

11. KEGIATAN BELAJAR 11 : PRASENTASION LAYER

1. Tujuan Pembelajaran

- Setelah mengikuti kegiatan belajar 11 ini siswa diharapkan dapat :
- 1) Memahami Presentation layer pada Jaringan Komputer
 - 2) Menganalisis Presentation layer pada Jaringan Komputer

2. Uraian Materi

PRESENTATION LAYER

Presentation layer merupakan lapisan ke-enam dari model referensi OSI. *Presentation layer* melakukan fungsi-fungsi tertentu yang diminta untuk menjamin pene-muan sebuah penyelesaian umum bagi masalah. Tidak seperti *layer-layer* di bawahnya yang hanya melakukan pemindahan bit dari satu tempat ke tempat lainnya, *presentation layer* memperhatikan sintaks dan semantik informasi yang dikirimkan.

Satu contoh layanan presentasi adalah *encoding data*. Kebanyakan pengguna tidak memindahkan string bit biner yang *random*. Para pengguna “saling bertukar data seperti nama orang, tanggal, jumlah uang dan agihan. Item-item tersebut dinyatakan dalam bentuk string karate, bilangan karkte, bilangan interger, bilangan *floating point*, struktur data yang dibentuk dari beberapa item yang lebih sederhana. Terdapat perbedaan antara satu komputer dengan komputer lainnya dalam memberi kode untuk menyatakan string karakter (misalnya ASCII dan UNICODE), interger (misalnya komplemen satu dan komplemen dua), dan sebagainya. Untuk memungkinkan dua buah komputer yang memiliki presentasi yangberbeda untuk dapat berkomunikasi, struktur data yang akan dipertukarkan dapat dinyatakan dengan cara abstrak, sesuai dengan *encoding standard*yang akan digunakan pada saluran. *Presentation layer*mengatur data-struktur abstrak ini dan mengkonversi dari

representation yang digunakan pada sebuah komputer menjadi *representation standard* jaringan, dan sebaliknya.

Lapisan ini berhubungan dengan sintaks data yang dipertukarkan diantara entitas aplikasi. Tujuannya adalah untuk mengatasi masalah perbedaan format penyajian data. Lapisan ini mendefinisikan sintaks yang digunakan antar entitas aplikasi.

1 Layanan Presentation Layer

Lapisan presentasi memberikan layanan pengelolaan pemasukan data, pertukaran data dan pengendalian struktur data. Implementasi utama dari lapisan presentasi adalah penyediaan fungsi yang standar dan umum.

Cara ini lebih efisien dibandingkan dengan pemecahan yang dilakukan sendiri oleh pemakai jaringan. Contoh dari protokol lapisan presentasi yang paling banyak dikenal dan dipakai orang adalah enkripsi data dan kriptografi.

1.2. Definisi Enkripsi

Enkripsi adalah sebuah proses yang melakukan perubahan sebuah kode dari yang bisa dimengerti menjadi sebuah kode yang tidak bisa dimengerti atau tidak bisa dibaca. Enkripsi dapat diartikan sebagai kode atau *chiper*. Sebuah *chiper* menggunakan suatu algoritma yang dapat mengkodekan semua aliran data dari sebuah pesan menjadi *cryptogram* yang tidak dimengerti. Karena teknik *chiper* merupakan suatu sistem yang telah siap untuk di automasi, maka teknik ini digunakan dalam sistem keamanan komputer dan jaringan.

Enkripsi dimaksudkan untuk melindungi informasi agar tidak terlihat oleh orang atau pihak yang bukan seharusnya. Enkripsi juga digunakan untuk verifikasi. Bila anda men-download software, misalnya, bagaimana anda tahu bahwa software yang anda download adalah yang asli, bukannya yang telah dipasangkan trojan di dalamnya.

Dalam hal ini terdapat tiga kategori enkripsi yaitu:

- 1 Kunci enkripsi rahasia, dalam hal ini terdapat sebuah kunci yang enkripsi dan juga sekaligus mendekripsi informasi.
- 2 Kunci enkripsi publik, dalam hal ini dua kunci digunakan, satu untuk proses enkripsi dan yang lain untuk proses dekripsi.
- 3 Fungsi *one-way*, atau fungsi 1 arah adalah suatu fungsi dimana informasi dienkripsi untuk menciptakan “*signature*” dari informasi asli yang bisa digunakan untuk keperluan autentifikasi.

Enkripsi dibentuk dengan berdasarkan suatu algoritma yang akan mengacak suatu informasi menjadi bentuk bentuk yang tidak bisa dibaca atau tak bisa dilihat. Deskripsi adalah proses dengan algoritma yang sama untuk mengembalikan informasi teracak menjadi bentuk aslinya. Algoritma yang digunakan harus terdiri dari susunan prosedur yang direncanakan secara hati-hati yang harus secara efektif menghasilkan sebuah bentuk terenkripsi yang tidak bisa dikembalikan oleh seseorang bahkan sekalipun mereka memiliki algoritma yang sama.

1.3. Model Enkripsi

Dalam membahas model-model enkripsi beserta algoritma yang akan dipakai untuk setiap enkripsi ada 2 hal yang penting yang akan dijabarkan yaitu enkripsi dengan kunci pribadi dan enkripsi dengan kunci publik.

1.4. Enkripsi dengan Kunci Pribadi

Enkripsi dapat dilakukan jika si pengirim dan si penerima telah sepakah untuk menggunakan metode enkripsi atau kunci enkripsi tertentu. Metode enkripsi atau kuncinya harus dijaga ketat supaya tidak ada pihak luar yang mengetahuinya. Masalahnya sekarang adalah bagaimana untuk memberitahu pihak penerima mengenai metode atau kunci yang akan kita pakai sebelum komunikasi yang aman bisa berlangsung. Kesepakatan cara enkripsi atau kunci dalam enkripsi ini bisa dicapai lewat jalur komunikasi lain yang lebih aman, misalnya dengan

bertemu langsung. Tetapi bagaimana jika jalur komunikasi yang lebih aman ini tidak memungkinkan. Yang jelas, kunci ini tidak bisa dikirim lewat jalur email biasa karena masalah keamanan.

Cara enkripsi dengan kesepakatan atau kunci enkripsi diatas dikenal dengan istilah inkripsi dengan kunci pribadi, karena cara enkripsi atau kunci yang hanya boleh diketahui oleh dua pribadi yang berkomunikasi tersebut. Cara enkripsi inilah yang umum digunakan pada saat ini baik untuk kalangan pemerintah mupun kalangan bisnis. Cara enkripsi ini juga di kategorikan sebagai kriptografi simetris, karena keduanya belah pihak mengetahui kunci yang sama. Selain masalah komunikasi awal untuk penyampaian kunci, cara enkripsi ini juga mempunyai kelemahan yang lain. Kelemahan ini timbul jika terdapat banyak orang yang ingin saling komunikasi. Karena setiap pasangan harus sepakat dengan kunci pribadi tertentu, tiap orang harus menghafal banyak kunci dan harus menggunakan secara tepat sebab jika tidak, maka si penerima tidak bisa mengartikannya.

Misalnya jika ada 3 orang, A, B, C saling berkomunikasi, pasangan A dan B harus sepakat dengan kunci tertentu yang tidak boleh diketahui oleh C, sehingga surat antara A dan B tidak bisa disadap oleh C. Demikia pula hal ini juga berlaku untuk pasangan B dan C atau pasangan A dan C. Jadi total ada 3 kunci yang beredar di kelompok tadi. Dengan kata lain, jika ada n orang lain saling berkomunikasi dengan cara enkripsi ini, total terdapat $n*(n-1)/2$ buah kunci yang beredar. Hal ini akan menimbulkan masalah dalam pengaturan sebuah kunci. Hal ini akan menimbulkan masalah dalam pengaturan sebuah kunci. Misalnya, kunci yang mana yang akan dipakai untuk mengirim ke A.

Ada beberapa model enkripsi yang termasuk dalam golongan ini, diantaranya adalah : Simple Substitution Cipher, DES, Triple DES, Rivest Code 4 (RC4), IDEA, skipjack, Caesar Cipher, Cost Block Cipher, Letter Map, Transposition Cipher, Blowfish, Enigma cipher.

1.5. Simple substitution cipher

Sebuah cipher adalah suatu metode untuk mengenkripsi sejumlah pesan yaitu mengubah pesan ke dalam suatu yang tidak mudah dibaca.

Pesan yang asli disebut plaintext. Substitution cipher adalah sebuah kondisi dimana masing-masing huruf dari sebuah plaintext diganti oleh simbol yang lain. Biasanya yang digunakan dalam penggantian simbol ini adalah huruf-huruf dari sederetan alfabet.

Sebuah alfabet adalah serangkaian urutan simbol-simbol. Sebagai contoh, secara normal alfabet Inggris terdiri dari simbol A sampai dengan Z dan hal ini digolongkan dalam rangkaian urutan simbol.

Substitusi sederhana adalah dimana dalam pesan simbol plaintext selalu diganti dengan simbol ciphertext yang sama. Dengan kata lain terjadi hubungan satu per satu diantara huruf-huruf dalam ciphertext maupun plaintext.

