BAB II
KAJIAN PUSTAKA

A. Matriks

Matriks adalah susunan berbentuk persegi panjang dari bilangan-bilangan yang diatur dalam baris dan kolom (Hadley, 1992). Bilangan-bilangan di dalam susunan tersebut dinamakan entri dalam matriks. Secara umum matriks ditulis sebagai berikut :

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \qquad (2.1)$$

Susunan di atas disebut sebuah matriks *m* kali *n* (ditulis *m* x *n*) karena memiliki *m* baris dan *n* kolom. Setiap matriks yang banyak baris dan kolomnya sama disebut matriks persegi. Sebuah matriks dengan *n* baris dan *n* kolom disebut matriks persegi berordo *n* (Hadley, 1992).

Matriks mempunyai operasi yaitu (Hadley, 1992):

1. Kesamaan

   Dua matriks *A* dan *B* disebut sama jika kedua elemen-elemen matriks tersebut sama $(a_{ij} = b_{ij}, untuk\ setiap\ i\ dan\ j)$ atau dilambangkan sebagai $A = B$, jika ada salah satu elemen dari matriks tersebut yang tidak sama, maka $A \neq B$.

2. Perkalian dengan skalar

Terdapat sebuah matriks *A* dan sebuah skalar γ, hasil perkalian γ dan *A*, ditulis γA, didefinisikan sebagai

$$\gamma A = \begin{bmatrix} \gamma a_{12} & \cdots & \gamma a_{1n} \\ \gamma a_{21} & \cdots & \gamma a_{2n} \\ \vdots & \ddots & \vdots \\ \gamma a_{m1} & \cdots & \gamma a_{mn} \end{bmatrix} \tag{2.2}$$

3. Penjumlahan

Matriks *C* yang didapatkan dari penjumlahan matriks *A* yang mempunyai *m* baris dan *n* kolom dan matriks *B* yang mempunyai *m* baris dan *n* kolom merupakan sebuah matriks yang mempunyai *m* baris dan *n* kolom dimana elemen-elemennya diberikan oleh $c_{ij} = a_{ij} + b_{ij}$, di definisikan sebagai

$$C = \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mn} \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} + \begin{bmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{m1} & \cdots & b_{mn} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11} + b_{11} & \cdots & a_{1n} + b_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} + b_{m1} & \cdots & a_{mn} + b_{mn} \end{bmatrix} \tag{2.3}$$

Operasi pengurangan didefinisikan sebagai :

$$A - B = A + (-1)B. \tag{2.4}$$

4. Perkalian

Jika matriks *A* berukuran *m* x *n* dan matriks *B* berukuran *n* x *r*, maka hasil kali matriks *AB* didefinisikan sebagai matriks *C* berukuran *m* x *r*, yang elemen-elemennya dihitung dari elemen-elemen dari *A*,*B* menurut

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj},\ i = 1, \dots, m,\ j = 1, \dots, r. \qquad (2.5)$$

Perkalian matriks *AB* bisa dilakukan jika dan hanya jika jumlah kolom matriks *A* sama dengan jumlah baris matriks *B*.

Selain ada operasi matriks, aturan-aturan ilmu hitung matriks harus diperhatikan dalam melakukan operasi matriks, berikut aturan-aturan hitung matriks, dengan *A,B,C* adalah matriks, *a,b* adalah sembarang konstanta dan *r,s* adalah bilangan bulat (Anton,2014) :

a) $A + B = B + A$
b) $A + (B + C) = (A + B) + C$
c) $A(BC) = (AB)C$
d) $A(B + C) = AB + AC$
e) $(B + C)A = BA + CA$
f) $a(B + C) = aB + aC$
g) $(a + b)C = aC + bC$
h) $(ab)C = a(bC)$
i) $a(BC) = (aB)C = B(aC)$
j) $A^r + A^s = A^{r+s}$
k) $(A^r)^s = A^{rs}$

B. Grup

(*G*,*) yang berisikan himpunan tak kosong *G* dan operasi * disebut grup jika memenuhi aksioma-aksioma berikut (Sukirman: 2005):

1. Operasi * bersifat tertutup di *G*

   $\forall\ a, b\ \in G$ maka $a * b\ \in G$

2. Operasi * bersifat asosiatif di *G*

$(a * b) * c = a * (b * c), \forall\, a, b, c \in G.$

3. *G* memuat elemen identitas terhadap operasi *,misal *e.*

   $e * a = a * e = a, \forall a \in G$

4. Setiap unsur di G mempunyai invers terhadap operasi *, untuk setiap $a \in G$ terdapat $a^{-1} \in G$ sedemikian sehingga $a^{-1} * a = a * a^{-1} = e$.

Suatu himpunan tak kosong disebut semi grup jika memenuhi aksioma 1 dan 2.

