PROGRAM STUDI

S1 SISTEM KOMPUTER

UNIVERSITAS DIPONEGORO

KEAMANAN MULTIMEDIA

Oky Dwi Nurhayati, ST, MT

Email: okydn@undip.ac.id

MATERI

- Pemahaman Dasar Keamanan Multimedia
- Kriptografi
- Steganografi
- Watermarking

PERLUNYA KEAMANAN MULTIMEDIA

- Perkembangan bisnis konten digital telah membawa peluang baru bagi kejahatan klasik di bidang teknologi informasi, yaitu pembajakan sehingga dibutuhkan suatu mekanisme untuk mengatasi permasalahan pembajakan konten mobile ini. Dari sinilah Digital Rights Management lahir.
- Metode-metode untuk memberikan melindungi data digital, seperti: encryption, copy protection, visible marking, header marking, steganografi, kriptografi, watermarking.

Evolusi dari pengamanan data

Steganography

a. Yunani (Greek) vs Persia

Pesan disembunyikan di meja yang dilapisi lilin

b. Histalaeus

Pesan ditato di kepala budak yang telah digunduli

c. Digital watermarking

Menandai kepemilikan gambar digital

CRYPTOGRAPHY

Private key cryptosystem (Sistem kripto kunci privat)

Simetrik (kunci untuk mengunci dan membuka sama/satu)

Public key cryptosystem (Sistem kripto kunci publik)

Asimetrik (kunci untuk mengunci dan membuka berbeda)

KRIPTOGRAFI

Dua konsep utama :

- Enkripsi adalah proses dimana informasi/data yang hendak dikirim diubah menjadi bentuk yang hampir tidak dikenali sebagai informasi awalnya dengan menggunakan algoritma tertentu.
- Dekripsi adalah kebalikan dari enkripsi yaitu mengubah kembali bentuk tersamar tersebut menjadi informasi awal.

Algoritma kriptografi berdasarkan jenis kunci yang digunakan dapat dibedakan menjadi dua jenis yaitu :

a. Algoritma *simetris*

Dimana kunci yang digunakan untuk proses enkripsi dan dekripsi adalah kunci yang sama. Contoh algoritma Simetri

- Blok Chiper : DES, IDEA, AES
- Stream Chiper : OTP, A5 dan RC4

b. Algoritma *asimetris*

Dimana kunci yang digunakan untuk proses enkripsi dan dekripsi menggunakan kunci yang berbeda. Contoh algoritma : RSA, DSA, ElGamal

Prinsip-prinsip Kriptografi

a. Confidelity (kerahasiaan) yaitu layanan agar isi pesan yang dikirimkan tetap rahasia dan tidak diketahui oleh pihak lain (kecuali pihak pengirim, pihak penerima / pihak-pihak memiliki ijin). Umumnya hal ini dilakukan dengan cara membuat suatu algoritma matematis yang mampu mengubah data hingga menjadi sulit untuk dibaca dan dipahami.

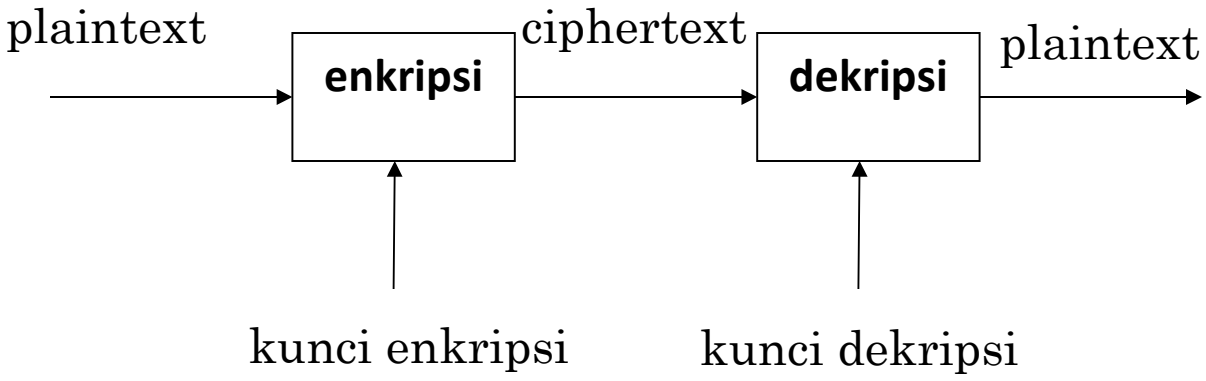
b. Data integrity (keutuhan data) yaitu layanan yang mampu mengenali/mendeteksi adanya manipulasi (penghapusan, pengubahan atau penambahan) data yang tidak sah (oleh pihak lain).

c. Authentication (keotentikan) yaitu layanan yang berhubungan dengan identifikasi. Baik otentikasi pihak-pihak yang terlibat dalam pengiriman data maupun otentikasi keaslian data/informasi.

d. Non-repudiation (anti-penyangkalan) yaitu layanan yang dapat mencegah suatu pihak untuk menyangkal aksi yang dilakukan sebelumnya (menyangkal bahwa pesan tersebut berasal dirinya).

Istilah-istilah dalam bidang Kriptografi

Plaintext (M) adalah pesan yang hendak dikirimkan (berisi data asli).
Ciphertext (C) adalah pesan ter-enkrip (tersandi) yang merupakan hasil enkripsi.
Enkripsi (fungsi E) adalah proses pengubahan *plaintext* menjadi *ciphertext*.
Dekripsi (fungsi D) adalah kebalikan dari enkripsi yakni mengubah *ciphertext* menjadi *plaintext*, sehingga berupa data awal/asli.
Kunci adalah suatu bilangan yang dirahasiakan yang digunakan dalam proses enkripsi dan dekripsi.