Sebagai contoh, amati pesan rahasia berikut ini :

E-----E-E---E—E—E-----E—

Satu masalah dalam hal ini adalah pola E- dan pola E—E. Karena ada 2 huruf kata bahasa Inggris yang melalui dengan E, maka hipotesa kita bahwa $T=E$ mungkin salah. Jenis pengetahuan yang lain, yang dapat kita gunakan untuk memecahkan cryptogram ini adalah bahwa 2 huruf yang paling sering muncul dalam bahasa Inggris adalah :

OF TO IN IS IT BE BYE BY HE AS ON AT OR AN SO IF NO

Karena ada kata-kata dalam 2 huruf ini yang terdapat dalam sebuah pesan dan diawali dan diakhiri dengan huruf K, barangkali hipotesa kita mungkin lebih baik apabila kita mengasumsikan jika $K=O$. Jika kita mencoba substitusi ini, kita akan mendapat hasil sebagai berikut :

-O-O—O—O----- ---O-

Karena kedua huruf yang paling sering muncul dalam alfabet Inggris adalah T, barangkali hipotesa kita berguna untuk yang lain, yaitu menjadi $T=T--$. Dengan kata lain, T ini berdiri sendiri. Dari hipotesa ini, kita akan memperoleh hasil sebagai berikut :

TO-O—OT-T-T-T----- ---T-O-

Dari hasil ini, kita bisa mulai melihat titik terang yang menjanjikan. Pada contoh diatas, T—kita bisa mengasumsikan bahwa paling umum 3 huruf dalam kata yang terdapat dalam alfabet inggris yang sering mulai dengan T adalah THE. Jika kita membuat tebakan bahwa B=H dan L=E, maka kita akan mendapatkan hasil sebagai berikut :

TO-EO—OT TO-E TH-T- THE – E-T-O-

Dari hasil ini mulai kelihatan lebih baik. Pola TH-T dapat kita tebak adalah THAT. Pola –OT kita tebak adalah NOT. Jika kita mengasumsika lagi bahwa S=A dan J=N, maka kita akan mendapatkan hasil sebagai berikut :

TO- EO- NOT TO-E THAT – THE –E-T- ON

Kata terakhir dalam pesan berakhir dengan pola T-ON, yang bisa kita tebak adalah TION. Dan jika kita membuat tebakan C=I, maka kita akan mendapat hasil sebagai berikut :

TO- E O- NOT TO- E THAT I- THE –E – TION

Dan sekarang nampak hasilnya dan kita sekarang mempunyai kata-kata seperti HAMLET pernah kemukakan yaitu :

TO BE OR NOT TO BE THAT IS THE QUESTION

Dengan contoh ini dapat ditunjukan, walaupun ada 26 ! cara untuk menciptakan cryptogram subtitusi sederhana, kita biasanya dapat memecahkan pesan yang sangat pendek dengan membuat keputusan dengan berdasarkan pengetahuan frekuensi huruf dan kata, pola kata seperti THE dan THAT dan dengan membuat serangkaian tebakan dalam bentuk ciphertext K yang diganti dengan O.

1.6. Data Encryption Standard (DES)

Standar ini dibuat oleh National Beraue of Standard USA pada tahun 1977. DES menggunakan 56 bit kunci algoritma enkripsi ini termasuk

yang kuat dan tidak mudah diterobos. Cara enkripsi ini telah dijadikan standar oleh pemerintah amerika serikat sejak 1977 dan menjadi standard ANSI tahun 1981.

DES seharusnya terdiri dari algoritma enkripsi data yang diimplementasikan dalam peralatan elektronik untuk tujuan tertentu. Peratalan ini dirancang menurut cara yang mereka gunakan dalam sistem atau jaringan komputer untuk melengkapi perlindungan *cryptographic* pada data biner.

Metode implementasi akan tergantung pada aplikasi dan lingkungan di sekitar sistem itu. Peralatan itu diimplementasikan tetapi sebelumnya diuji dan divalidkan secara akurat untuk menampilkan transformasi dalam bentuk algoritma.

Algoritma DES¹ dirancang untuk menulis dan membaca berita blok data yang terdiri dari 64 bit dibawah kontrol kunci 64 bit. Dalam pembacaan berita harus dikerjakan dengan menggunakan kunci yang sama dengan waktu menulis berita, dengan penjadualan alamat kunci bit yang diubah sehingga proses membaca adalah kebalikan dari proses menulis.

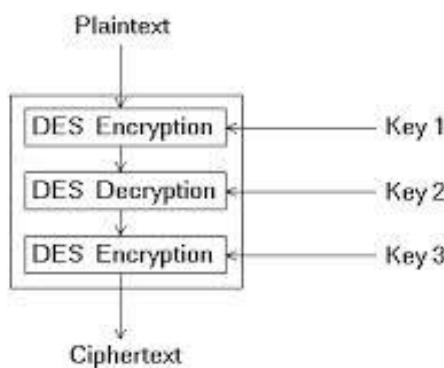
1.7. Triple Data Encryption Standard (Triple DES)

Setelah kita berbicara tentang model enkripsi DES, maka bahasan ini masih ada kaitannya dengan enkripsi DES yaitu Triple DES. Cara ini dipakai untuk membantu DES lebih kuat lagi, yaitu dengan melakukan enkripsi DES tiga kali dengan menggunakan dua kunci yang berbeda. Ternyata, enkripsi dua kali saja dengan dua kunci yang berbeda tidak meningkatkan derajat ketangguhan, hal ini dapat diperlihatkan secara matematis. *Triple DES* ini telah banyak dipakai oleh lembaga keuangan dalam usaha meningkatkan ketangguhan DES.

Triple DES adalah jawaban untuk menutupi kekurangan dari DES. Karena model enkripsi Triple DES didasarkan pada algoritma DES maka sangat mudah untuk memodifikasi *software* yang menggunakan Triple DES. Panjang kunci yang digunakan Lebih panjang sehingga dapat mematahkan serangan yang tiba tiba datang.

Triple DES ini merupakan model yang lain dari operasi DES yang mungkin lebih sederhana. Cara kerja dari model enkripsi ini adalah mengambil 3 kunci sebanyak 64 bit dari seluruh kunci yang mempunyai panjang 192 bit. Triple DES memungkinkan pengguna memakai 3 sub kunci dengan masing masing pajangnya 64 bit. Prosedur untuk enkripsi sama dengan DES, tetapi diulang sebanyak 3 kali. Data dienkrip dengan kunci pertama kemudian dienkrip dengan kunci kedua dan pada akhirnya dienkrip lagi dengan kunci ketiga.²

Perhatikan gambar berikut ini : (Gambar 8.1)



Gambar 8.1 *Triple DES*

Akibatnya , Triple DES menjadi 3 kali lebih lambat dari DES, tetapi lebih aman jika digunakan sebagaimana mestinya. Sayangnya, ada beberapa kunci yang menjadi kunci lemah. Jika semua kunci yaitu 3 kunci, kunci pertama dan kunci kedua atau kuncikedua dan kunci ketiga sama maka prosedur enkripsi secara esensial sama dengan standar DES.

Dengan catatan bahwa meskipun kunci input untuk DES mempunyai panjang 64 bit, kunci yang sebenarnya digunakan oleh DES hanya 56 bit sehingga kurang tepat kalau untuk di terapkan pada masing masing bit.

1.8. Rivest Code 4 (RC4)

RC4 merupakan salah satu algoritma kunci simetris yang berbentuk *stream cipher*. Algoritma ini ditemukan pada tahun 1987 oleh Ronald Rivest dan menjadi simbol keamanan RSA. RC4 menggunakan variable yang panjang kuncinya dari 1 sampai 256 bit yang digunakan untuk menginisialisasikan aliran *pseudo random* bit dan kemudian untuk menggenerasikan aliran *pseudo random* yang menggunakan *XOR* dengan *plaintext* untuk menghasilkan *ciphertext*. Masing masing elemen dalam tabel saling ditukarkan minimal sekali.

Kunci RC4 sering terbatas hanya 440 bit, tapi kadang kadang juga menggunakan kunci 128 bit. Biasanya RC4 digunakan dalam paket software perdagangan seperti LOTUS NOTES dan Oracle Secure SQL. Algoritma RC4 bekerja dalam 2 fase yaitu *key setup* dan *ciphering*. *Key setup* adalah fase pertama dan yang paling sulit dari algoritma ini. Selama *Key setup* N bit (N menjadi panjang kunci), kunci enkripsi digunakan untuk menghasilkan variable enkripsi dengan menggunakan 2 aturan yaitu bagian variable dan kunci serta jumlah N dari operasi percampuran. Percampuran ini terdiri dari penukaran bit, operasi modulo dan rumus yang lain. Operasi modulo adalah hasil sisa dari proses pembagian. Contoh $11/4=2$ sisa 3. Oleh karena itu $11 \bmod 4$ sama dengan 3.