Contoh 2.1:

Misalkan $M_2(\mathbb{R})$ adalah himpunan matriks 2 x 2 dan determinan tidak sama dengan nol dengan entri bilangan rill,maka ($M_2(\mathbb{R})$, $\times$) adalah grup karena berlaku :

1. $\forall\, A, B \in M_2(\mathbb{R})$, maka $AB \in M_2(\mathbb{R})$. (tertutup)

   Karena det(A) ≠ 0 dan det(B) ≠ 0 maka det(AB) ≠ 0

2. $\forall\, A, B \in M_2(\mathbb{R})$, menurut sifat matriks yang c, maka

   $(AB)C = A(BC)$. (asosiatif)

3. Ambil $I \in M_2(\mathbb{R})$, sehingga $AI = IA = A, \forall\, I \in M_2(\mathbb{R})$. (mempunyai identitas yaitu $I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$)

4. $\forall\, A \in M_2(\mathbb{R})$, terdapat $A^{-1} \in M_2(\mathbb{R})$. Sehingga $AA^{-1} = A^{-1}A = 0$ (mempunyai invers yaitu $A^{-1} = \frac{1}{\det(A)} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$)

C. Kekongruenan

Definisi 2.1(Burton, 2007) :

Misalkan n suatu bilangan bulat positif, bilangan bulat a kongruen dengan b modulo n ditulis a ≡ b (mod n), bila n membagi (a-b) yaitu jika a – b = kn untuk suatu bilangan bulat k.

Contoh 2.2 :

22 ≡ 2 (mod 10). Terdapat bilangan bulat *k* sehingga 22 – 2 = 10*k*.

13 ≡ 6 (mod 7). Terdapat bilangan bulat *k* sehingga 13 – 6 = 7*k*.

Relasi kekongruenan mempunyai sifat, yaitu (Burton, 2007):

1) $a \equiv a \pmod{n}$
2) Jika $a \equiv b \pmod{n}$ maka $b \equiv a \pmod{n}$
3) Jika $a \equiv b \pmod{n}$ dan $b \equiv c \pmod{n}$ maka $a \equiv c \pmod{n}$
4) Jika $a \equiv b \pmod{n}$ dan $c \equiv d \pmod{n}$ maka $a + c \equiv b + d \pmod{n}$ dan $ac \equiv bd \pmod{n}$
5) Jika $a \equiv b \pmod{n}$ maka $a + c \equiv b + c \pmod{n}$ dan $ac \equiv bc \pmod{n}$
6) Jika $a \equiv b \pmod{n}$ maka $a^k \equiv b^k \pmod{n}$, dengan $k \in B^+$ (B^+ adalah bilangan bulat positif).

Bukti:

1) Untuk setiap bilangan bulat a diketahui bahwa $a - a = 0 \times n$, sedemikian sehingga $a \equiv a \pmod{n}$.
2) Jika $a \equiv b \pmod{n}$ maka, $a - b = k \times n$, untuk setiap bilangan bulat k. Karena itu, $b - a = (-k) \times n$, untuk setiap bilangan bulat $(-k)$. Sehingga dapat ditulis $b \equiv a \pmod{n}$.
3) Jika $a \equiv b \pmod{n}$ maka, $a - b = k \times n$, untuk setiap bilangan bulat k. Jika $b \equiv c \pmod{n}$ maka, $b - c = l \times n$, untuk setiap bilangan bulat l.

$$\begin{array}{l} a - b = k \times n \\ b - c = l \times n \quad + \\ \hline a - c = (k + l)n \end{array}$$

$a - c = (k + l)n$ dapat ditulis $a \equiv c \pmod{n}$, untuk setiap bilangan bulat $(k + l)$.

4) Jika $a \equiv b \pmod{n}$ maka, $a - b = k \times n$, untuk setiap bilangan bulat k. Jika $c \equiv d \pmod{n}$ maka, $c - d = l \times n$, untuk setiap bilangan bulat l.

$$\begin{array}{l} a - b = k \times n \\ c - d = l \times n \quad + \\ \hline a + c - (b + d) = (k + l)n \end{array}$$

$a + c - (b + d) = (k + l)n$ dapat ditulis $a + c \equiv b + d \pmod{n}$, untuk setiap bilangan bulat $(k + l)$.

$$\begin{array}{l} a = b + (k \times n) \\ c = d + (l \times n) \quad \times \\ \hline ac = bd + (kln + dk + bl)n \end{array}$$

$ac = bd + (kln + dk + bl)n$ <=> $ac - bd = (kln + dk + bl)n$,

dapat ditulis $ac \equiv bd (\mathrm{mod}\, n)$, untuk setiap bilangan bulat $(kln + dk + bl)$.

5) Jika $a \equiv b$ (mod n) maka, $a - b = k \times n$, untuk setiap bilangan bulat k.

$$a - b + (c - c) = k \times n <=> a + c - (b + c) = k \times n$$

$a + c - (b + c) = k \times n$, dapat ditulis $a + c \equiv b + c$ (mod n), untuk setiap bilangan bulat k.

$$(a - b)\frac{c}{c} = k \times n <=> ac - bc = kc \times n$$

$ac - bc = kc \times n$, dapat ditulis $ac \equiv bc (\mathrm{mod}\, n)$, untuk setiap bilangan bulat kc.