Secara matematis, proses enkripsi merupakan pengoperasian fungsi E (enkripsi) menggunakan e (kunci enkripsi) pada M *(plaintext)* sehingga dihasilkan C *(ciphertext),* notasinya :

$$\mathbf{E_e(M) - C}$$

Sedangkan untuk proses dekripsi, merupakan pengoperasian fungsi D (dekripsi) menggunakan d (kunci dekripsi) pada C *(ciphertext)* sehingga dihasilkan M *(plaintext),* notasinya :

$$\mathbf{D_d(C) = M}$$

Sehingga dari dua hubungan diatas berlaku :

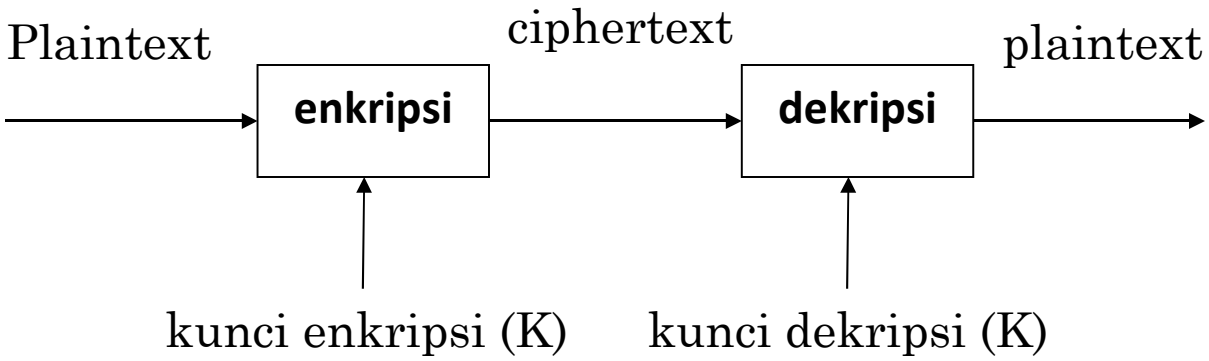
$$\mathbf{D_d(E_e(M)) = M}$$

Pada umumnya kunci publik *(public key)* digunakan sebagai kunci enkripsi sementara kunci privat *(private key)* digunakan sebagai kunci dekripsi.

ALGORITMA SIMETRIS

Algoritma simetris *(symmetric algorithm)* adalah suatu algoritma dimana kunci enkripsi yang digunakan sama dengan kunci dekripsi sehingga algoritma ini disebut juga sebagai *single-key algorithm*.

Diagram proses enkripsi dan dekripsi algoritma simetris



Pengirim dan penerima harus memilih suatu suatu kunci tertentu yang sama untuk dipakai bersama, dan kunci ini haruslah rahasia bagi pihak yang tidak berkepentingan sehingga algoritma ini disebut juga algoritma kunci rahasia *(secret-key algorithm)*.

AES (Advanced Encryption Standard)

- DES dianggap sudah tidak aman.
- Perlu diusulkan standard algoritma baru sebagai pengganti DES.
- National Institute of Standards and Technology (NIST) mengusulkan kepada Pemerintah Federal AS untuk sebuah standard kriptografi kriptografi yang baru.
- NIST mengadakan lomba membuat standard algoritma kriptografi yang baru. Standard tersebut kelak diberi nama Advanced Encryption Standard (AES).
- Pada bulan Oktober 2000, NIST mengumumkan untuk memilih Rijndael (dibaca: Rhine-doll)
- Pada bulan November 2001, Rijndael ditetapkan sebagai AES
- Diharapkan Rijndael menjadi standard kriptografi yang dominan paling sedikit selama 10 tahun

AES (ADVANCED ENCRYPTION STANDARD)

- Tidak seperti *DES* yang berorientasi bit, *Rijndael* beroperasi dalam orientasi *byte*.
- Setiap putaran mengunakan kunci internal yang berbeda (disebut *round key*).
- *Enciphering* melibatkan operasi substitusi dan permutasi.
- Karena *AES* menetapkan panjang kunci adalah 128, 192, dan 256, maka dikenal *AES*-128, *AES*-192, dan *AES*-256

