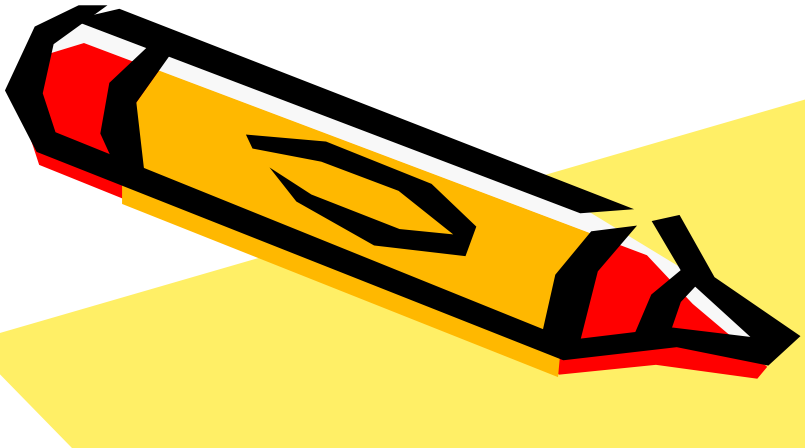


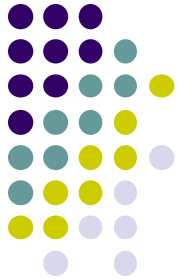
Steganografi

Bahan Kuliah IF4020 Kriptografi

Oleh: Rinaldi Munir

Program Studi Informatika, STEI-ITB

Prolog



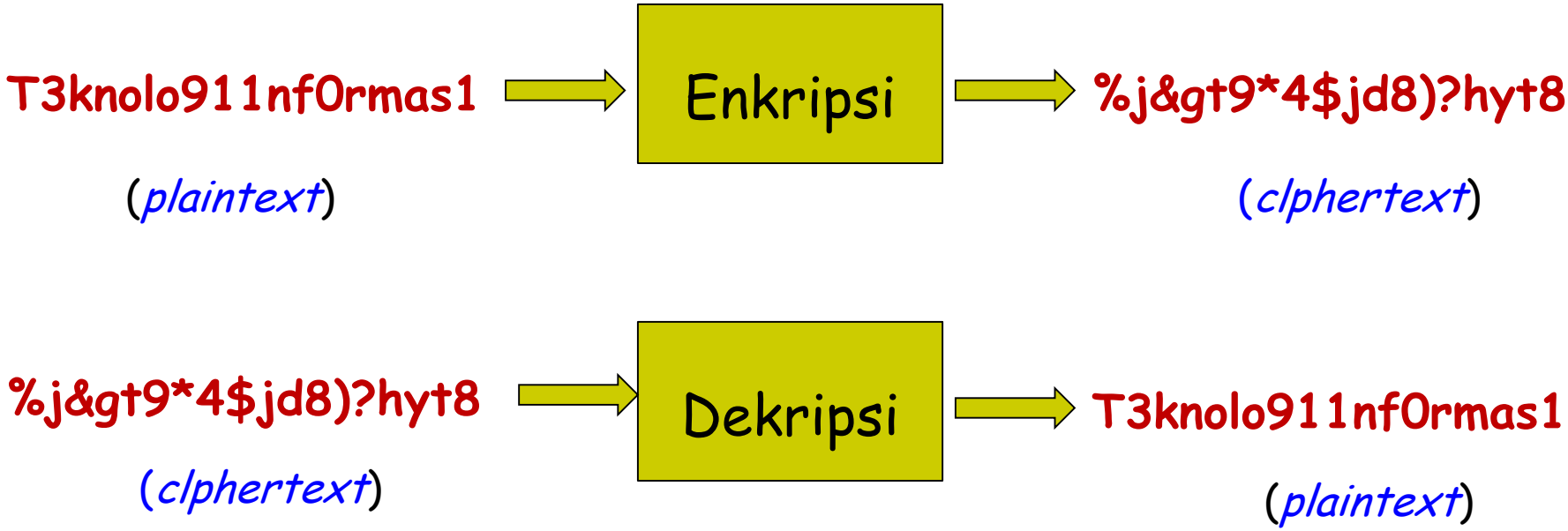
- Misalkan anda mempunyai data rahasia seperti *password*.

  Password: **T3knolo911nf0rmas1**

- Anda ingin menyimpan *password* tersebut dengan aman (tidak bisa diketahui orang lain).

- Bagaimana caranya agar *password* tersebut dapat disimpan dengan aman?

Cara I: Mengenkripsinya



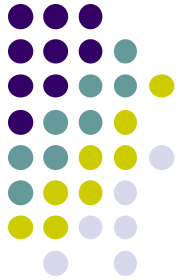
→ **Bidang KRIPTOGRAFI (*cryptography*)**

Cara II: Menyembunyikannya

T3knolo911nfOrmas1

→ Bidang STEGANOGRAFI (*steganography*)

Apa Steganografi itu?