6) Dibuktikan benar untuk $k = 1$

$a \equiv b$ (mod n) (diketahui)

diasumsikan benar untuk $k = p$

$a^p \equiv b^p$(mod n)

akan dibuktikan benar untuk $k = p + 1$

dari sifat 4, jika $a \equiv b$ (mod n) dan $a^p \equiv b^p$(mod n) maka $aa^p \equiv bb^p$(mod n) *equivalent* dengan $a^{p+1} \equiv b^{p+1}$(mod n) (terbukti).

Dengan demikian terbukti bahwa jika $a \equiv b$ (mod n) maka $a^k \equiv b^k$), dengan $k \in B^+$ (B^+ adalah bilangan bulat positif).

D. Kriptografi

Kriptografi merupakan ilmu sekaligus seni dalam mengamankan pesan. Dalam buku pengantar ilmu kriptografi karya Dony Ariyus (2007) dijelaskan bahwa kriptografi muncul lebih dari 4000 tahun yang lalu yang diperkenalkan oleh orang mesir. Dahulu kriptografi digunakan dalam dunia militer untuk mengirimkan pesan rahasia ke medan perang.

1. Definisi Kriptografi

Kriptografi adalah ilmu yang mempelajari prinsip – prinsip dan teknik – teknik sedemikian sehingga dapat menyembunyikan suatu informasi atau pesan menjadi sebuah teks tersembunyi atau *ciphertexts* dan kemudian *ciphertext* tersebut diubah menjadi pesan asli *plaintext* oleh pengguna resmi dengan menggunakan kunci rahasia atau *secret key* (Yan, 2002). Kriptografi merupakan cara untuk mengamankan pesan dan berperan penting dalam pengiriman informasi atau pesan yang bersifat rahasia. Proses penyandian dilakukan dengan aturan tertentu sehingga hanya dapat dibaca oleh yang berhak.

2. Tujuan Kriptografi

Tujuan dari kriptografi menurut (Menezes, Oorscoot dan Vanstone, 1996) :

a. Kerahasiaan (*confidentiality*) adalah layanan yang digunakan untuk menjaga segala bentuk isi informasi tetapi bisa diketahui oleh yang berhak. Rahasia adalah istilah lain yang memiliki arti sama dengan *confidentiality* dan *privacy*. Ada banyak pendekatan untuk memberikan

kerahasiaan, mulai dari keamanan fisik sampai algoritma matematika yang mana diberikan agar data sulit dipahami.

b. Integritas Data(*Data Integrity*) adalah layanan penjagaan perubahan data oleh pihak yang tidak ber hak. Untuk meyakinkan integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi data yang dilakukan oleh pihak yang tidak ber hak. Termasuk hal-hal manipulasi data seperti sisipan, penghapusan dan pergantian.

c. Otentikasi (*Authentication*) adalah layanan yang berhubungan dengan identifikasi. Fungsi ini berlaku untuk sistem maupun informasi itu sendiri. Dua pihak yang saling berkomunikasi harus mengidentifikasi satu sama lain. Informasi dikirim melalui saluran harus di identifikasi keasliannya, isi data, waktu pengiriman dan lainnya. Aspek dari kriptografi dibagi menjadi dua kelas utama : *entity authentication* dan *data origin authentication.*

d. Nirpenyangkalan (*non-repudiation*) adalah layanan yang mencegah terjadinya penyangkalan pengiriman suatu informasi oleh yang mengirimkan maupun yang menerima.

3. Terminologi Kriptografi

Di dalam kriptografi akan sering ditemukan berbagai istilah atau terminologi. Beberapa istilah penting yang sering digunakan dalam kriptografi yaitu:

a. Pesan, *Plaintext,* dan *Ciphertext.*

Pesan adalah data atau informasi yang dapat dibaca dan dimengerti maknanya. Nama lain untuk pesan adalah *plaintext*, atau takjelas (*cleartext*) (Munir, 2006). Pesan dalam bentuk umum dapat dengan mudah dimengerti oleh orang lain. Oleh karena itu, agar pesan yang dikirimkan tidak diketahui oleh orang yang tidak ber hak, pesan perlu disandikan ke bentuk yang tidak dipahami. Bentuk pesan yang tersandi disebut *ciphertext*.

Contoh 2.3 (contoh *plaintext*)

Ayah sedang pergi ke bandung.

Contoh 2.4 (contoh *ciphertext*)

Ecel wiherk tivkm oi ferhyrk.