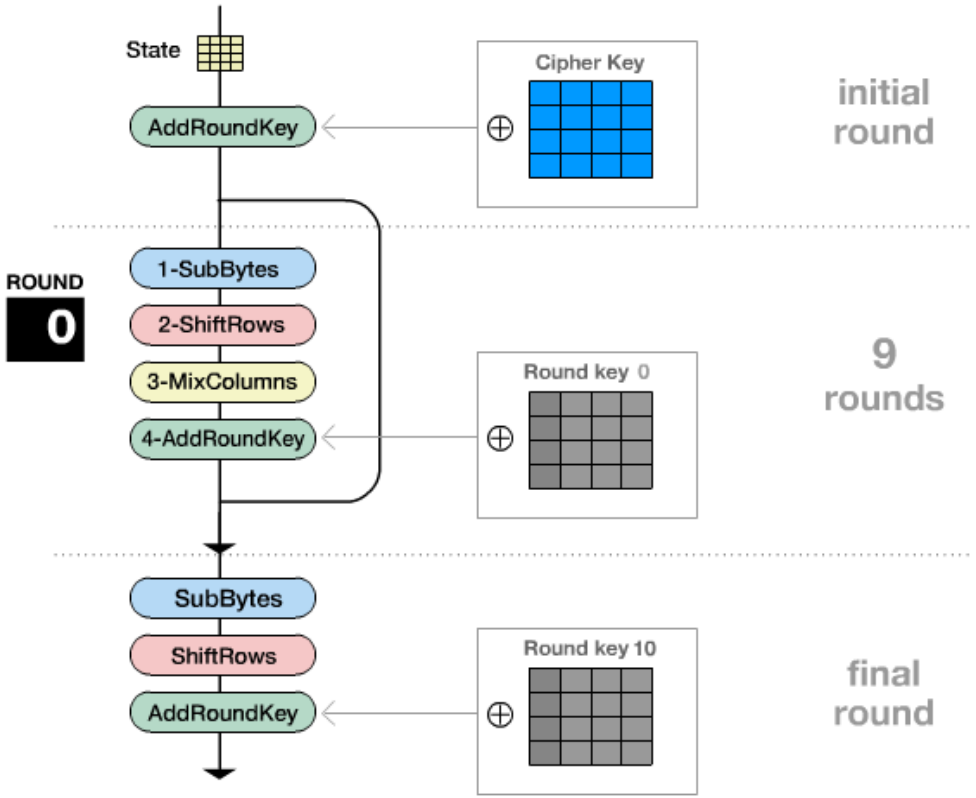
	Panjang Kunci (*Nk words*)	Ukuran Blok (*Nb words*)	Jumlah Putaran (*Nr*)
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

Catatan: 1 *word* = 32 bit

AES (ADVANCED ENCRYPTION STANDARD)

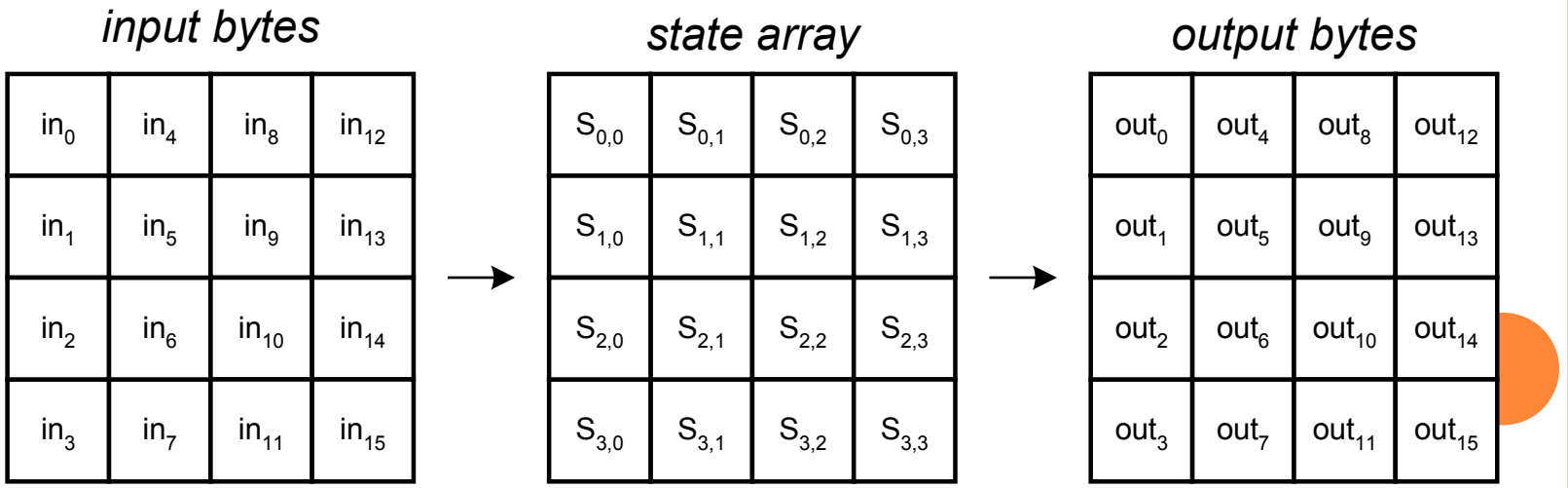
- Garis besar Algoritma *Rijndael* yang beroperasi pada blok 128-bit dengan kunci 128-bit adalah sebagai berikut (di luar proses pembangkitan *round key*):
  - *AddRoundKey*: melakukan *XOR* antara *state* awal (plainteks) dengan *cipher key*. Tahap ini disebut juga *initial round*.
  - Putaran sebanyak *Nr* – 1 kali. Proses yang dilakukan pada setiap putaran adalah:
    - *SubBytes*: substitusi *byte* dengan menggunakan tabel substitusi (*S-box*).
    - *ShiftRows*: pergeseran baris-baris *array state* secara *wrapping*.
    - *MixColumns*: mengacak data di masing-masing kolom *array state*.
    - *AddRoundKey*: melakukan XOR antara *state* sekarang *round key*.
  - *Final round*: proses untuk putaran terakhir:
    - *SubBytes*
    - *ShiftRows*
    - *AddRoundKey*

AES (ADVANCED ENCRYPTION STANDARD)