- Dari Bahasa Yunani: steganos + graphien

  "steganos" (*στεγανός*): tersembunyi

  "graphien" (*γραφία*) : tulisan

  steganografi: tulisan tersembunyi (*covered writing*)

- ***Steganography***: ilmu dan seni menyembunyikan pesan rahasia dengan suatu cara sedemikian sehingga tidak seorang pun yang mencurigai keberadaan pesan tersebut.

Tujuan: pesan tidak terdeteksi keberadaannya

Perbedaan Kriptografi dan Steganografi

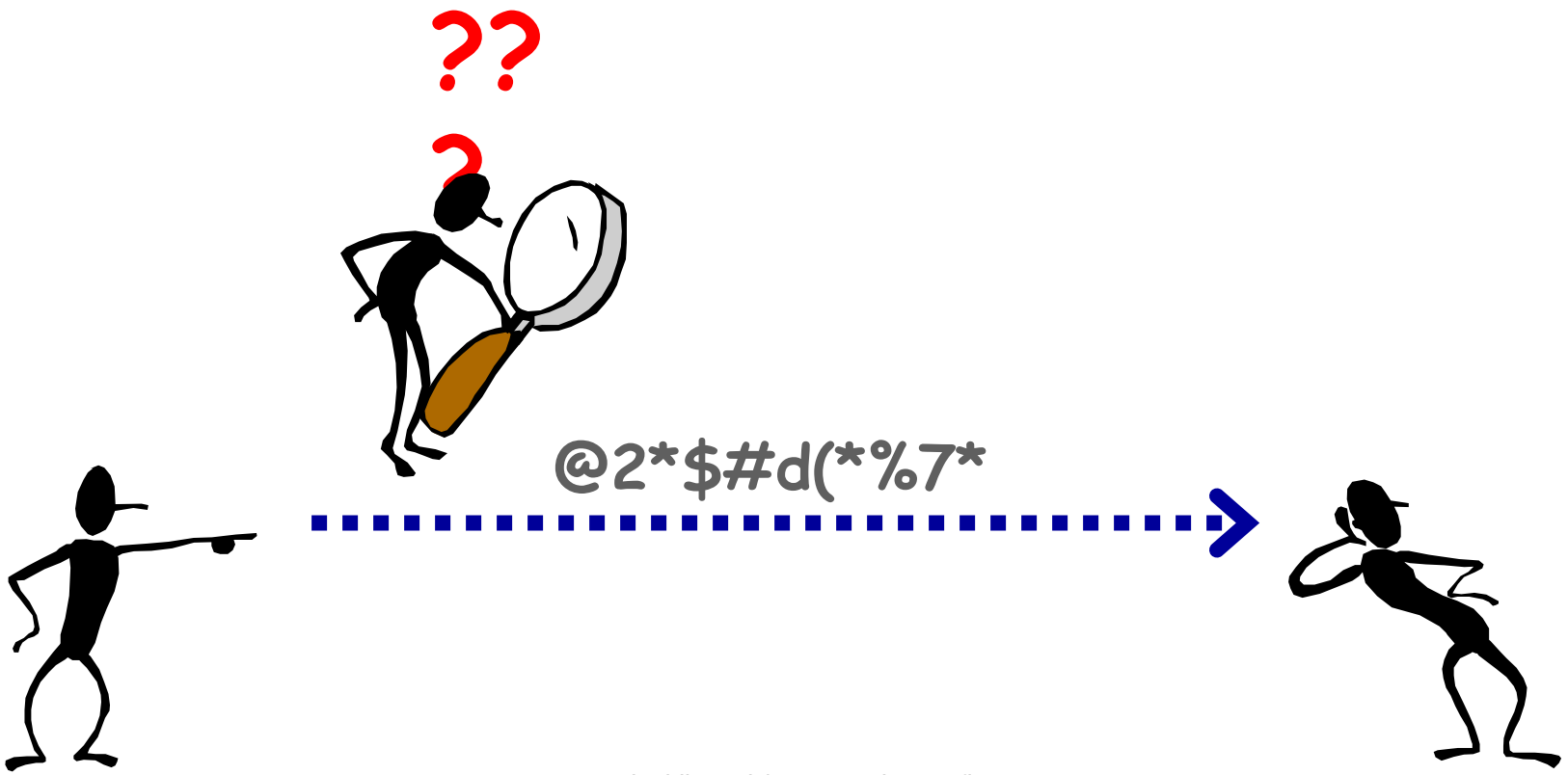
- Kriptografi: menyembunyikan ***isi*** (*content*) pesan