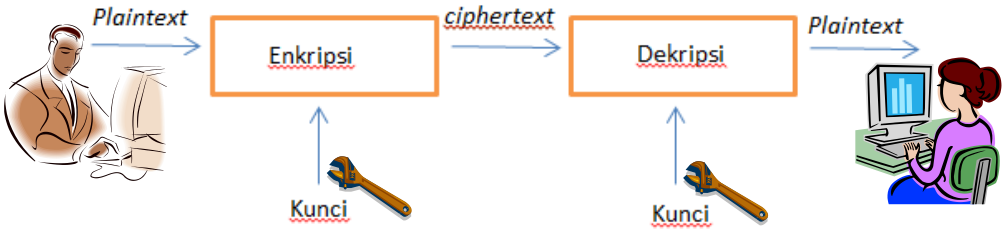
b. Pengirim dan Penerima

Pengirim dan penerima merupakan objek dalam aktifitas komunikasi yang mengakibatkan pertukaran pesan bisa dilakukan. Pengirim(*sender*) adalah entitas yang mengirim pesan kepada entitas lainnya. Penerima(*receiver*) adalah entitas yang menerima pesan (Munir, 2006). Entitas berupa manusia, mesin komputer, dan lainnya.

c. Enkripsi dan Dekripsi

Enkripsi adalah proses penyandian dengan merubah kode atau pesan yang dapat dibaca oleh semua pihak(*Plaintext*) menjadi kode atau pesan yang hanya dimengerti oleh beberapa pihak(*ciphertext*), sedangkan proses untuk mengubah *ciphertext* menjadi *plaintext* disebut dekripsi.

Proses enkripsi dapat menyandikan pesan yang bersifat rahasia, dengan proses enkripsi, pesan yang mudah untuk diketahui bisa menjadi pesan yang sulit untuk di ketahui. Harus mempunyai teknik khusus untuk mendekripsikan pesan yang sudah dienkripsi. Jika tidak mengetahui kunci yang digunakan, maka akan sulit untuk mengetahui pesan yang sudah dienkripsikan. Skema enkripsi dan dekripsi terdapat pada gambar 2.1.



Gambar 2.1 Skema Enkripsi dan Dekripsi

d. Algoritma dan Kunci

Algoritma kriptografi disebut juga *cipher* yaitu fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Konsep matematis yang mendasari algoritma kriptografi adalah relasi antara dua buah himpunan yang berisi elemen - elemen *plaintext* (Munir,2006). Proses dekripsi dan enkripsi membutuhkan kunci. Kunci adalah parameter yang digunakan untuk transformasi enkripsi dan dekripsi (Munir, 2006). Semakin sulit kunci untuk ditebak, akan semakin sulit juga pesan rahasia untuk dipecahkan oleh orang yang tidak ber hak. Biasanya, kunci berupa deretan bilangan maupun string.

Kunci untuk proses enkripsi dan dekripsi terbagi menjadi dua, yaitu algoritma simetri (*secret key algorithm*) dan algoritma asimetri (*public key algorithm*). Pada algoritma simetri kunci yang digunakan dalam proses enkripsi dan dekripsi adalah sama, sehingga kedua belah pihak harus sepakat menentukan kunci yang akan dipilih dan agar pesan yang dikirim aman, kunci harus dirahasiakan. Tingkat keamanan algoritma simetri tergantung pada kerahasiaan kunci yang digunakan. Algoritma asimetri, kunci yang digunakan dalam proses enkripsi dan dekripsi akan berbeda. Pada algoritma asimetri, kunci enkripsi disebut kunci publik (*public key*), kunci ini dapat diketahui oleh setiap orang, dan kunci dekripsi disebut dengan kunci rahasia (*private key*), kunci ini hanya diketahui oleh penerima pesan (Munir,2006).

Ada dua jenis algoritma kriptografi simetri, yaitu *block cipher* dan *stream cipher* (Dwi,2012). Algoritma kriptografi *block cipher*, proses enkripsi dekripsi dilakukan dengan memotong plaintext menjadi blok-blok. Sedangkan algoritma kriptografi *stream cipher*, proses enkripsi dekripsi dilakukan secara mengalir dan dengan kunci yang mengalir juga (Dwi,2012). Algoritma kriptografi *stream cipher* memiliki ke unggulan yaitu tidak membatasi panjang *plaintext*. Algoritma ini cocok untuk digunakan dalam enkripsi suatu komunikasi yang berlangsung dan berkelanjutan, seperti komunikasi melalui telepon.

Vigenere cipher atas grup $\mathbb{Z}_{26} = \{0,1,2,\dots,25\}$ merupakan salah satu contoh kriptografi yang menggunakan algoritma *stream cipher*.

Algoritma *Vigenere cipher* atas grup $\mathbb{Z}_{26}$ akan dijelaskan sebagai berikut (Dwi, 2012):

Diberikan *plaintext* $x_1, x_2, x_3, \ldots$ dengan $x_i \in \mathbb{Z}_{26}$ dan kunci $k_1, k_2, k_3, \ldots$ dengan $k_i \in \mathbb{Z}_{26}$. *Ciphertext* $y_1, y_2, y_3, \ldots$ diperoleh proses enkripsi sebagai berikut

$$y_1 = x_1 + k_1, y_2 = x_2 + k_2, \ldots, y_n = x_n + k_n, \ldots (\text{mod } 26).$$

Dalam penelitian ini akan menggunakan *Vigenere cipher* modulo 94 untuk melakukan proses enkripsi dan dekripsi.