1.9. International Data Encryption Algoritma (IDEA)

Dikembangkan pada tahun 1990 di Swiss oleh kriptografer ternama James Massey dan Xuejia Lai. Algoritma ini menggunakan kunci sepanjang 128 bit. Sampai saat ini nampak sangat tangguh dan belum ada yang menghasilkan menemukan kelemahannya. Algoritma blok cipher dalam IDEA beroperasi dengan menggunakan 64 bit *plaintext* dan blok cipher text yang dikendalikan oleh 12 inovasi dasar dalam desain algoritmanya yang berbentuk tabel.³

Proses dalam algoritma itu terdiri dari 8 putaran enkripsi yang diikuti oleh transformasi output. 64 bit *plaintext* dibagi menjadi 4 bagian yang masing masing terdiri dari 16 bit sub blok dan operasi yang digunakan adalah operasi aljabar dengan 16 bit angka. Putaran enkripsi yang pertama, 16 bit sub blok yang pertama dikombinasikan dengan 16 bit *plaintext* yang kedua dengan menggunakan penambahan modulo 2^{16} , dan

dengan 16 bit *plaintext* yang lain menggunakan penambahan modulo $2^{16}+1$. Dan seterusnya sampai 4 bagian yang terdiri dari 16 bit sub blok dikenai operasi itu.

1.10. Skipjack

Skipjack adalah algoritma enkripsi yang dikembangkan pada tahun 1987 dan baru beroperasi pada tahun 1993. Skipjack ini merupakan algoritma rahasia yang dikembangkan oleh Badan Keamanan Nasional Amerika Serikat yang dalam algoritmanya menggunakan kunci sepanjang 80 bit. Metode inilah yang dipakai dalam *Clipper Chip* dan *Forteza Pccard*, perangkat keras yang dipakai untuk enkripsi. Perintah AS menganjurkan pemakaian *chip* ini untuk peralatan komunikasi sipil(telepon,komputer,dan lain lain), tetapi hal ini banyak ditentang oleh kalangan akademis ,karena peralatan ini masih memungkinkan aparat keamanan untuk menyadap komunikasi yang disandikan dengan alat ini jika diperlukan. Hal ini dianggap mengurangi hak privasi dari masyarakat sipil dalam berkomunikasi. *Clipper chip* masih kontroversial, algoritma *skipjack* ini tergolong algoritma yang tangguh.

Sebagai contoh *clipper chip* ini digunakan untuk melengkapi transmisi telepon dan *Forteza card* digunakan untuk mengenkrip email dan lalu lintas jaringan. Karakteristik kunci dari kedua peralatan ini didesain dengan *backdoors* yang mengizinkan agen pemerintah memonitor transmisi enkripsi tertentu dengan otoritas yang tepat. Skipjack telah dianalisa secara intensif dan tidak mempunyai kelemahan dan tidak ada serangan satupun yang bisa menerobos algoritma ini.

Skipjack mengenkrip dan mendekrip data dalam blok 64 bit dengan menggunakan kunci sepanjang 80 bit. Hal ini berarti mengambil 64 bit blok *plaintext* sebagai input dan 64 bit blok *ciphertext* sebagai output. Skipjack mempunyai 32 lingkaran sehingga algoritma utama akan diulang sebanyak 32 kali untuk menghasilkan *ciphertext*. Jadi dengan adanya putaran ini, maka keamanan dari sebuah pesan akan meningkat.

1.11. Caesar Cipher

Model enkripsi ini pertama kali digunakan oleh Julius Caesar untuk berkomunikasi dengan tentaranya. Adapun cara Julius Caesar berkomunikasi dengan tentaranya dengan cara menggeser setiap huruf dalam pesan yang menjadi algoritma standar, sehingga dia dapat menginformasikan semua keputusannya dan kemudian mengirim pesan ini dalam bentuk yang aman.

Standar *Caesar cipher* memiliki tabel karakter sandi yang dapat ditentukan sendiri. Ketentuan ini berdasarkan suatu kelipatan tertentu, misalnya tabel karakter sandi memiliki kelipatan tiga dari tabel karakter aslinya :

Huruf asli : a b c d e f g h i j k l m n o p q r s t u v w x y z

Huruf sandi : d e f g h i j k l m n o p q r s t u v w x y z a b c

Dalam contoh ini huruf a diganti dengan huruf d, huruf b diganti dengan huruf e dan seterusnya sampai z diganti dengan huruf c. dari sini kita bisa melihat bahwa pengeseran huruf menggunakan 3 huruf ke kanan.

Sehingga jika dikirimkan berita aslinya “transaksi” akan menjadi “wudqvdnvl”. Ketentuan tabel karakter sandi dapat diubah sesuai dengan jumlah kelipatan dari huruf aslinya. Dari algoritma ini, apabila terjadi musuh melakukan sabotase terhadap pesan, itu akan menjadi sia-sia karena hanya kelompok Caesar yang dapat membaca.

Dari masalah yang semakin lama semakin luas, muncul algoritma enkripsi baru yang merupakan pengembangan dari *Caesar cipher* yang dapat memecahkan berbagai masalah yang muncul. Algoritma enkripsi dinamakan *vigenere cipher*. Dimana dasar dari algoritma ini adalah beberapa huruf dari kata kunci yang diambil dari penggeseran yang dilakukan oleh *Caesar cipher*.

Misalnya, jika kata kuncinya adalah “bam”, kemudian setiap huruf ketiga dari *plaintext* mulai pada huruf pertama akan digeser oleh b (=1) dan setiap huruf ketiga pada permulaan huruf kedua akan digeser oleh a (=6) dan setiap huruf ketiga pada permulaan huruf ketiga akan digeser oleh m (=12). Tetapi kita tidak bisa tergantung secara pasti dari pembacaan ini.

1.12. Cost Block Cipher

COST merupakan blok cipher dari bekas Uni Sovyet, yang merupakan singkatan dari “Gosudarstvennyi Standard” atau Standar Pemerintah, standar ini bernomor 28147-89 oleh sebab itu metode ini sering disebut sebagai GOST 28147-89.

GOST merupakan blok *cipher* 64 bit dengan panjang kunci 256 bit. Algoritma ini menginterasi algoritma enkripsi sederhana sebanyak 32 putaran (*round*).⁴ Untuk mengenkripsi pertama-tama plainteks 64 bit dipecah menjadi 32 bit bagian kiri, L dan 32 bit bagian kanan, R. subkunci (*subkey*) untuk putaran *i* adalah K_i . Pada satu putaran ke-*i* operasinya adalah sebagai berikut :

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \text{ xor } f(R_{i-1}, K_i)$$

Sedangkan pada fungsi *f* mula-mula bagian kanan data ditambah dengan subkunci ke-*i* modulus 2^{32} . Hasilnya dipecah menjadi delapan bagian 4 bit dan setiap bagian menjadi input *s-box* yang berbeda. Di dalam GOST terdapat 8 buah *s-box*, 4 bit pertama, 4 bit kedua menjadi *s-box* kedua, dan seterusnya. Output dari 8 *s-box* kemudian dikombinasikan menjadi bilangan 32 bit kemudian bilangan ini dirotasi 11 bit kekiri. Akhirnya hasil operasi ini di-xor dengan data bagian kiri yang kemudian menjadi bagian kanan dan bagian kanan menjadi bagian kiri (*swap*). Pada implementasinya nanti, rotasi pada fungsi *f* dilakukan pada awal saat inisialisasi sekalikus membentuk *s-box* 32 bit dan dilakukan satu kali saja sehingga menghemat operasi dan dengan demikian mempercepat proses enkripsi atau dekripsi.

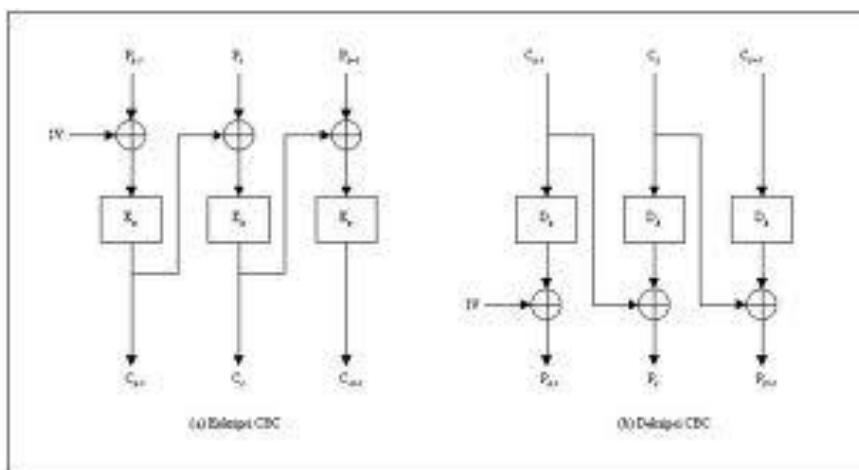
Dubkunci dihasilkan secara sederhana yaitu dari 256 bit kunci yang dibagi menjadi delapan 32 bit blok : k_1, k_2, \dots, k_8 . Setiap putaran menggunakan subkunci yang berbeda. Dekripsi sama dengan enkripsi dengan ukuran k_i dibalik. Standar GOST tidak menentukan bagaimana menghasilkan *s-box* sehingga ada spekulasi bahwa sebagai organisasi di bekas Sovyet mempunyai *s-box* yang baik dan sebagian diberi *s-box* yang buruk sehingga mudah diawasi. Kelemahan GOST yang diketahui sampai

saat ini adalah karena *key schedule*-nya yang sederhana, sehingga pada keadaan tertentu menjadi titik lemahnya terhadap metode kriptanalisis seperti *Related-key Cryptanalysis*. Tetapi hal ini dapat diatasi dengan melewatkkan kunci kepada fungsi *hash* yang kuat secara kriptografi seperti SHA-1, kemudian menggunakan hasil *hash* untuk input inisialisasi kunci. Kecepatan dari metode ini cukup baik, tidak secepat Blowfish tetapi lebih cepat dari IDEA.

