tiap byte pada kedelapan *carrier* seperti berikut ini :

1001010*0* 0000110*1* 1100100*0* 1001011*0*

0000111*0* 1100101*1* 1001111*1* 0001000*1*

Pada contoh diatas, hanya sebagian dari *Low Bit Encoding* yang berubah (ditunjukkan dengan karakter miring). Dengan mengubah *Low Bit Encoding* , nilai dari byte tidak akan berubah banyak, sehingga akan sulit dideteksi oleh telinga manusia.

Phase Coding

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 system pendengaran manusia untuk merasakan fase absolut yang mengabaikan suara yang lebih lemah jika dua suara itu dating bersamaan.

Spread Spcetrum

Encoding dari *spread spectrum* pada audio ini melibatkan keseluruhan dari spektrum frekuensinya. Kemudian mentransmit audio tersebut melalui frequensi 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 s*pread spectrum* ini adalah teknik paling aman untuk mengirimkan pesan didalam audio, tapi teknik ini juga menimbulkan *noise* secara acak pada audio, dan hal ini berarti ada beberapa data asli dari audio tersebut hilang.

Echo Data Hiding

Teknik e*cho data hiding* dilakukan dengan cara menambahkan data kedalam sumber sinyal dengan cara menggunakan echo. 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.

2.2 Definisi Kriptografi

Kriptografi adalah ilmu sekaligus seni untuk menjaga kerahasiaan pesan (data atau informasi) dengan cara menyamarkannya menjadi bentuk yang tidak dapat dimengerti. Tujuan penyandian adalah agar isi pesan tidak dapat dimengerti oleh orang yang tidak berhak. Kehidupan saat ini dikelilingi oleh kriptografi,

diantaranya mulai dari transmisi pada teknolgi nirkabel *bluetooth*, ATM, telepon genggam, jaringan komputer di kantor, *internet*, sampai ke pangkalan militer.

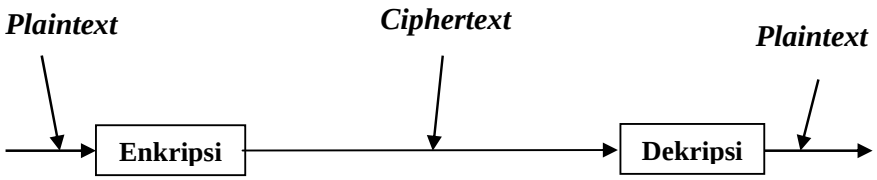
Kriptografi merupakan ilmu dan seni untuk menjaga pesan agar aman. "*Crypto*" berarti "*secret*" (rahasia) dan "*graphy*" berarti "*writing*" (tulisan) [6]. Para pelaku atau praktisi kriptografi disebut *cryptographers*. Sebuah algoritma kriptografik (*cryptographic algorithm*), disebut *cipher*, merupakan persamaan matematik yang digunakan untuk proses enkripsi dan dekripsi. Biasanya kedua persamaan matematik (untuk enkripsi dan dekripsi) tersebut memiliki hubungan matematis yang cukup erat.

2.2.1 Terminologi Dasar Kriptografi

Untuk mengenal kriptografi lebih jauh, ada beberapa terminologi yang harus diketahui terlebih dulu sebagai berikut :

1. Plainteks (*plaintext* atau *cleartext*, artinya teks jelas yang dapat dimengerti), yaitu pesan asli yang akan disandikan.
2. Chiperteks (*ciphertext* atau *cryptogram*, artinya teks tersandi), yaitu pesan hasil penyandian.
3. Enkripsi (*encryption* atau *enchipering*), yaitu proses penyandian dari plainteks ke chiperteks.
4. Dekripsi (*decryption* atau *dechipering*), yaitu proses pembalikan dari chiperteks ke plainteks.

Secara sederhana proses enkripsi/ dekripsi dapat digambarkan seperti pada gambar 2.4.



Gambar 2.4 Proses Enkripsi dan Dekripsi Sederhana

5. Algoritma kriptografi (*cipher*), yaitu aturan untuk *enciphering* dan *deciphering*. Algoritma kriptografi dapat diartikan juga sebagai fungsi matematika yang digunakan untuk enkripsi dan dekripsi.
6. Kunci, adalah parameter yang digunakan untuk transformasi *ciphering*.
7. Sistem Kriptografi (*cryptosystem*), yaitu algoritma kriptografi beserta kuncinya.
8. Penyusup (*intruder* / *interceptor*), adalah orang yang mencoba menangkap pesan selama ditransmisikan. Nama lain dari penyususp ini adalah *enemy*, *eavesdropper*.
9. Kriptografer, yaitu orang yang menggunakan algoritma kriptografi untuk merahasiakan pesan dan mendekripsikannya kembali.
10. Kriptanalisis (*cryptanalysis*), adalah ilmu dan seni untuk memecahkan chiperteks, berupa proses untuk memperoleh *plaintext* dari *ciphertext* tanpa mengetahui kunci yang diberikan. Pelakunya disebut kriptanalis
11. Kriptanalis. Persamaan kriptografer dan kriptanalis adalah bahwa keduanya sama-sama menerjemahkan *ciphertext* menjadi *plaintext*. Perbedaannya,

kriptografer bekerja atas legitimasi pengirim atau penerima pesan, sedangkan kriptanalis bekerja atas nama penyusup yang tidak berhak.