e. Sistem kriptografi

Sistem kriptografi terdiri dari 5 tupel (*P,C,K,* ε *, D*) yang memenuhi syarat sebagai berikut (Stinson, 2006) :

1) *P* adalah himpunan berhingga semua *plaintext*;
2) *C* adalah himpunan berhingga semua *ciphertext*;
3) Ruang kunci *K* adalah himpunan berhingga semua kunci;
4) Untuk setiap $k \in K$, terdapat aturan enkripsi $e_k \in \varepsilon$ dan korespondensi aturan deskripsi $d_k \in D$. Untuk setiap $e_k : P \rightarrow C$ dan $d_k : C \rightarrow P$, adalah fungsi sedemikian sehingga $d_k(e_k(x)) = x$, untuk setiap *plaintext* x $\in P$.

Syarat yang paling penting adalah syarat ke-4, yang berarti jika sebuah *plaintext x* dienkripsi menggunakan e_k dan menghasilkan *ciphertext*, kemudian didekripsikan menggunakan d_k maka diperoleh *plaintext x* kembali.

f. Penyadap

Penyadap adalah orang yang berusaha menangkap pesan selama ditransmisikan dengan tujuan mendapatkan informasi sebanyak-banyaknya mengenai sistem kriptografi yang digunakan untuk berkomunikasi dengan maksud memecahkan *ciphertext*. Nama lain dari penyadap adalah *enemy*, *bad guy*, *intruder* dan lainnya (Munir,2006).

g. Kriptoanalisis

Kriptoanalisis berasal dari bahasa yunani yang terdiri dari dua suku kata yaitu *kriptos* dan *analyein. Kriptos* artinya menyembunyikan sedangkan *analyein* artinya memecahkan, sehingga kriptoanalisis adalah ilmu dan seni untuk memecahkan *ciphertext* menjadi *plaintext* tanpa mengetahui kunci yang digunakan. Pelakunya disebut kriptoanalis (Munir, 2006).

E. Teori Chaos

Chaos merupakan hal yang berhubungan dengan kekacauan, keacakan, ketidakberaturan. Teori chaos merupakan cabang dari matematika yang mempelajari bagaimana membangkitkan bilangan secara acak (Alvin:tanpa tahun). Meskipun chaos menunjukan keacakan, sistem chaotic dapat ditentukan secara matematis.

Dalam buku Teori Chaos karya Yani Kusmarni (2008) dijelaskan pada tahun 1880, seorang ahli matematika perancis bernama Henri Poincare adalah orang yang pertama yang merumuskan yang sekarang dikenal sebagai chaos. Henri Poincare menemukan bahwa terdapat orbit yang bersifat nonperiodik,

yang berarti tidak bekerja secara teratur. Awalnya gagasan Henri tidak terlalu dihargai oleh para ilmuan, namun pada tahun 1898 Jacques Hadamard mempublikasikan teori yang hampir sama dengan Teori Henri.

Teori Chaos pertama kali dicetuskan oleh Edward Lorenz yang merupakan seorang ahli metereologi AS pada tahun 1961. Edward menemukan teori chaos secara tidak sengaja saat melakukan peramalan cuaca dengan bantuan komputer. Dalam melakukan percobaan Edward hanya mengambil 3 digit dibelakang koma dari hasil yang didapatkan pada saat melakukan peramalan cuaca untuk dijadikan kondisi awal yang baru. Hasil yang didapatkan sangat jauh berbeda. Data dari hasil percobaan dimasukkan dalam bentuk grafik maka terciptalah yang biasa disebut efek kupu-kupu.

Teori Chaos berguna dalam membangkitkan bilangan secara acak, dan juga peka terhadap kondisi awal, perubahan nilai awal sekecil 10^{-100} akan membangkitkan bilangan yang benar-benar berbeda. Hal ini sangat berguna dan dapat diterapkan di dalam kriptografi sebagai pembangkit kunci, yang selanjutnya digunakan untuk proses enkripsi serta dekripsi, maka dari itu dipilih Teori Chaos untuk diterapkan dalam kriptografi.

1. Definisi Teori Chaos

Chaos adalah tingkah laku yang sangat kompleks, *irregular* dan *random* di dalam sebuah sistem yang deterministik (Steward, 1994). Suatu keadaan di mana sebuah sistem tidak dapat diprediksi di mana akan ditemukan di tempat berikutnya disebut chaos. Chaos bersifat acak akan tetapi, apabila keadaan lebih diperhatikan dalam jangka waktu yang cukup lama maka akan menemukan keteraturan dari chaos itu sendiri.

2. Jenis Fungsi dalam Teori Chaos

Teori Chaos memiliki banyak fungsi, beberapa fungsi yang umum dalam teori chaos yaitu (Alvin:tanpa tahun):

a. Logictic Map

Fungsi logistic merupakan fungsi chaos. Bentuk fungsi logistic:

$$f(x) = rx(1-x) \qquad (2.8)$$

Dengan

$r =$ Sembarang bilangan rill ($0 \leq r \leq 4$).