Pada metode blok *cipher* ada yang dikenal sebagai mode operasi. Mode operasi biasanya mengkombinasikan *cipher* dasar, *feedback* dan beberapa operasi sederhana. Operasi cukup sederhana saja karena keamanan merupakan fungsi dari metode *cipher* yang mendasarinya bukan pada modenya. Mode pertama adalah ECB (*Electronic CodeBook*) dimana setiap blok dienkrip secara *independen* terhadap blok lainnya.

Dengan metode operasi ini dapat saja sebuah pesan disisipkan diantara blok tanpa diketahui untuk tujuan tertentu, misalnya untuk mengubah pesan sehingga menguntungkan si pembobol. Mode lainnya adalah CBC (*Cipher Block Chaining*) dimana *plaintext* dikaitkan oleh operasi xor dengan *cipherteks* sebelumnya, metode ini dapat dijelaskan seperti pada Gambar 8.2.

Untuk mode ini diperlukan sebuah *Initialization Vector* (IV) yang akan di-xor dengan *plaintext* yang paling awal. IV ini tidak perlu dirahasiakan, karena bila kita perhatikan jika terdapat n blok maka akan terdapat (n-1) IV yang diketahui. Metode lain yang dikenal adalah CFB (*Cipher Feedback*), OFB (*Output Feedback*), *Counter Mode*, dan lain-lain.



Gambar 8.2 Mode operasi CBC

1.13. Letter Map

Standard *letter map* menggunakan table korespondensi yang dipilih secara sembarang misalnya:

Huruf asli : a b e d e f g h l j . . .

Huruf sandi : q w e r t y u l o p . . .

Sehingga jika dikirimkan berita asli “baca” akan menjadi “wpep”. Ketentuan ini tidak mutlak, aturan sandi bisa berubah – ubah tergantung dari orang yang mengirimkannya .

1.14. Tranportation Cipher

Standard *transportation cipher* menggunakan huruf kunci yang di beri nama dan nomor kolom sesuai dengan urutan huruf pada huruf kunci tersebut, misalkan ditentukan huruf kunci adalah SARANA akan digunakan untuk mengirimkan berita “naskah buku segera dikirimkan sebelum deadline”.

Perhatikan Tabel 8.1 berikut ini:

Table 8.1 Contoh dari Standart Transportation Chiper

S	A	R	A	N	A
1	6	3	4	2	5
N	A	S	K	A	H
B	U	K	U	S	E
G	E	R	A	D	I
K	I	R	I	M	K
A	N	S	E	B	E
L	U	M	D	E	A
D	L	I	N	E	

Pada saat dikirimkan, berita tersebut menjadi “NBGKALDASDMBEE SKRRSMI KUAIEDN HEIAKEA AUEINUL”.

1.15. Blowfish

Blowfish merupakan metoda enkripsi yang mirip dengan DES dan diciptakan oleh Bruce Schneier yang ditujukan untuk mikroprosesor besar

(32 bit ke atas dengan cache data yang besar). Blowfish dikembangkan untuk memenuhi kriteria disain sebagai berikut:

- Cepat, pada implementasi yang optimal Blowfish dapat mencapai kecepatan 26 *clock cycle* per byte.
- Kompak , Blowfish dapat berjalan pada memori kurang dari 5 KB
- Sederhana, Blowfish hanya menggunakan operasi yang sederhana yaitu : penambahan (*addition*), XOR, dan penelusuran table (*table lookup*) pada *operand* 32 bit. Desainnya mudah untuk dianalisa yang membuatnya resisten terhadap kesalahan implementasi. Keamanan yang variable, panjang kunci Blowfish dapat bervariasi dan dapat mencapai 448 bit (56 byte).

Blowfish dioptimalkan untuk aplikasi dimana kunc tidak sering berubah, seperti jalur komunikasi atau enkripsi file otomatis. Blowfish jauh lebih cepat dari DES bila diimplementasikan pada 32 bit mikroprosesor dengan cache data yang besar. Blowfish merupakan blok *Cipher* 64-bit dengan panjang kunci variabel. Algoritma ini terdiri dari 2 bagian :*key expansion* dan enkripsi data. *Key expansion* merubah kunci yang dapat mencapai 448 bit menjadi beberapa array subkunci (*subkey*) dengan total 4168 byte.

Enkripsi data terdiri dari iterasi fungsi sederhana sebanyak 16 kali. Setiap putaran terdiri dari permutasi kunci *dependen* dan substitusi kunci dan data *dependen*. Semua operasi adalah penambahan dan XOR pada variable 32-bit. Tambahan operasi lain –nya hanyalah empat penelusuran table (*table lookup*) array berindeks untuk setiap putaran .

1.16. Enigma Cipher

Enigma Cipher adalah suatu metode yang terkenal pada waktu perang dunia ke 2 bagi pihak jerman. Waktu itu dikembangkan sesuatu metode atau model yang di sebut dengan mesin Enigma. ⁵mesin didasarkan pada system 3 rotor yang menggantikan huruf dalam *ciphertext* dengan huruf dalam *plaintext*. *Rotor* itu akan berputar dan menghasilkan hubungan antara huruf yang satu dengan huruf yang lain, sehingga menampilkan berbagai substitusi seperti pergeseran Caesar.

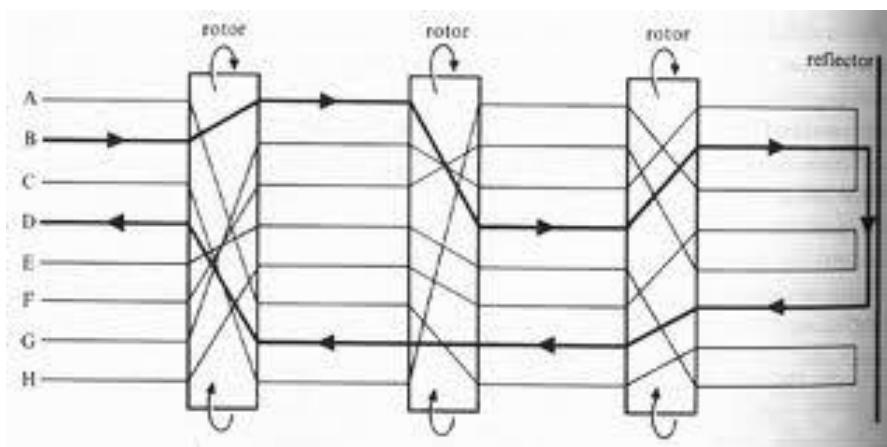
Ketika satu huruf diketik pada keyboard mesin, hal pertama yang dilakukan adalah pengiriman ke *rotor* pertama yang kosong kemudian akan menggeser huruf menurut kondisi yang ada. Setelah itu huruf baru akan melewati *rotor* kedua, dimana akan terjadi pergantian oleh substitusi menurut kondisi yang telah ditentukan di *rotor* kedua. Baru setelah itu, huruf baru ini akan melewati *rotor* ketiga dan hasilnya akan di substitusikan lagi. Sampai huruf baru ini akhirnya akan dikembalikan pada *reflector* dan kembali lagi melalui 3 *rotor* dalam urutan yang terbalik.

Kondisi yang membuat Enigma kuat adalah putaran *rotor*.

Karena huruf plaintext melewati *rotor* pertama akan berputar 1 posisi. 2 *rotor* yang lain akan meninggalkan tulisan sampai *rotor* yang pertama telah berputar 26 kali (jumlah huruf dalam alphabet serta 1 putaran penuh). Kemudian *rotor* kedua akan berputar 1 posisi. Sesudah *rotor* kedua terus berputar 26 kali (26×26 huruf, karena *rotor* pertama harus berputar 26 kali untuk setiap waktu *rotor* kedua berputar), *rotor* ketiga akan berputar 1 posisi.

Siklus ini akan berlanjut untuk seluruh pesan yang dibaca. Dengan kata lain, hasilnya merupakan geseran yang digeser. Sebagai contoh, huruf s dapat disandikan sebagai huruf b dalam bagian pertama pesan, kemudian huruf m berikutnya dalam pesan. Sehingga dari 26 huruf dalam alphabet akan muncul pergeseran $26 \times 26 \times 26$ yaitu posisi *rotor* yang mungkin.

Gambar dibawah ini diambil dari Alan Turing : The Enigma; Simon and Schuster; 1983 oleh Andrew Hodges. (Gambar 8.3)



Gambar 8.3 *Enigma Machine*

Dari sini kita dapat penjelasan tentang grafis dari apa yang terjadi ketika kunci ditekan pada mesin enigma. Supaya lebih sederhana, dalam gambar itu hanya dimunculkan 8 huruf alfabet, sedangkan mesin yang asli menggunakan semua huruf yaitu 26 huruf.