12. Kriptologi (*cryptology*), yaitu studi mengenai kriptografi dan kriptanalisis.

2.2.2 Tujuan Kriptografi

Kriptografi bertujuan untuk memberi layanan keamanan yang juga dinamakan aspek-aspek keamanan. Keamanan komputer jaringan melingkupi empat aspek utama yaitu *privacy*, *integrity*, *autentication* dan *availabilty* .

Selain untuk menjaga kerahasiaan (*confidentiality*) pesan, kriptografi juga digunakan untuk menangani masalah keamanan yang mencakup dua hal berikut:

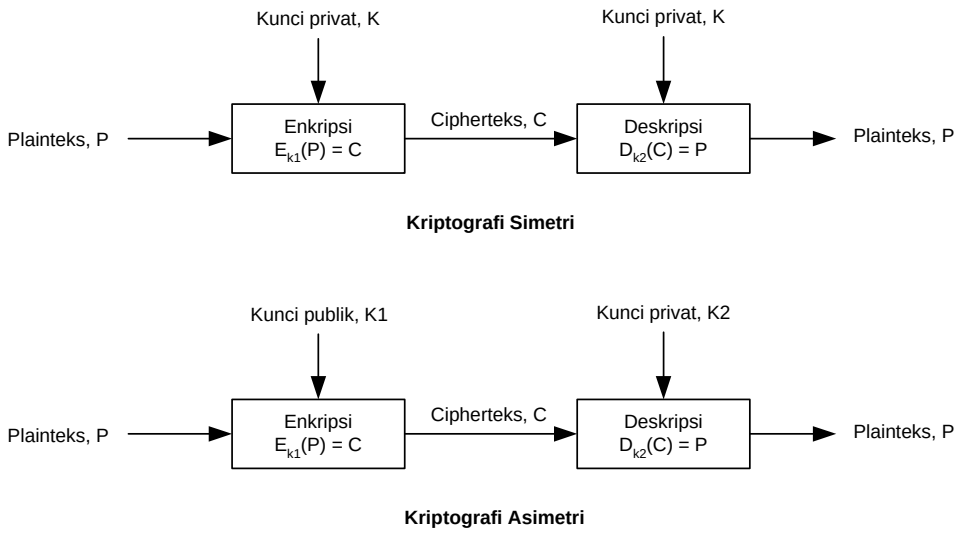
1 Keabsahan pengirim (*user authentication*). Hal ini berkaitan dengan keaslian pengirim. Dengan kata lain, masalah ini dapat diungkapkan sebagai pertanyaan:“Apakah pesan yang diterima benar-benar berasal dari pengirim yang sesungguhnya?”

2 Keaslian pesan (*message authentication*). Hal ini berkaitan dengan keutuhan pesan (*data integrity*). Dengan kata lain, masalah ini dapat diungkapkan sebagai pertanyaan: “Apakah pesan yang diterima tidak mengalami perubahan (modifikasi)?”

3 Anti-penyangkalan (*nonrepudiation*).Pengirim tidak dapat menyangkal (berbohong) bahwadialah yang mengirim pesan.

2.2.3 Kunci Dalam Kriptografi

Berdasarkan kuncinya kriptografi dibagi menjadi dua yaitu kriptografi kunci simetris dan asimetris, sebagai penekanan dapat dilihat pada gambar 2.5.



Gambar 2.5 Kunci dalam Kriptografi

Jika kunci enkripsi sama dengan kunci dekripsi, maka sistem kriptografinya disebut **sistem simetri** atau **sistem konvensional**. Algoritam kriptografinya disebut algoritma simetri atau algoritma konvensional. Contoh algoritma simetri: *DES* (*Data Encyption Standard*).