  → Tujuan: agar pesan tidak dapat dibaca oleh pihak ketiga (lawan)

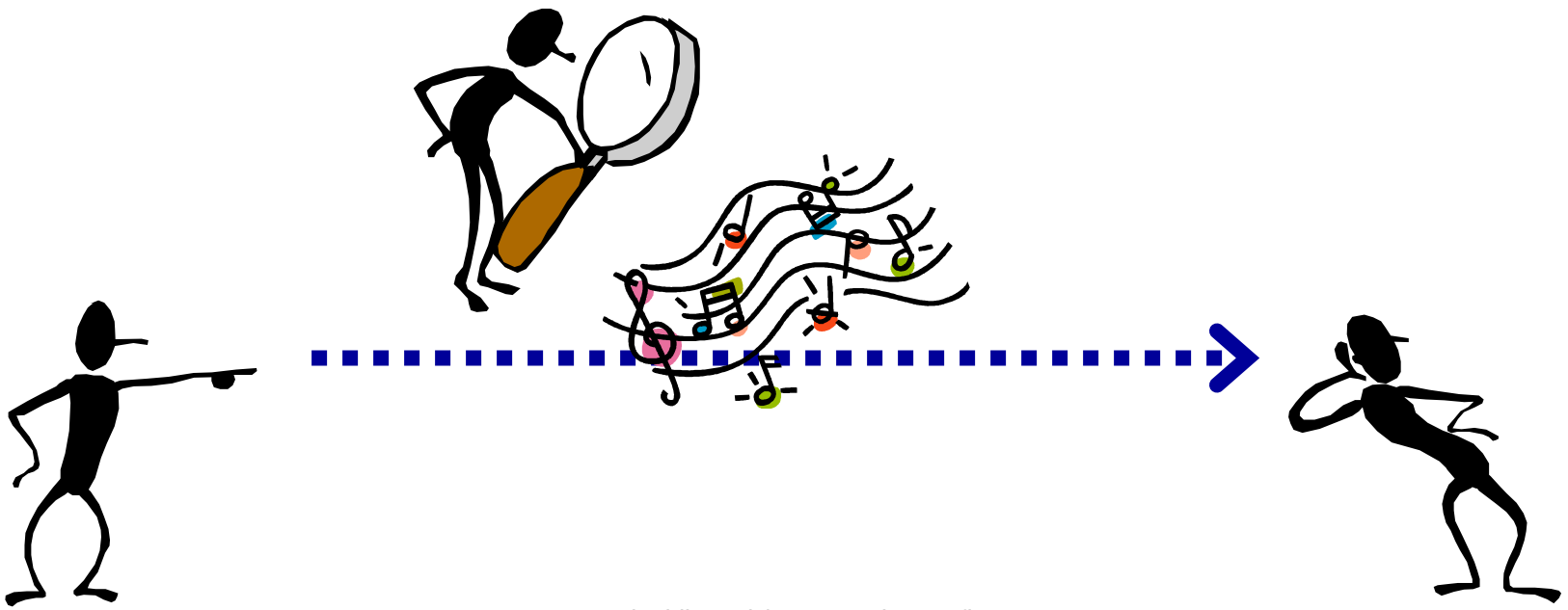
- Steganografi: menyembunyikan ***keberadaan*** (*existence*) pesan

  → Tujuan: untuk menghindari kecurigaan (*conspicuous*) dari pihak ketiga (lawan)

Kriptografi

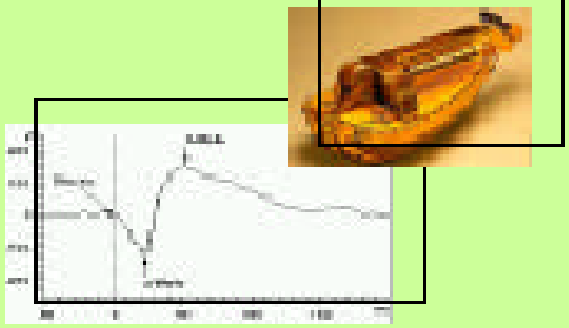


Steganografi



Stego-data are inconspicuous. Steganography will **not be detected**.

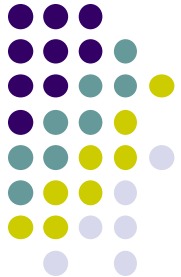
George obtains oranges yet eights' are rubbish!