1.17. Enkripsi dengan Kunci Publik

Cara enkripsi ini mempunyai banyak kelebihan, salah satunya adalah tiap orang hanya perlu memiliki satu set kunci, tanpa peduli berapa banyak orang yang akan di ajak berkomunikasi. Jadi jika ada n orang berkomunikasi dengan cara ini hanya dibutuhkan n set kunci. Selain itu, cara enkripsi ini tidak membutuhkan saluran aman untuk pengiriman kunci, sebab kunci yang dikirim ini memang harus di ketahui publik. Cara enkripsi ini sangat praktis sehingga masyarakat umum pun dapat dengan mudah memakainya.

Cara kerja enkripsi ini secara singkat dapat diterangkan sebagai berikut. Setiap orang yang menggunakan enkripsi ini harus memiliki dua buah kunci, satu di sebut kunci rahasia yang hanya boleh diketahui oleh dirinya sendiri dan yang lain di sebut kunci publik yang di sebarkan ke orang lain. Kedua kunci ini dibuat secara acak dengan menggunakan rumus matematika tertentu, jadi kunci ini berkaitan erat secara matematis. Jika si A hendak mengirim pesan dengan si B, si A perlu mengenkripsi pesan tersebut dengan kunci publik si B. Pesan si A yang telah dienkripsi dengan menggunakan kunci publik si B hanya bisa dibuka dengan kunci rahasia si B. Walaupun dienkripsi dengan kunci publik si B, pesan ini tidak bisa dibuka dengan kunci publik itu sendiri. Adalah kewajiban si B untuk menjamin keamanan kunci rahasianya.

Karena kunci rahasia ini tidak perlu diketahui si pengirim berita, kunci ini tidak akan pernah dikirim lewat jalur umum. Hal ini membuat cara ini jauh lebih aman daripada kunci pribadi. Orang lain, misalnya saja si C, Dapat mengirim ke B dengan kunci publik si B yang sama. Walaupun mengetahui publik si B, pesan yang telah dienkripsi dengan itu sangat sulit dibuka. Cara enkripsi ini dikategorikan dalam kriptografi asimetris, karena kunci yang dipakai untuk mengenkripsi dan untuk membuka enkripsi adalah dengan menggunakan 2 kunci yang berbeda.

Ada beberapa algoritma yang terkenal dari cara enkripsi ini, misalnya : Sistem Diffie Hellman, RSA, dan PGP.

1.18. Sistem Diffie Hellman

Kunci pertukaran ini di temukan oleh Whitfield Diffie dan Martin Hellman pada tahun 1976 dan sebelumnya ditemukan oleh Malcolm Williamson pada tahun 1974. Sistem ini dipakai untuk menyandikan pertukaran pesan antar dua pihak secara interaktif. Pada awalnya, masing-masing pihak mempunyai sebuah kunci rahasia yang tidak diketahui lawan bicara. Dengan berdasarkan pada masing-masing kunci rahasia ini, kedua pihak dapat membuat sebuah kunci sesi (*session key*) yang dipakai untuk pembicaraan selanjutnya.

Pembuatan kunci sesi ini dilakukan seperti halnya suatu tanya jawab matematis, hanya pihak yang secara aktif ikut dalam tanya jawab ini sajalah yang bisa mengetahui kunci sesinya. Penyadap yang tidak secara aktif mengikuti tanya jawab ini tidak akan bisa mengetahui kunci sesi ini. Meskipun tidak mungkin mengenkrip langsung dalam Sistem Diffie Hellman, Hal ini masih berguna dalam pengiriman pesan rahasia.

Metode Differ Hellman⁶, Seperti RSA juga menggunakan aritmetik modulus, tetapi disini modulus hanya difokuskan pada bilangan prima, yang disebut P. Dalam sistem Diffie Hellman ,ada 2 kelompok yang masing-masing berpikir dari angka acak rahasia yaitu X dan Y. Masing-masing mengirim kedua komponen itu sehingga satu kelompok tahu ada X dan ada A^Y dan kelompok yang lain tahu ada Y dan A^X . Masing –masing kelompok dapat menghitung $A^{(X*Y)}$ yang dijabarkan menjadi $(A^Y)^X$ dan juga $(A^X)^Y$. Misalnya seseorang mendengar perhitungan ini secara diam-diam, dia tidak akan mengerti maksudnya.

1.19. RSA

RSA adalah singkatan dari hruf depan dari 3 orang yang menemukannya pada tahun 1977 di MTT yaitu, Adi Shamir dan Len Adleman. Algoritma ini merupakan cara enkripsi publik yang paling kau saat ini. Algoritma RSA melibatkan seleksi digit angka prima dan mengalikan secara bersama untuk mendapat jumlah, yaitu n. angka-angka ini dilewati algoritma

matematis untuk menentukan kunci publik $KU=\{e,n\}$ dan kunci pribadi $KR=\{d,n\}$ yang secara matematis berhubungan. Ini merupakan hal yang sulit untuk menentukan e dan d diberi n . Dasar inilah yang menjadi algoritma RSA.⁷

Sekali kunci telah diciptakan, sebuah pesan dapat dienkrip dalam blok dan melewati persamaan berikut ini:

$$C=M^e \text{ mod } n \quad (1)$$

Dimana C adalah *ciphertext*, M adalah *plaintext*, sedangkan e adalah kunci publik penerima. Dengan demikian pesan di atas dapat dienkrip dengan persamaan berikut:

$$C=M^e \text{ mod } n \quad (2)$$

Dimana d adalah kunci pribadi penerima. Sebagai contoh, kita mengasumsikan bahwa $M=19$ (kita akan menggunakan jumlah yang kecil untuk hal yang sederhana dan nantinya secara normal jumlah-jumlah ini akan menjadi besar). Kita akan menggunakan angka 7 sebagai huruf q . Jadi $n=7 \times 17 = 119$, kemudian e dihitung menjadi 5 dan dihitung lagi menjadi 77. $KU=\{5, 119\}$ dan $KR=\{77, 119\}$. Kita dapat melalui nilai yang dibutuhkan dengan persamaan 1 untuk mencari nilai C . Dalam hal ini $C=66$, kemudian hasil dienkrip $C(66)$ dapat digunakan untuk mendapatkan nilai *plaintext* yang asli. Untuk persamaan (2) juga mendapat nilai 19 dan *plaintext* yang asli.

1.20. PGP (Pretty Good Privacy)

Setiap orang mempunyai 2 kunci yaitu kunci publik dan kunci pribadi. Ketika seseorang ingin mengirim sesuatu pada si penerima, pengirim mengenkrip dengan kunci publik si penerima. Kemudian hanya cara untuk mendekripnya dengan kunci pribadi si penerima. Salah satu keuntungan lain dari PGP adalah mengizinkan pengirim menandai perubahan selama perjalanan.

Berdasarkan pada teori ini, PGP mengizinkan seseorang untuk mengungkapkan kunci publik mereka dan menjaga kunci pribadi yang

sifatnya rahasia. Hasilnya seseorang dapat mengenkrip pesan kepada orang lain sepanjang mereka mempunyai kunci publik.

PGP adalah suatu metode enkripsi informasi yang bersifat rahasia, sehingga jangan sampai diketahui oleh orang yang tidak berhak. Informasi ini bisa berupa email yang sifatnya rahasia nomor kode kartu kredit, atau pengiriman dokumen rahasia perusahaan melalui internet. PGP menggunakan metode kriptografi yang disebut "*public key encryption*"; yaitu suatu metode kriptografi yang sangat *sophisticated*.

Adapun prinsip kerja dari PGP adalah sebagai berikut:

PGP, seperti yang telah dijelaskan sebelumnya, menggunakan teknik yang disebut publik *key encryption* dengan dua kunci. Kode-kode ini berhubungan secara intrinsik, namun tidak mungkin untuk memecahkan satu dan yang lainnya. Ketika dibuat satu kunci, maka secara otomatis akan dihasilkan sepasang kunci yaitu kunci publik dan kunci rahasia. Si A dapat memberikan kunci publik kemanapun tujuan yang diinginkannya, melalui telepon, internet, *keyserver*, dan sebagainya. Kunci rahasia yang disimpan pada mesin si A dan mennggunakan *messenger* *decipherakan* dikirimkan ke si A.. Jadi orang lain yang akan menggunakan kunci publik milik A (yang hanya dapat didekripsi oleh kunci rahasia milik si A), mengirimkan pesan kepada A, dan A akan menggunakan kunci rahasia untuk membacanya.

Mengapa menggunakan dua kunci? Karena dengan *conventional crypto*, di saat terjadi transfer informasi kunci, diperlukan suatu *secure channel*. Dan jika memiliki sesuatu *secure channel*, mengapa masih *crypto*? Dengan *public-key system*, tidak akan menjadi masalah siapa yang melihat kunci milik kita, karena kunci yang dilihat orang lain adalah adalah yang digunakan hanya untuk enkripsi dan hanya pemilik saja yang mengetahui kunci rahasia tersebut. Kunci rahasia merupakan kunci yang berhubungan secara fisik dengan komputer pemilik, kunci publik yang ada da kemudian dimungkinkan lagi *passphrase*. Dengan demikian, seseorang mungkin

dapat mencuri *passphrase* yang kita ketikkan,namun ia hanya dapat membaca jika ia dapat mengakses komputer kita.

Setelah mengetahui prinsip kerja dari PGP,berikut akan ditunjukkan penerapannya pada jaringan.Kunci publik sangat lambat bila dibandingkan dengan konvensional.jadi PGP akan mengkombinasikan dua algoritma,yaitu RSA and IDEA,untuk melakukan enkripsi *plaintext*.