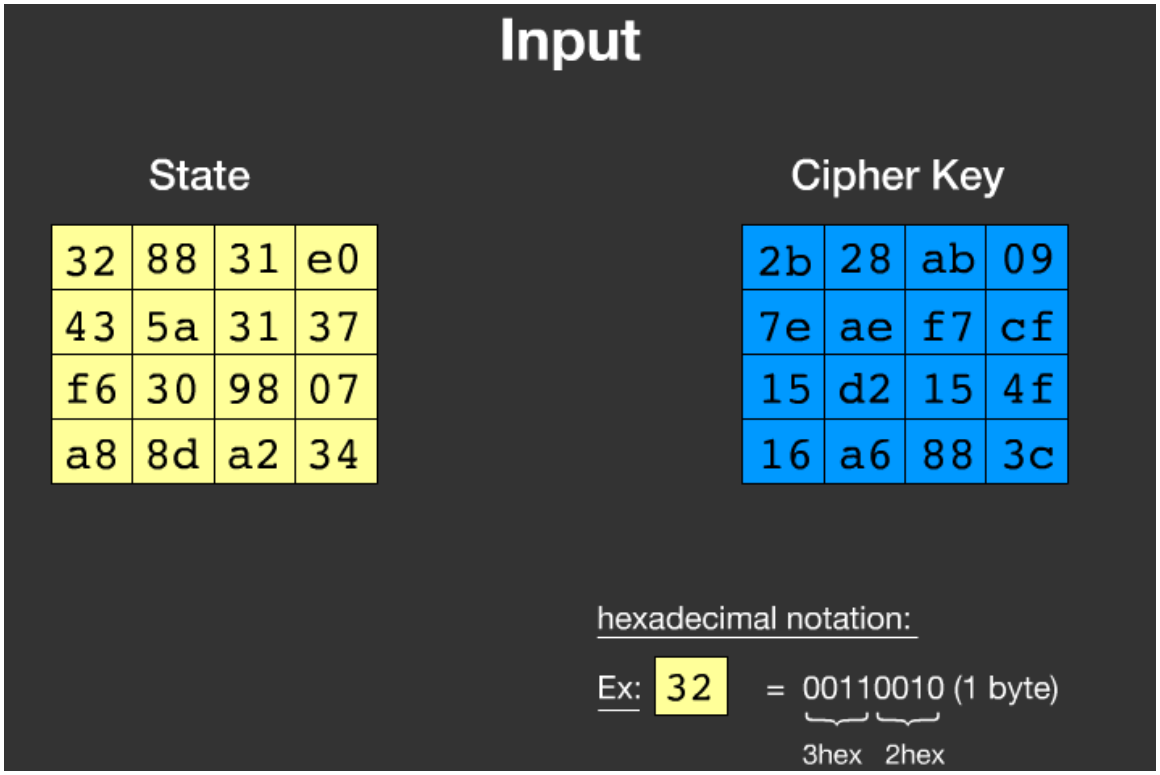
AES (ADVANCED ENCRYPTION STANDARD)

- Selama kalkulasi plainteks menjadi cipherteks, status sekarang dari data disimpan di dalam *array of bytes* dua dimensi, `state`, yang berukuran `NROWS` × `NCOLS`.
- Untuk blok data 128-bit, ukuran `state` adalah 4 × 4.
- Elemen *array* `state` diacu sebagai $S[r,c]$, $0 \le r < 4$ dan $0 \le c < Nb$ (Nb adalah panjang blok dibagi 32.
- Pada *AES*-128, $Nb = 128/32 = 4$)



AES (ADVANCED ENCRYPTION STANDARD)

- Contoh: (elemen state dan kunci dalam notasi HEX)

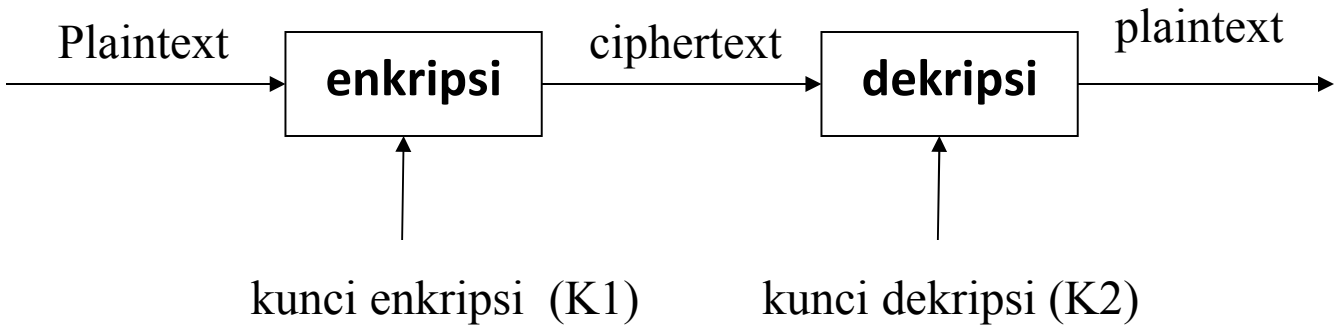


ALGORITMA ASIMETRIS

Suatu algoritma dimana kunci enkripsi yang digunakan tidak sama dengan kunci dekripsi

Pada algoritma ini menggunakan dua kunci yakni kunci publik *(public key)* dan kunci privat *(private key)*. Kunci publik disebarkan secara umum sedangkan kunci privat disimpan secara rahasia oleh si pengguna. Walau kunci publik telah diketahui namun akan sangat sukar mengetahui kunci privat yang digunakan.

Diagram proses enkripsi dan dekripsi algoritma asimetris



RSA

- Ditemukan oleh tiga orang yaitu **R**on Rivest, Adi **S**hamir, dan Leonard **A**dleman yang kemudian disingkat menjadi RSA.
- Termasuk algritma asimetri karena mempunyai dua kunci, yaitu kunci publik dan kunci privat.
- Algoritma kunci-publik yang paling terkenal dan paling banyak aplikasinya.
- Ditemukan oleh tiga peneliti dari MIT (Massachussets Institute of Technology), yaitu Ron Rivest, Adi Shamir, dan Len Adleman, pada tahun 1976.
- Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima.