Selanjutnya, dalam sistem kriptografi fungsi logistic digunakan secara iteratif sehingga menghasilkan persamaan (2.9)

Persamaan logistik merupakan contoh pemetaan polinomial derajat dua. Persamaan ini dipublikasikan pada tahun 1976 oleh seorang ahli biologi yang bernama Robert May setelah melanjutkan persamaan logistik yang dikembangkan oleh Pierre Francois. Secara matematis, persamaan logistic dinyatakan dengan persamaan :

$$x_{i+1} = rx_i(1 - x_i). \quad (2.9)$$

b. Henon Map

Fungsi Henon (Henon Map) disajikan secara iteratif dalam persamaan (2.10) dan (2.11). Henon Map adalah persamaan sistem dinamis yang menerapkan sistem diskrit. Michel Henon adalah orang yang memperkenalkan persamaan henon sebagai model sederhana dari Poincare section Lorenz model. Persamaan ini menggunakan sebuah titik (*x*,*y*) dan sangat bergantung pada dua parameter. Persamaan henon yaitu :

$$x_{n+1} = y_n + 1 - ax_n^2 \quad (2.10)$$

$$y_{n+1} = bx_n. \quad (2.11)$$

Dengan

a,*b* = sembarang bilangan rill

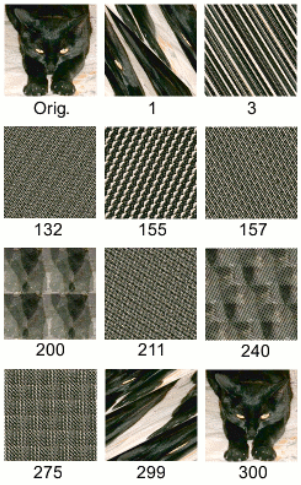
c. Arnold's Cat Map

Arnold's Cat Map ditemukan oleh Vladimir Arnold pada tahun 1960. Ketika melakukan penelitian, Arnold menggunakan sebuah gambar kucing dalam melakukan percobaan, sehingga algoritma ini dinamakan dengan Arnold's Cat Map(ACM). ACM mentransformasikan koordinat (*x*,*y*). Fungsi ACM dituliskan sebagai berikut:

$$f(x,y) = \begin{cases} \left(2x, \frac{y}{2}\right), & 0 \le x \le 0.5, 0 \le y \le 1 \\ \left(2x - 1, \frac{y+1}{2}\right), & 0.5 \le x \le 1, 0 \le y \le 1 \end{cases} \quad .(2.12)$$

Fungsi ACM memiliki ke unggulan yaitu kecepatan dalam mengenkripsikan data. Tetapi, fungsi ACM mempunyai periode, yaitu jika mencapai iterasi tertentu, hasil dari fungsi ini akan kembali ke awal.

Contoh dari periode akan di tampilkan pada gambar 2.2, dimana informasi yang akan di sandikan berupa gambar berukuran 150 x 150 *pixels*. Gambar tersebut akan kembali ke gambar asli setelah di lakukan iterasi yang ke 300.



Gambar 2.2 Proses enkripsi gambar menggunakan Arnold's Cat Map

d. Duffing Map

Duffing Map merupakan salah satu sistem dinamis dengan menggunakan waktu diskrit yang menerapkan sifat chaos. Duffing map mengambil sebuah titik pada sebuah kordinat lalu memetakan menjadi sebuah titik yang baru. Pemetaan dipengaruhi oleh dua buah nilai yang merupakan konstanta. Duffing Map dituliskan sebagai berikut :

$$x_{n+1} = y_n \tag{2.13}$$

$$y_{n+1} = -bx_n + ay_n - y_n^3. \tag{2.14}$$

Dengan

a,*b* = sembarang bilangan rill

Dalam penelitian ini dipilih Fungsi Arnold's Cat Map dalam Membangkitkan kunci, karena terdapat dua parameter *x* dan *y* yang membuat kunci makin sulit ditebak oleh pihak 3.

F. Protokol Perjanjian Kunci

Protokol perjanjian kunci merupakan skema dalam kriptografi yang digunakan untuk mengatasi masalah perjanjian kunci rahasia. Dalam kriptografi simetris, kunci yang digunakan harus benar-benar dirahasiakan agar pihak 3 tidak bisa mengetahui kunci. Kunci berguna dalam proses enkripsi dan dekripsi diantara dua pihak yang saling bertukar pesan, oleh karena itu kunci harus sangat dirahasiakan agar pesan asli tidak dapat diketahui oleh pihak 3.

Tingkat keamanan dari protokol perjanjian kunci terdapat pada tingkat kesulitan dari suatu permasalahan matematis dan bertujuan agar kedua belah pihak dapat menyepakati kunci yang sama tanpa bisa diketahui oleh pihak 3. Ada beberapa protokol perjanjian kunci, salah satunya protokol kunci Diffie-Hiellman yang merupakan contoh perjanjian kunci yang paling sederhana. Perjanjian kunci ini di keluarkan pada tahun 1976, skema protokol perjanjian kunci Diffie-Hiellman dapat dilihat pada tabel 1.