2. Definisi Kriptografi

Cryptography atau kriptografi adalah suatu ilmu ataupun seni mengamankan pesan,dan dilakukan oleh cryptographer.Sedang,cryptanalysis adalah suatu ilmu dan seni membuka (breaking) *ciphertext* dan orang melakukannya disebut crptanalyst.

Cryptography system atau *cryptosystem* adalah suatu fasilitas untuk mengkonversikan *plaintext* ke *ciphertext* dan sebaliknya.Dalam sistem ini,seperangkat parameter yang menentukan transformasi penchiperan tertentu disebut suatu kunci.Proses enkripsi dan dekripsi diatur oleh satu atau beberapa kunci kriptografi,Secara umum,kunci-kunci yang digunakan untuk proses pengenkripsi tidak perlu identik,tergantung pada sistem yang digunakan.

Algoritma kriptografi terdiri dari algoritma enkripsi (E) dan algoritma deskripsi (D).Algoritma enkripsi menggunakan kunci enkripsi(KE),sedangkan algoritma dekripsi menggunakan kunci dekripsi (KD).

Secara umum operasi enkripsi dan dekripsi dapat diterangkan secara matematis berikut:

$$EK(M)=C(\text{proses enkripsi})$$

$$DK(C)=(\text{proses dekripsi})$$

Pada saat enkripsi kita menyandikan pesan M dengan suatu kunci K lalu dihasilkan pesan C.sedangkan proses dekripsi,pesan C tersebut

diuraikan dengan menggunakan kunci K sehingga dihasilkan pesan yang sama seperti pesan sebelumnya.

Dengan demikian keamanan suatu pesan tergantung pada kunci ataupun kunci-kunci digunakan, dan tidak tergantung pada algoritma yang digunakan tersebut dapat dipublikasikan dan dianalisis, serta produk-produk yang menggunakan algoritma tersebut dapat dproduksi secara umum. Tidaklah menjadi masalah apabila seseorang mengetahui algoritma yang kita gunakan. Selama ia tidak mengetahui kunci yang dipakai, ia tetap tidak membaca pesan.

2.1. Algoritma Kriptografi

Sampai sekarang, berbagai algoritma kriptografi telah diusulkan dan masing-masing mempunyai karakteristik yang berbeda-beda. Diantara karakteristik-karakteristik itu paling mendasar yang akan digunakan pada sistem jaringan, jaringan komputer maupun internet. Komponen-komponen yang sangat penting adalah *secrecy, integrity, dan authenticity*.

Secrecy adalah komponen yang akan digunakan untuk menjaga pesan yang biasanya digunakan oleh seseorang yang mengirim pesan. Komponen ini hanya mengizinkan seseorang yang tahu akan kunci pada pesan yang telah dienkripsi dengan algoritma kriptografi.

Integrity adalah komponen yang digunakan untuk memeriksa apakah sebuah pesan telah dirubah pada saat pengiriman, biasanya menggunakan algoritma hash, sebagai contoh, algoritma tanda tangan digital menggunakan konsep yang sama dengan tanda tangan biasa.

Berbagai algoritma kriptografi telah dikembangkan sampai sekarang. Kecuali fungsi hash, semua fungsi yang lain, menggunakan kunci untuk memperoleh atribut yang dikehendaki. Karakteristik kunci yang menggunakan algoritma kriptografi dapat digolongkan sebagai berikut: algoritma kriptografi kunci rahasia (algoritma kriptografi simetris), algoritma kriptografi public (algoritma kriptografi kunci asimetris) dan algoritma hash.

Secara umum, algoritma kriptografi kunci rahasia menyatakan bahwa algoritma dimana enkripsi digunakan dalam meng-

enkripsi data, dan kunci deskripsi untuk merubah kembali ke data aslinya. Karena atribut ini, algoritma kriptografi kunci rahasia disebut juga algoritma kriptografi kunci simetris.

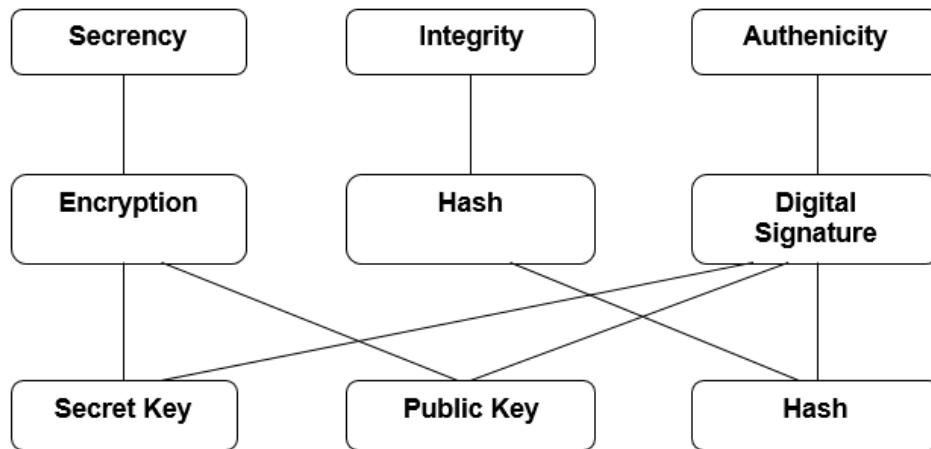
Untuk algoritma kriptografi kunci publik menyatakan bahwa algoritma enkripsi dan deskripsi berbeda. Algoritma kriptografi kunci publik mempunyai karakteristik yaitu tidak ada penghitungan kembali dari kunci deskripsi bahkan sesudah kunci enkripsi dilakukan. Berawal dari kondisi ini, kunci enkripsi disebut kunci pribadi. Sedangkan, algoritma *hash* menyatakan algoritma dimana panjang pesan yang khusus.

Algoritma *hash* yang digunakan dalam kriptografi dibagi menjadi 2 bagian yaitu : dengan kunci dan tanpa kunci. Ketika menggunakan fungsi *hash* dengan kunci maka menggunakan metode yang sama karena kondisi ini terjadi dalam algoritma kunci rahasia.

Algoritma kriptografi kunci rahasia dan publik menggunakan algoritma enkripsi tanda tangan digital secara berturut-turut. Dalam algoritma enkripsi, isi pesan hanya dapat dilihat oleh pribadi yang tahu kuncikripsi, sedangkan dalam tanda tangan digital, pengirim pesan dapat diketahui.

Dalam tanda tangan digital kunci publik, kunci enkripsi (kunci publik) untuk enkripsi data dan kunci deskripsi (kunci pribadi) untuk deskripsi data. Disini kunci rahasia digunakan untuk tanda tangan, sedangkan kunci publik digunakan untuk mengecek. Hal ini akan menolong kunci rahasia untuk tidak bisa dibaca oleh orang lain dan hanya mengizinkan orang yang diberi kuasa untuk membuat tanda tangan. Tetapi, kunci publik dapat dilihat oleh beberapa orang sehingga bisa dengan mudah memperoleh dan menggunakannya

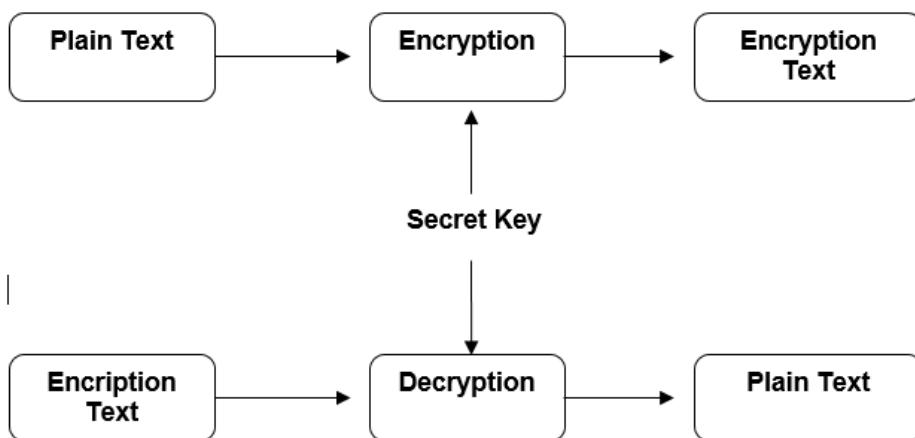
Karakteristik dan tipe dari algoritma kriptografi dapat di lihat pada gambar berikut ini : (Gambar 8.4)



Gambar 8.4 Tipe dan Karakteristik Algoritma Kriptografi

2.1.1. Algoritma Kriptografi Kunci Rahasia

Dalam algoritma kriptografi kunci rahasia, kunci algoritma digunakan untuk enkripsi data dan tidak diberi kuasa kepada publik melainkan hanya kepada orang tertentu yang tahu dan dapat membaca data dienkrip. Karakteristik algoritma kriptografi kunci rahasia adalah bahwa kunci enkripsi sama dengan kunci deskripsi seperti yang ditunjukkan pada gambar 8.5



Gambar 8.5 Algoritma Kriptografi Kunci Rahasia

Algoritma kriptografi kunci rahasia juga disebut algoritma kriptografi simetris. Untuk menggunakan algoritma kriptografi kunci rahasia dalam

komunikasi, kedua belah pihak hanya berkomunikasi satu dengan yang lainnya harus saling membagi kunci enkripsi sebelumnya.