RSA

Pembangkitan pasangan kunci

- Pilih dua bilangan prima, a dan b (rahasia)
- Hitung $n = a\ b$. Besaran n tidak perlu dirahasiakan.
- Hitung $\phi(n) = (a - 1)(b - 1)$.
- Pilih sebuah bilangan bulat untuk kunci publik, sebut namanya e, yang relatif prima terhadap $\phi(n)$.
- Hitung kunci dekripsi, d, melalui $ed \equiv 1$ (**mod** m) atau $d \equiv e^{-1} \bmod (\phi(n))$

Hasil dari algoritma di atas:

- Kunci publik adalah pasangan (e, n)
- Kunci privat adalah pasangan (d, n)

Catatan: *n* tidak bersifat rahasia, namun ia diperlukan pada perhitungan enkripsi/dekripsi

RSA

Kunci Publik

- Misalkan a = 47 dan b = 71 (keduanya prima), maka dapat dihitung:

  $n = a \times b = 3337$

  $\phi(n) = (a-1)\times(b-1) = 46 \text{ x } 70 = 3220.$

- Pilih kunci publik e = 79 (yang relatif prima dengan 3220 karena pembagi bersama terbesarnya adalah 1).
- Hapus a dan b dan kunci publiknya adalah n=3337 dan e=79

Kunci Privat

- Selanjutnya akan dihitung kunci privat d dengan kekongruenan:

$$e \times d \equiv 1 \pmod{m} = => d = \frac{1 + (k \times 3220)}{79}$$

Dengan mencoba nilai-nilai k = 1, 2, 3, …, diperoleh nilai d yang bulat adalah 1019. Ini adalah kunci privat (untuk dekripsi).

RSA

- Misalkan plainteks M = HARI INI
  atau dalam ASCII: 726582733273873

  Pecah M menjadi blok yang lebih kecil (misal 3 digit):

  $m_1 = 726$ $\quad$ $m_4 = 273$

  $m_2 = 582$ $\quad$ $m_5 = 787$

  $m_3 = 733$ $\quad$ $m_6 = 003$

  (Perhatikan, m_i masih terletak di dalam antara 0 sampai $n - 1$)

RSA

- *Enkripsi setiap blok:*
  $c_1 = 726^{79} \bmod 3337 = 215$
  $c_2 = 582^{79} \bmod 3337 = 776$, dst
  Chiperteks $C = 215\ 776\ 1743\ 933\ 1731\ 158$.

- *Dekripsi (menggunakan kunci privat d = 1019)*
  $m_1 = 215^{1019} \bmod 3337 = 726$
  $m_2 = 776^{1019} \bmod 3337 = 582$ dst untuk sisi blok lainnya
  Plainteks $M = 726582732737873$ yang dalam ASCII karakternya adalah HARI INI.

RSA

- ***Kekuatan dan Keamanan RSA***
  - Kekuatan algoritma *RSA* terletak pada tingkat kesulitan dalam memfaktorkan bilangan non prima menjadi faktor primanya, yang dalam hal ini $n = a \times b$.
  - Sekali n berhasil difaktorkan menjadi a dan b, maka $\phi(n) = (a - 1)\times(b - 1)$ dapat dihitung. Selanjutnya, karena kunci enkripsi e diumumkan (tidak rahasia), maka kunci dekripsi d dapat dihitung dari persamaan $ed \equiv 1 \pmod{n}$.
  - Penemu algoritma RSA menyarankan nilai a dan b panjangnya lebih dari 100 digit. Dengan demikian hasil kali n = a □ b akan berukuran lebih dari 200 digit.
  - Menurut Rivest dan kawan-kawan, usaha untuk mencari faktor bilangan 200 digit membutuhkan waktu komputasi selama 4 milyar tahun! (dengan asumsi bahwa algoritma pemfaktoran yang digunakan adalah algoritma yang tercepat saat ini dan komputer yang dipakai mempunyai kecepatan 1 milidetik).

ALGORITMA KRIPTOGRAFI KLASIK

- Algoritma kriptografi klasik berbasis karakter
- Menggunakan pena dan kertas saja, belum ada komputer
- Termasuk ke dalam kriptografi kunci-simetri
- Algoritma kriptografi klasik:
  a. *Cipher* Substitusi (*Substitution Ciphers*)
  b. *Cipher* Transposisi (*Transposition Ciphers*)

Cipher Substitusi

- Monoalfabet : setiap karakter chipertext menggantikan satu macam karakter plaintext
- Polyalfabet : setiap karakter chipertext menggantikan lebih dari satu macam karakter plaintext
- Monograf /unilateral: satu enkripsi dilakukan terhadap satu karakter plaintext
- Polygraf /multilateral: satu enkripsi dilakukan terhadap lebih dari satu karakter plaintext

Cipher Substitusi - Caesar Cipher

Tiap huruf alfabet digeser 3 huruf ke kanan

p_i : 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
c_i : **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**

Contoh:

Plainteks: AWASI ASTERIX DAN TEMANNYA OBELIX

Cipherteks: **DZDVL DVWHULA GDQ WHPDQQBA REHOLA**

Dalam praktek, cipherteks dikelompokkan ke dalam kelompok n-huruf, misalnya kelompok 4-huruf:
DZDV LDVW HULA GDQW HPDQ QBAR EHOL A

Atau membuang semua spasi:
DZDVLDVWHULAGDQWHPDQQBAREHOLA

Cipher Transposisi

- Cipherteks diperoleh dengan mengubah posisi huruf di dalam plainteks.
- Dengan kata lain, algoritma ini melakukan *transpose* terhadap rangkaian huruf di dalam plainteks.
- Nama lain untuk metode ini adalah **permutasi**, karena *transpose* setiap karakter di dalam teks sama dengan mempermutasikan karakter-karakter tersebut.