Tabel 1 Skema Protokol Perjanjian Kunci Diffie-Hiellman (Myasnikov dkk,2008)

Pihak 1 atau pihak 2 mempublikasikan suatu grup siklik *G* dengan elemen pembangkit $g \in G$.	
Pihak 1	Pihak 2
1) Memilih secara rahasia suatu bilangan positif *a* 2) Menghitung g^a 3) Mengirim g^a kepada pihak 2 4) Menerima g^b dari pihak 2 5) Menghitung $K_1 = (g^a)^b = g^{ab}$	1) Memilih secara rahasia suatu bilangan positif *b* 2) Menghitung g^b 3) Mengirim g^b kepada pihak 1 4) Menerima g^a dari pihak 1 5) Menghitung $K_2 = (g^b)^a = g^{ba}$
Pihak 1 dan pihak 2 telah menyepakati kunci rahasia *K*= $K_1 = K_2$	

Setiap grup siklik merupakan grup komutatif, maka *ab* = *ba*, sehingga diperoleh *K*= $K_1 = K_2$. Tingkat keamanan dari protokol perjanjian kunci Diffie-Hiellman didasarkan pada masalah logaritma diskrit pada grup siklik. Protokol perjanjian kunci yang lainnya adalah protokol perjanjian kunci stickel. Berbeda dengan perjanjian kunci sebelumnya yang menggunakan grup siklik, perjanjian kunci stickel menggunakan grup non komutatif, skema protokol perjanjian kunci stickel dapat dilihat pada tabel 2.

Tabel 2 Skema Protokol Perjanjian Kunci Stickel (Myasnikov dkk, 2008)

Pihak 1 atau pihak 2 mempublikasikan suatu grup non-komutatif G dan $a,b \in G, ab \neq ba$, dengan N order dari a dan M order dari b.	
Pihak 1	Pihak 2
1) Memilih secara rahasia $n < N, m < M$ & $n,m \in \mathbb{N}$ 2) Menghitung $U = a^n b^m$ 3) Mengirim U Kepada pihak 2 4) Menerima V dari pihak 2 5) Menghitung $K_1 = a^n V b^m$	1) Memilih secara rahasia $r < N$, $s < M$ & $r,s \in \mathbb{N}$ 2) Menghitung $V = a^r b^s$ 3) Mengirim V Kepada pihak 1 4) Menerima U dari pihak 1 5) Menghitung $K_2 = a^r U b^s$
Pihak 1 dan pihak 2 berhasil menyepakati nilai awal yang sama $K = K_1 = K_2$	

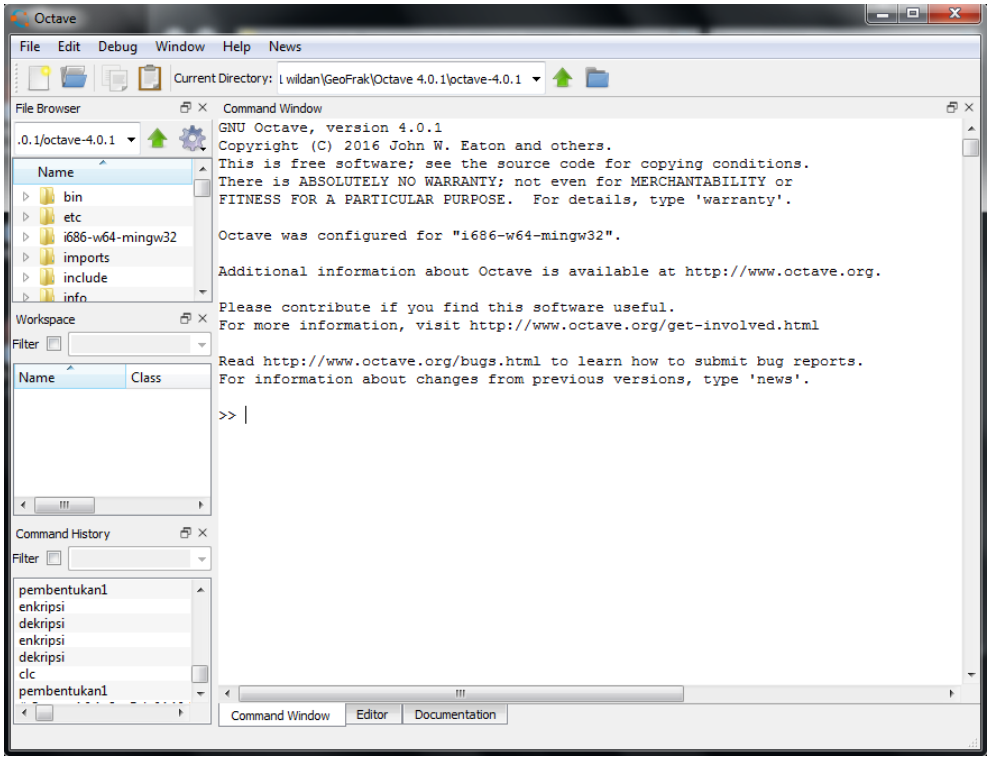
Dalam tabel di atas pihak 1 dan pihak 2 berhasil menyepakati kunci yang sama yaitu

$$K_1 = a^n V b^m = a^n a^r b^s b^m = a^{n+r} b^{m+s} = a^r a^n b^m b^s = a^r U b^s = K_2.$$