Dalam algoritma kriptografi kunci rahasia, kunci enkripsi dan kunci deskripsi adalah sama. Satu metode untuk menghasilkan kunci dengan cara menggunakan pembangkit bilangan acak yang telah di *install* kedalam komputer, sedangkan metode yang lainnya untuk merancang dan menghasilkan kunci untuk penggunaanya sendiri.

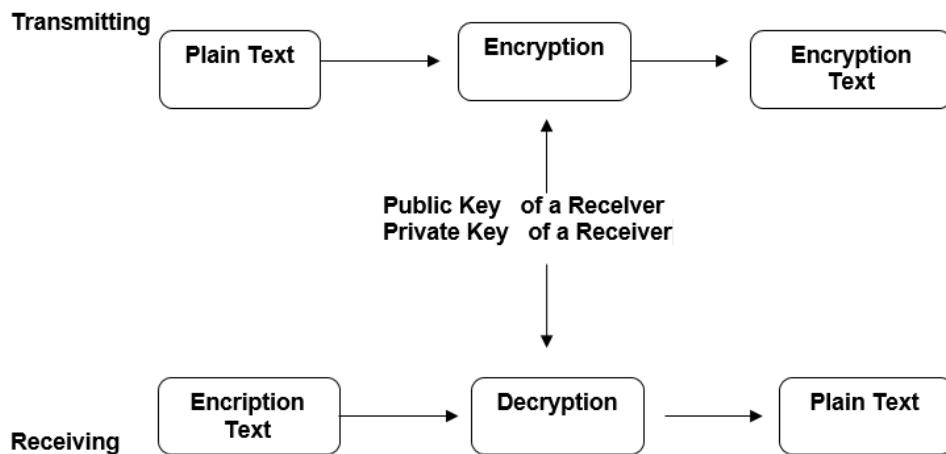
Algoritma kriptografi kunci rahasia memerlukan perawatan yang khusus dalam pemberian kunci deskripsi ke pihak yang lain, sejak orang yang tahu kunci deskripsi dapat mendekrip teks dengan mudah.

2.1.2 Algoritma Kriptografi Kunci Publik

Dalam algoritma kriptografi kunci publik, kunci enkripsi dan deskripsi sama. Untuk alasan ini, algoritma kunci publik disebut algoritma kriptografi kunci asimetris. Karakteristik algoritma kriptografi kunci publik adalah bahwa kunci deskripsi tidak dibuka bahkan sesudah kunci enkripsi dibuka. Untuk memperoleh atribut ini, algoritma kriptografi kunci publik dirancang pada mekanisme yang sulit untuk dipecahkan secara matematika.

Dalam algoritma kriptografi kunci publik, kunci enkripsi dibuka sehingga tidak seorangpun dapat menggunakannya, tetapi untuk deskripsi, hanya seseorang yang punya kunci yang dapat menggunakannya. Untuk alasan ini, kunci yang digunakan untuk enkripsi disebut juga kunci publik sedangkan kunci yang digunakan untuk deskripsi disebut kunci pribadi atau kunci rahasia.

Ketika menggunakan algoritma kriptografi kunci publik, kunci publik dibuka ke sejumlah orang. Sebagai contoh, perhatikan sebuah kunci yang dikirim kesurat kabar atau sebuah pesan yang dimasukkan ke *home page* dengan pesan: “Silahkan gunakan kunci berikut untuk mngirim teks kepada saya” (Gambar 8.6)



Gambar 8.6 Contoh Algoritma Kriptografi Kunci Publik

Algoritma kriptografi kunci publik digunakan untuk banyak area yang berbeda. Yang paling umum digunakan adalah dalam hal pengiriman kunci rahasia pada tahap awal, dimana algoritma kriptografi kunci rahasia digunakan dalam tanda tangan digital. Dalam hal ini algoritma kriptografi kunci public, seperti disebutkan sebelumnya, kunci public digunakan untuk enkripsi dan kunci pribadi untuk dekripsi penghitungan. Jadi, penghitungan enkripsi dapat dilakukan oleh seseorang sejak kunci dibuat public, sedangkan dekripsi hanya dapat dikerjakan oleh seorang yang mempunyai kunci.

Kerahasiaan dapat diperoleh dari atribut algoritma kriptografi kunci public. Teks yang dienkrip hanya dapat dilihat oleh orang yang mempunyai kunci pribadi. Sistem tanda tangan digital menerapkan atribut algoritma kriptografi kunci public. Dalam sistem itu, penanda tangan menghitung dengan kunci pribadi dari seseorang yang berharap bisa memeriksa tanda tangan dan menghitungnya dengan kunci public. Dalam hal ini, penanda tanganan hanya dapat dilakukan oleh orang yang ditunjuk (yang tahu kunci pribadi), sedangkan pemeriksaan tanda tangan dapat dilakukan oleh siapapun.

Karakteristik tertentu dari tanda tangan digital, tanda tangan itu bersifat unik dan tidak sama serta tidak bisa dibuat oleh yang lain. Sedangkan dalam pemeriksaan tanda tangan dapat dilakukan dengan mudah oleh seseorang yang berharap dapat memeriksanya dengan

memperoleh kunci public. Dengan cara ini, kebenaran penanda tangan dapat diperiksa.

Ketika membandingkan kelebihan dan kelemahan antara algoritma kriptografi kunci rahasia dan public, algoritma kriptografi kunci public pada umumnya mempunyai lebih banyak keuntungan dalam istilah kriptografi. Ini karena informasi rahasia dari seseorang tidak harus dikirim dan enkripsi informasi serta aplikasi berbeda satu dengan yang lain sehingga tidak mudah untuk diintegrasikan. Dapat digunakan untuk system distribusi kunci algoritma kriptografi kunci rahasia, tanda tangan digital untuk pemeriksaan pembuat pesan yang dikirim dan identifikasi untuk pemeriksaan identitas pengguna.

Kendati kelebihan yang telah disebutkan diatas, algoritma kriptografi kunci rahasia digunakan lebih luas daripada algoritma kriptografi kunci public yaitu tentang kecepatan proses pengirimannya. DES, mewakili algoritma kriptografi kunci rahasia dan RSA, mewakili algoritma kriptografi kunci public.

Berangkat dari kelebihan dan kelemahan ini, dua system ini sering digabungkan. Pertama, algoritma kriptografi kunci public digunakan dalam tahap pembuatan dan pembagian sesi kunci yang digunakan dalam berkomunikasi. Setelah itu baru menggunakan algoritma kriptografi kunci rahasia yaitu dalam tahap enkripsi dan dekripsi dari teks yang dilakukan dengan kunci sesi.

2.1.3. Algoritma Hash

Fungsi *hash*mengurangi data dari ukuran yang berubah-ubah menjadi ukuran yang khusus. Fungsi *hash*dibutuhkan dalam bagian konfigurasi system untuk memudahkan pengecekan terhadap kelebihan data. Seluruh data dapat diperiksa untuk melihat apakan data yang berkapasitas besar dapat diulang, sebab hal ini akan mendatangkan kerugian besar dalam kecepatan dan waktu.

Fungsi *hash*digunakan dalam kriptografi yaitu dalam hal membagi atribut yang mirip. Terutama dalam hal tanda tangan digital. Sebagai

contoh, DOS menandai 320 bit dari pesan 160 bit. Bagaimanapun juga, ketika kalimat dapat lebih panjang, pesan ini akan menghasilkan kelambatan dalam proses pengiriman dan penyimpanan, karena panjang pesan menjadi ganda dari pesan aslinya.

Hal ini terjadi pada waktu sesudah tanda tangan dimasukkan dan dibagi dalam blok 160 bit untuk tanda tangan itu sendiri. Sehingga hal ini menyebabkan kecepatan turun dan pesan dianggap tidak valid. Fungsi *hash*dapat mengatasi masalah ini. Caranya adalah pesan dibagi dalam ukuran yang lebih kecil dan panjang yang berubah-ubah dari teks dibuat dalam ringkasan pesan.

Sampai sekarang, berbagai tipe fungsi *hash*telah diusulkan. Contohnya adalah MD5, SHA-1 dan RIPEMD-160. Diantara contoh ini, MD5 tidak memuaskan karena nilai *hash*nya hanya 128 bit. Sedangkan untuk SHA-1 dan RJPEMD-160 mempunyai nilai *hash* lebih dari 160 bit dan tidak mempunyai masalah yang sama dengan MD5.

Bagaimanapun, fungsi *hash*dalam kriptografi dapat dibuat oleh siapapun, tetapi biasanya sering dikombinasikan dengan fungsi yang punya integritas, seperti dalam hal kombinasi antara algoritma tanda tangan digital dan fungsi *hash*atau fungsi *hash*dengan algoritma kriptografi kunci rahasia.

c. Rangkuman.

Berfungsi untuk mentranslasikan data yang hendak ditransmisikan oleh aplikasi ke dalam format yang dapat ditransmisikan melalui jaringan. Protokol yang berada dalam level ini adalah perangkat lunak redirektor (redirector software), seperti layanan Workstation (dalam Windows NT) dan juga Network shell (semacam Virtual Network Computing (VNC) atau Remote Desktop Protocol (RDP)).