Cipher Transposisi

Contoh: Misalkan plainteks adalah

```
FAKULTAS SISTEM KOMPUTER UNIVERSITAS
DIPONEGORO SEMRG
```

Enkripsi:

FAKULTAS
SISTEMKO
MPUTERUN
IVERSITA
SDIPONEG
OROSEMRG

Cipherteks: (baca secara vertikal)
FSMISOAIPVDRKSUEIOUTTRPSLEES….SONAGG
FSMISO AIPVDR KSUEIO …SONAGG

Algoritma Kriptografi Modern

- Beroperasi dalam mode bit (algoritma kriptografi klasik beroperasi dalam mode karakter)
- Kunci, plainteks, cipherteks, diproses dalam rangkaian bit
- Operasi bit xor paling banyak digunakan
- Tetap menggunakan gagasan pada algoritma klasik: substitusi dan transposisi, tetapi lebih rumit (sangat sulit dipecahkan)
- Perkembangan algoritma kriptografi modern didorong oleh penggunaan komputer digital untuk keamanan pesan.
- Komputer digital merepresentasikan data dalam biner.

Algoritma Enkripsi dengan rangkaian b

Pesan (dalam bentuk rangkaian bit) dipecah menjadi beberapa blok

Contoh: Plainteks `100111010110`
Bila dibagi menjadi blok 4-bit

`1001  1101  0110`

maka setiap blok menyatakan `0` sampai `15`:

`9  13  6`

Bila plainteks dibagi menjadi blok 3-bit:

`100  111  010  110`

maka setiap blok menyatakan `0` sampai `7`:

`4  7  2  6`

Perbedaan utama antara steganography dan watermarking adalah pada tujuan atau implementasi kedua metode tersebut. Steganography dimaksudkan dalam komunikasi informasi, sedangkan watermarking dimaksudkan untuk perlindungan hak cipta atau milik siapa dokumen tersebut

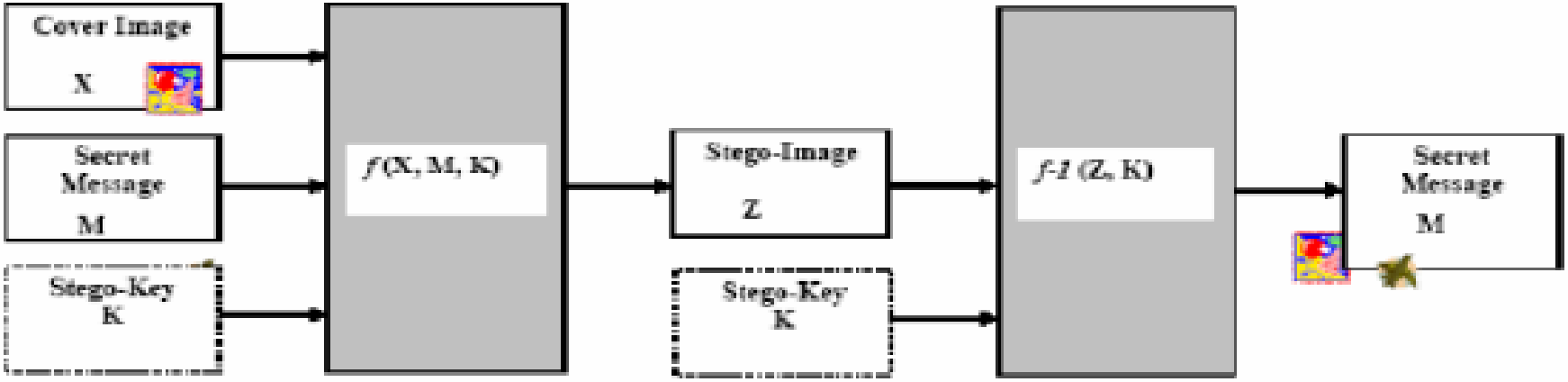
Steganografi adalah suatu teknik untuk menyembunyikan informasi yang bersifat pribadi dengan sesuatu yang hasilnya akan tampak seperti informasi normal lainnya, media yang digunakan umumnya merupakan suatu media yang berbeda dengan media pembawa informasi rahasia .

Watermarking adalah merupakan salah satu bagian dari stega-nography, karena memang teknik-teknik yang dilakukan dalam watermarking merupakan salah satu bagian dari steganogra-phy.

Steganography juga berbeda dengan cryptography yaitu terletak pada hasil keluarannya . Hasil dari cryptography biasanya berupa data yang berbeda dari bentuk aslinya dan biasanya datanya seolah-olah berantakan namun dapat dikembalikan ke data semula. Sedangkan hasil dari keluaran steganography memiliki bentuk yang sama dengan data aslinya, tentu saja persepsi ini oleh indra manusia, tetapi tidak oleh komputer atau pengolah data digital lainnya ..

Steganography keberadaan informasi yang disembunyikan tidak terlihat/diketahui dan terjadi penyampulan tulisan (covered writing). Sedangkan pada cryptography informasi dikodekan dengan enkripsi atau teknik pengkodean dan informasi diketahui keberadaanya tetapi tidak dimengerti maksudnya.

Teknik Steganografi



Tujuan dari teknik-teknik steganografi adalah menyembunyikan keberadaan pesan.

Keamanan dari steganografi ini bergantung pada kunci, yang hanya diketahui oleh pengirim dan penerima pesan. Dalam sistem steganografi yang kuat, hanya pihak yang memiliki kuncilah yang dapat melakukan ekstraksi pesan.