Dipilih perjanjian kunci stickel untuk menentukan pembangkit kunci karena bisa menggunakan matriks non komutatif dan sebarang bilangan asli untuk perpangkatan matriks yang hanya diketahui oleh dirinya sendiri yang membuat pihak 3 sulit untuk menebak kunci.

G. Octave

Octave merupakan suatu perangkat lunak gratis dan bahasa tingkat tinggi untuk komputasi numerik dan visualisasi data. Octave dikembangkan oleh John W. Eaton dan sekarang pengembangan serta pemeliharaan *Octave* dilakukan oleh beberapa orang *volunteer* dari berbagai penjuru dunia (Arief,2008).



Gambar 2.3 Tampilan Awal *Octave*

1. Tools

Terdapat beberapa tools utama dalam *octave*, yaitu :

a. *Command Window*

Comman window digunakan untuk memasukan variabel dan menjalankan fungsi atau biasa disebut script. Perintah yang ditulis di *command window* langsung ditampilkan, dan jika perintah salah maka akan muncul pesan eror dalam *command window*.

b. *Command History*

Perintah yang dijalankan dalam *command window* akan tersimpan semuanya di *command history*. *Command history* membantu dalam melihat perintah sebelumnya atau juga dapat mengulang perintah yang telah dilakukan.

c. *Text Editor*

Text editor berguna untuk membuat M-file atau *script*. Hampir sama dengan *command window*, tetapi *text editor* tidak langsung memunculkan hasil dari perintah. Perintah dalam *text editor* harus dipanggil dalam *command window* agar hasilnya keluar.

2. Operasi

Octave memiliki beberapa operasi aritmatika, beberapa operasi yang sering digunakan dalam *Octave* akan ditampilkan dalam tabel berikut:

Tabel 3: Operasi Aritmatika dalam Octave

Simbol	Kegunaan
+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
^	Perpangkatan

Selain operasi aritmatika, *octave* juga memiliki operasi perbandingan. Beberapa operasi perbandingan akan ditampilkan dalam tabel berikut:

Tabel 4 : Operasi Perbandingan dalam Octave

Simbol	Dekripsi
<	Kurang dari
<=	Kurang dari sama dengan
==	Sama dengan
>	Lebih dari
>=	Lebih dari sama dengan
~=	Tidak sama dengan

3. Pernyataan

Octave menyediakan beberapa struktur-struktur pernyataan. Beberapa pernyataan yang sering digunakan dalam *Octave*, yaitu pernyataan *if*, pernyataan *switch*, pernyataan *while*, dan pernyataan *fo*r. Adapun penjelasan dari beberapa pernyataan, yaitu (Eaton,2007):

a. Pernyataan *if*

Pernyataan ini mempunyai bentuk *if-else-end*. Jika kondisi *if* terpenuhi, maka akan penjalankan perintah yang ada di dalam *if* tersebut. Jika tidak maka akan menjalankan perintah selanjutnya atau *else*. Ada beberapa bentuk dalam pernyataan *if*, yaitu :

```
If (kondisi)
  Perintah
Else
  Perintah
End
```

```
If (kondisi1)
  Perintah
Elseif (kondisi2)
  Perintah
Elseif...
  .
  .
  .
Else
  Perintah jika semua kondisi tidak terpenuhi
End.
```

b. Pernyataan *switch*

Switch (kondisi1)
Case (kondisi2)
Perintah
Case (kondisi3)
Perintah
Case ...
.
.
.
Otherwise
Perintah
End.

Penyataan ini membandingkan antara kondisi1 dan setelahnya, jika kondisi1 sama dengan kondisi2 maka perintah yang ada di dalam kondisi2 akan dijalankan, dan seterusnya. Jika semua kondisi tidak terpenuhi maka akan dijalankan perintah pada *otherwise*.

c. Pernyataan *for*

For (x = array)
Perintah
End.

Pernyataan ini menjalankan perintah untuk *x* sama dengan *array* secara berulang-ulang. Pernyataan ini cocok untuk perhitungan iterasi dan sejenisnya.

d. Pernyataan *while*

Sedikit berbeda dengan pernyataan *for* yang melakukan perulangan dengan jumlah ditentukan, pernyataan *while* melakukan perulangan dengan jumlah yang tak terbatas.

Pernyataan *while* akan terus berulang selama elemen dalam kondisi terpenuhi, jika tidak terpenuhi maka pernyataan *while* akan berenti. Bentuk umumnya yaitu:

While (kondisi)
  Perintah
End.