Beberapa sistem kriptografi menggunakan kunci yang berbeda untuk enkripsi dan dekripsi. Misalkan kunci enkripsi adalah *K*1 dan kunci dekripsi yang adalah *K*2, yang dalam hal ini *K1* ¹ *K2*. Sistem kriptograsi semacam ini dinamakan sistem **sistem asimetri** atau **sistem kunci-publik**. Algoritam kriptografinya disebut algoritma asimetri atau algoritma kunci-publik. Contoh algoritma asimetri: *RSA* (*Rivest-Shamir-Adleman*)

2.2.4 Pembangkit Bilangan Pseudorandom

Bilangan acak (*random*) banyak digunakan dalam kriptografi. Pembangkitan bilangan acak pada steganografi, dapat digunakan untuk menentukan kunci penyisipan dan ekstraksi data dari berkas media. Namun, sangat sulit untuk menghasilkan bilangan yang benar-benar acak dengan menggunakan komputer. Komputer hanya mampu menghasilkan bilangan semu acak (*pseudorandom*). Deret bilangan *pseudorandom* adalah deret bilangan yang kelihatan acak dengan kemungkinan pengulangan yang sangat kecil atau periode pengulangan yang sangat besar.

*2.2.4.1 **Linear Congretial Generator***

Salah satu algoritma pembangkitan bilangan pseudorandom adalah *Linear Congruental Generator* (*LCG*). Algoritma ini diciptakan oleh D. H. Lehmer pada tahun 1951. Deret bilangan bulat dalam *LCG* diformulasikan sebagai berikut :

$$Zi = (aZi_{-1} + c) \pmod{m} \qquad (2.1)$$

Dalam hal ini :

Zi : bilangan bulat ke-*i*

a : bilangan pengali

c : bilangan penambah

m : modulus

*Z*0 : nilai awal berupa bilangan bulat tak negatif

Dengan demikian nilai *Zi* terdefinisi pada :

0 = *Zi* = *m*-1 , *i* = 1, 2, 3, …

Untuk memulai bilangan acak ini dibutuhkan sebuah bilangan bulat *Z*0, yang dijadikan sebagai nilai awal (bibit pembangkitan). Bilangan acak pertama yang dihasilkan selanjutnya menjadi bibit pembangkitan bilangan bulat acak selanjutnya. Jumlah bilangan acak yang tidak sama satu sama lain (unik) adalah sebanyak *m*. Semakin besar nilai m, semakin kecil kemungkinan akan dihasilkan nilai yang sama.

Sebagai contoh : Z*n* = $(7Z_{i-1} + 11) \bmod 17 ; Z_0 = 0$

Rangkaian bilanagan acak yang dibangkitkan oleh persamaan diatas adalah

n	**Z_n**
0	0
1	11
2	3
3	15
4	14
5	7
6	9
7	6
8	2
9	8
10	16
11	4
12	5
13	12
14	10
15	13
16	0
17	11
18	3
19	15
20	14
21	7
22	9
23	6
24	2

Priode *LCG* diatas adalah 16, sebab setelah 16 kali perhitungan bilanagan acaknya terulang kembali. *LCG* mempunyai priode tidak lebih besar dari *m,* dan pada kebanyakan kasus periode penuh (*m - 1*) jika memenuhi syarat berikut [6]:

1. *c* relatif prima terhadap *m.*
2. *a – 1* dapat dibagi dengan semua faktor prima dari *m.*
3. *a – 1* adalah kelipatan 4 jika *m* adalah kelipatan 4.

4. $m >$ maks (a, b, Z_0).
5. $a > 0$, $c > 0$.

Meskipun *LCG* secara teoritis mampu menghasilkan bilangan acak yang cukup, namun *LCG* sangat sensitif terhadap pemilihan nilai- nilai *a*, *c* dan *m*. Pemilihan nilai –nilai yang buruk dapat mengarah pada implementasi *LCG* yang tidak bagus.

Keunggulan *LCG* terletak pada kecepatannya dan hanya membutuhkan sedikit operasi bit. Akan tetapi, *LCG* tidak dapat digunakan untuk kriptografi karena bilangan acaknya dapat diprediksi urutan kemunculannya. Oleh karena itu *LCG* tidak aman digunakan untuk kriptografi. Namun demikian, *LCG* tetap berguna untuk aplikasi non-kriptografi[6].

2.3 Berkas MIDI

Standard MIDI File (SMF) adalah format berkas yang digunakan untuk menampung data MIDI (*Musical Instrument Digital Interface*) ditambah beberapa jenis data yang biasa dibutuhkan oleh *sequencer* (pemutar berkas MIDI). Format ini menampung standard MIDI *message* ditambah dengan *time-stamp* untuk setiap *message*. Format ini juga membolehkan menyimpan informasi mengenai tempo, waktu dan kunci, *signature*, nama *track* dan *pattern*, dan informasi lainnya yang biasa dibutuhkan oleh sequencer. Satu SMF dapat menyimpan informasi untuk beberapa *pattern* dan *track*, sehingga setiap *sequencer* yang berbeda dapat mendukung struktur ini ketika membuka berkas tersebut.