Terdapat beberapa istilah yang berkaitan dengan steganografi.

b.Hiddentext atau embedded message: pesan atau informasi yang disembunyikan.

c.Covertext atau cover-object: pesan yang digunakan untuk menyembunyikan embedded message.

d.Stegotext atau stego-object: pesan yang sudah berisi embedded message. Dalam steganografi digital, baik hiddentex atau covertext dapat berupa teks, audio, gambar, maupun video.

Dalam menyembunyikan pesan, ada beberapa kriteria yang harus dipenuhi.

1). Impercepbility. Keberadaan pesan tidak dapat dipersepsi oleh indrawi. Jika pesan disisipkan ke dalam sebuah citra, citra yang telah disisipi pesan harus tidak dapat dibedakan dengan citra asli oleh mata. Begitu pula dengan suara, telinga haruslah mendapati perbedaan antara suara asli dan suara yang telah disisipi pesan.

2). Fidelity. Mutu media penampung tidak berubah banyak akibat penyisipan. Perubahan yang terjadi harus tidak dapat dipersepsi oleh indrawi.

3). Recovery. Pesan yang disembunyikan harus dapat diungkap kembali. Tujuan steganografi adalah menyembunyikan informasi, maka sewaktu-waktu informasi yang disembunyikan ini harus dapat diambil kembali untuk dapat digunakan lebih lanjut sesuai keperluan

Beberapa istilah yang sering digunakan dalam teknik steganografi:

b. Carrier file : file yang berisi pesan rahasia tersebut

c.Steganalysis : proses untuk mendeteksi keberadaanpesan rahasia dalam suatu file

c. Stego-medium : media yang digunakan untuk membawa pesan rahasia

d. Redundant bits : sebagian informasi yang terdapat di dalam file yang jika dihilangkan tidak akan menimbulakn kerusakan yang signifiakan (setidaknya bagi indera manusia)

e. Payload : informasi yang akan disembunyikan.

TEKNIK WATERMARKING

Tujuan yang ingin dicapai dari penggunaan watemarking :

- Tamper-proofing : Watemarking digunakan sebagai alat indikator yang menunjukkan apakah data digital yang asli telah mengalami perubahan dari aslinya (mengecek integritas data).
- Feature location : Watemarking sebagai alat identifikasi isi dari data digital pada lokasi-lokasi tertentu, misalnya penamaan suatu objek tertentu dari beberapa objek yang ada pada suatu citra digital.

TEKNIK WATERMARKING

- Annotation/caption : Watermark berisi keterangan tentang data digital itu sendiri, misalnya pada broadcast monitoring pada penayangan iklan di stasiun TV. Selain itu, watermark juga dapat digunakan untuk mengirimkan pesan rahasia.
- Copyright-Labeling : Watemarking digunakan sebagai metoda untuk menyembunyikan label hak cipta pada data digital atau sebagai bukti autentik kepemilikan atas dokumen digital tersebut.

JENIS-JENIS WATEMARKING

o Robust watermarking : Jenis watermark ini tahan terhadap serangan (attack), namun biasanya watermark yang dibubuhi ke dokumen masih dapat ditangkap oleh indera penglihatan atau pendengaran manusia.

o Fragile watermarking : Jenis watermark ini akan mudah rusak jika terjadi serangan, namun kehadirannya tidak terdeteksi oleh indera manusia. Jika diinginkan untuk membuat suatu algoritma yang dapat mengimplementasikan watemarking yang memiliki fidelity yang tinggi (adanya watermark tidak disadari oleh pengamatan manusia) maka hasilnya akan semakin rentan terhadap serangan.

Ada tiga tahap utama dalam proses watemarking :

a. mengintegrasikan watermark pada citra (embedding)

b. serangan terhadap citra yang telah dibubuhi watermark, baik yang disengaja (misalnya dikompresi, dipotong sebagian, di-filter, dan sebagainya) ataupun yang tidak disengaja (misalnya disebabkan oleh noise atau gangguan dalam saluran transmisi data).

c. proses ekstraksi watermark dari dokumen yang akan diuji.

Algoritma Enkripsi MPEG

MPEG (IS0 Moving Picture Expert Group) adalah salah satu ekstensi file multimedia yang paling sering digunakan.

Duua masalah besar dalam enkripsi file multimedia.

- Pertama adalah besar ukuran file yang cukup besar (sebagai contoh, ukuran file MPEG yang berdurasi 2 jam mempunyai ukuran sekitar 1 GB).
- File multimedia perlu diproses secara *real-time* (file MPEG dengan *High-Definition* mempunyai *data rates* sekitar 45 Mbps atau lebih*)*.

Memproses data besar seperti ini dalam waktu yang sangat lama akan menjadi beban pada *codec, memori,* sarana penyimpanan data dan komunikasi pada jaringan.

Algoritma harus dibuat sedemikian rupa sehingga mempunyai efisiensi setinggi mungkin dan tingkat atau rasio kompresi yang tinggi

Algoritma Seleksi (*Selective Algorithm*)

Dasar dari Algoritma seleksi adalah berdasarkan pada struktur frame IPB pada file MPEG. Algoritma ini hanya mengenkripsi frame **I saja karena secara konseptual,** frame P dan B menjadi tidak berguna bila tidak mengetahui frame I yang berkorespodensi.