Contoh protocol yang ada dalam Presentation Layer :

- SMTP (Simple Mail Transfer Protocol), protokol untuk pertukaran mail.
- SNMP (Simple Network Management Protocol), protokol untuk manajemen jaringan.
- Telnet, protokol untuk akses dari jarak jauh.

- TFTP (Trivial FTP), protokol untuk transfer file.

d.Tes Formatif

1. Sebutkan dan jelaskan model-model enkripsi!
2. Dengan model *transposition cipher* dan ditentukan huruf kunci adalah “Jaringan” bagaimana informasi setelah dienkripsi kalau susunan tabel dengan nomer 74856321 dan digunakan untuk mengirimkan berita “naskah buku segera dikirimkan sebelum deadline”
3. Sebutkan dan jelaskan model-model kriptografi!
4. Sebutkan dan jelaskan algoritma dari :
 - a. Simple Substitution Cipher
 - b. DBS dan *TripleDES*
 - c. RiverstCode4(RC4:)
 - d. IDEA
 - e. Skipjack
 - f. Caesar Cipher
 - g. Cost Block Cipher
 - h. Letter Map
5. Berikan contoh kalimat sebelum diberi enkripsi dan hasil enkripsi dari algoritma yang ada di no 4

e. Lembar Jawaban Test Formatif

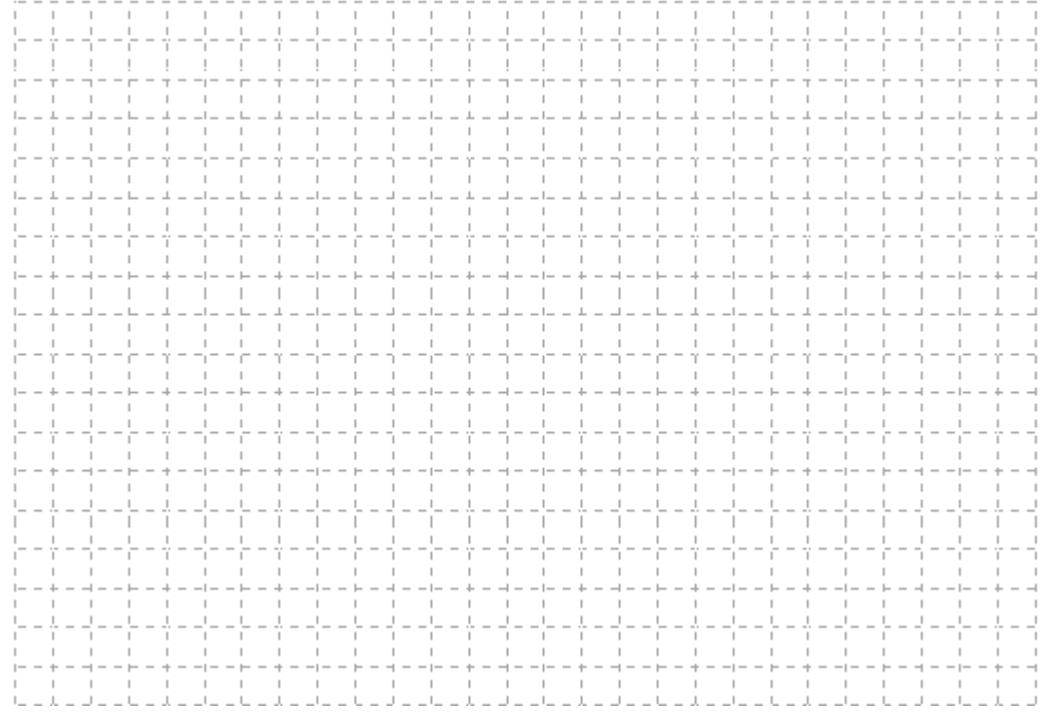
- Test Essay (LJ.01).



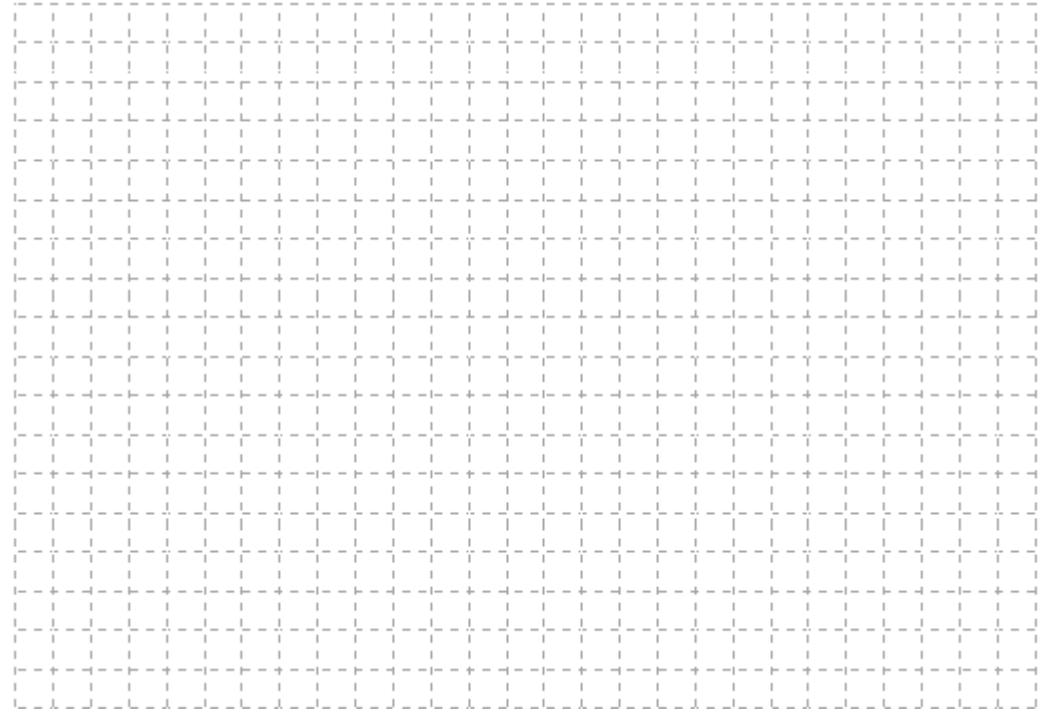
- **Test Essay (LJ.02).**



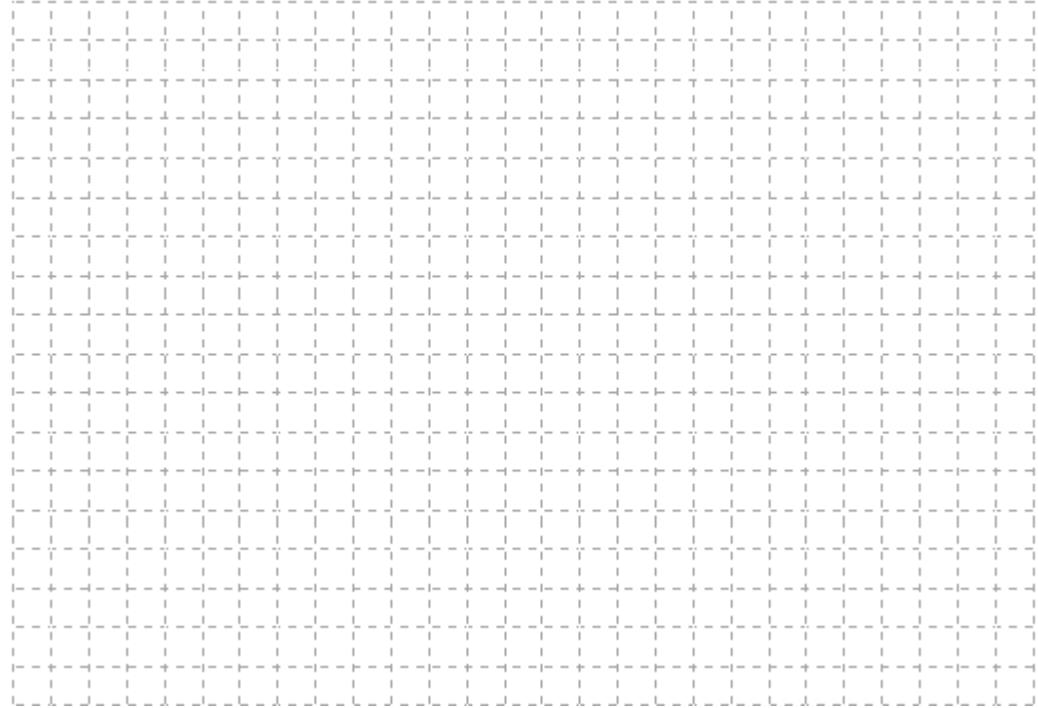
- **Test Essay (LJ.03).**

A large rectangular grid consisting of 20 columns and 25 rows of dashed lines, intended for handwritten responses.

- **Test Essay (LJ.04).**

A second large rectangular grid consisting of 20 columns and 25 rows of dashed lines, intended for handwritten responses.

- **Test Essay (LJ.05).**

A large rectangular grid consisting of 20 columns and 25 rows of small dashed squares, intended for students to write their essay responses.

3. Lembar Kerja Siswa