Meyer dan Gadegast telah mendesign sebuah baris bit yang menyerupai MPEG bernama SECMPEG, yang menggunakan enkripsi seleksi dan informasi tambahan pada *header, dan mempunyai waktu eksekusi software* yang cepat.
SECMPEG dapat menggunakan algoritma enkripsi standard seperti DES dan RSA dan mengimplementasikan empat tingkat keamanan.
Tingkat pertama adalah dengan mengenkripsi semua header. Tingkat kedua adalah mengenkripsi sebuah header ditambah dengan koefisien DC dan bagian bawah dari AC pada blok I. Tingkat ketiga adalah dengan mengenkripsi sebuah frame I dan semua blok I pada frame P dan B. Tingkat keempat adalah mengenkripsi semua data. SECMPEG tidak cocok atau sesuai dengan MPEG standar. Sebuah *Encoder khusus* diperlukan untuk melihat sebuah baris data SECMPEG yang tidak dienkripsi.

Algoritma Permutasi Zig-Zag

Ide dasarnya adalah daripada memetakan blok 8x8 kedalah vector 1x64 dalam urutan yang “zig-zag”, lebih baik menggunakan sebuah daftar
permutasi yang acak untuk memetakan sebuah blok 8x8 yang individual kedalam vector 1x64. Algoritma Permutasi ZigZag terdiri atas tiga langkah :

1. Membuat sebuah daftar permutasi dengan jumlah bilangan 64. Hal ini dapat dilakukan diluar algoritma.
2. Selesaikan prosedur pembagian setelah blok 8x8 selesai diquatisasi. Anggap koefisien DC dapat dimisalkan dengan 8 digit bilangan biner
d7d6d5d4d3d2d1d0. Kemudian bagi bilangan tersebut menjadi 2 buah bagian yaitu d7d6d5d4 dan d3d2d1d0, yang mana keduanya berada dalam rentang [0..15] dalam decimal. Kemudian koefisien DC diubah menjadi d7d6d5d4 dan akhir koefisien AC diubah menjadi d3d2d1d0.

Prosedur pembagian ini didasari atas obeservasi berikut:
2)Biasanya, nilai dari koefisien DC lebih besar daripada nilai dari koefisien AC, karena itu koefisien DC dapat dengan mudah dikenali bahkan setelah permutasi. Dengan membagi koefisien DC menjadi 2 buah bilangan yang lebih kecil, akan lebih sulit untuk membedakannya dari koefisien AC.
3)Setelah dibagi, sebuah ruang ekstra diperlukan untuk menyimpan bilangan yang dibagi, sebagai contoh d3d2d1d0. Ini akan menambah panjang arus MPEG. Tetapi harus diperhatikan bahwa akhir koefisien AC adalah bilangan dengan tingkat signifikan paling kecil, yang bisa diubah menjadi 0 tanpa degradasi visual yang signifikan. Jadi tempat ini bisa digunakan untuk menyimpan d3d2d1d0.
4)Kemudian masukan daftar permutasi acak kedalam blok yang telah terbagi.

Ada dua buah metode tambahan untuk mencoba meningkatkan algortima permutasi zig-zag. Metode pertama adalah dengan mengelompokkan koefisien DC dari tiap 8 blok menjadi satu dan mengaplikasikan DES pada tiap kelompok. Kemudian prosedur pembagian dan permutasi dilakukan pada tiap blok. Kedelapan koefisien DC dapat dikelompokan dengan urutan sekuensial atau acak. Metode kedua adalah dengan menggunakan urutan lemparan koin biner bersama dengan dua buah daftar permutasi. Cara ini membuat metode dasar lebih aman terhadap serangan *known-plaintext. Pembuat disarankan* untuk mengikuti tambahan berikut:
1) dua buah daftar permutasi dibuat dan
2) untuk tiap blok 8x8, sebuah koin dilempar. Bila ekor maka daftar permutasi satu dimasukan; bila kepala maka daftar permutasi dua dimasukan kedalam blok. Daftar biner lemparan koin bersama dengan kedua daftar permutasi adalah kunci rahasianya.

ALGORITMA ENKRIPSI VIDEO

Arus MPEG berbeda dengan data tekstual tradisional karena ia mempunyai tipe data yang khusus dan ia dikompresi. Perhatikan bahwa kesamaan dari kompresi dan enkripsi adalah keduanya mencoba membuang informasi yang kurang bermanfaat. Karena itu pembelajaran sturktur MPEG dan sifat statistiknya membawa kepada Algoritma Enkripsi Video.

MPEG adalah algoritma yang membuang informasi yang tidak dibutuhkan dari sekuens gambar. Ini berarti ia mempunyai distribusi nilai *byte yang lebih merata dan ia* berbeda dengan data tekstual.

Dalam VEA proses dilakukan *byte ke byte* dikarenakan :

- lebih mudah untuk memproses data dengan pengetahuan tentang *byte*;
- sebuah byte tunggal tidak berarti dalam arus video karena biasanya isi video dikodekan dalam beberapa *byte*.

Tahapan dalam algoritma ini adalah:

(i) Anggap data dari frame I adalah dalam bentuk sebagai berikut a1a2a3a4..a2n-1a2n

(ii) Pilih bytes dengan nomor ganjil dan nomor genap untuk membentuk dua buah arus bytes yang baru. Dibuat daftar ganjil dan daftar genap.

(iii) Lakukan operasi XOR pada kedua buah arus

$$\begin{array}{lcccc} & a_1 & a_3 & \ldots & a_{2n-1} \\ \text{Xor} & a_2 & a_4 & \ldots & a_{2n} \\ \hline & c_1 & c_2 & \ldots & c_n \end{array}$$

(iv) Pilih fungsi enkripsi E (contoh : DES) untuk mengenkripsi a2a4… a2n. Hasil c*ipher* dari enkripsi tersebut dapat ditulis sebagai c1c2… cnE(a2a4…a2n).