Encrypted messages are conspicuous. They will be detected as ciphertext or silly data.

hIwDIwFpbAtjdf0BA/9KBX2jS17O5SRQsu2PF
caBqUXIQdyt1Fri/Wsg+eXoYsxnJI1Cn2JD7vjl
F2GH8GEr/vGQk8SQVCMyXzfPkgW0tr6RJX
AEIFF9rjnDB3kOmmVc1adrTQnLrqiC/I5r/kJs
ezowgZl82T/QVk59YsuChd+Ce8vqI/kICeqmv
w9J2amre3uxpWIOqCEQNzZyHx8HeYPf29k
Xu+uk1gekZZVdELmLD/Wa/xBKFTNUBrl+16
ewoQBxQ8+3cTXSIGPTqdzDSasgQG17Z1sr
/Lhu9qze+m64GYY0OukeiCPvhHJQuXZn2UW

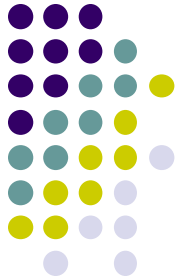
Information Hiding



- *Information hiding*: bidang ilmu yang mempelajari cara menyembunyikan pesan sehingga tidak dapat dipersepsi (baik secara visual maupun audial).

- Yang termasuk ke dalam *information hiding*:
  1. Kriptografi
  2. Steganografi

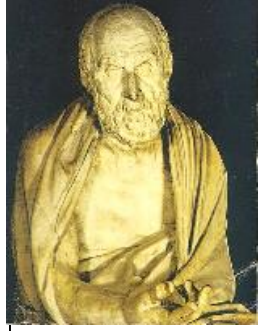
Sejarah Steganografi



- Usia steganografi setua usia kriptografi, dan sejarah keduanya berjalan bersamaan.

- Periode sejarah steganografi dapat dibagi menjadi:
  1. Steganografi kuno (*ancient steganography*)
  2. Steganografi zaman renaisans (*renaissance steganography*).
  3. Steganografi zaman perang dunia
  4. Steganografi modern

Ancient Steganography



Herodatus

- **Steganografi dengan media kepala budak.**

  Ditulis oleh Herodatus (485 – 525 BC), sejarawan Yunani pada tahun 440 BC di dalam buku: *Histories of Herodatus*). Kisah perang antara kerajaan Persia dan rakyat Yunani.

  Herodatus menceritakan cara **Histaiaeus** mengirim pesan kepada **Aristagoras of Miletus** untuk melawan Persia**.** Caranya**:** Dipilih beberapa budak. Kepala budak dibotaki, ditulisi pesan dengan cara tato, rambut budak dibiarkan tumbuh, budak dikirim. Di tempat penerima kepala budak digunduli agar pesan bisa dibaca.

- **Penggunaan *tablet wax***

  Orang-orang Yunani kuno menulis pesan rahasia di atas kayu yang kemudian ditutup dengan lilin (*wax*).

  Di dalam bukunya, Heradatus menceritakan Demaratus mengirim peringatan tentang serangan yang akan datang ke Yunani dengan menulis langsung pada tablet kayu yang kemudian dilapisi lilin dari lebah.

- **Penggunaan tinta tak-tampak (*invisible ink*)**

Pliny the Elder.
AD 23 - 79

Pliny the Elder menjelaskan penggunaan tinta dari getah tanaman *thithymallus*. Jika dituliskan pada kertas maka tulisan dengan tinta tersebut tidak kelihatan, tetapi bila kertas dipanaskan berubah menjadi gelap/coklat

The **Ancient Chinese** wrote notes on small pieces of silk that they then wadded into little balls and coated in wax, to be swallowed by a messenger and retrieved at the messenger's gastrointestinal convenience.

- **Penggunaan kain sutra dan lilin**

- Orang Cina kuno menulis catatan pada potongan-potongan kecil sutra yang kemudian digumpalkan menjadi bola kecil dan dilapisi lilin.

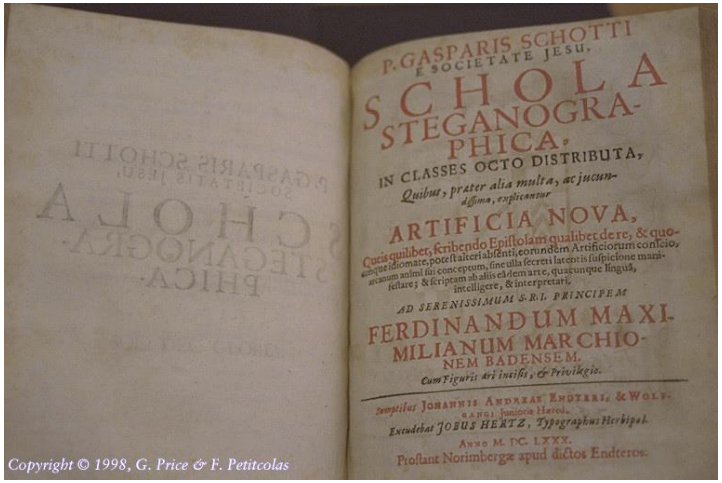
- Selanjutnya bola kecil tersebut ditelan oleh si pembawa pesan.

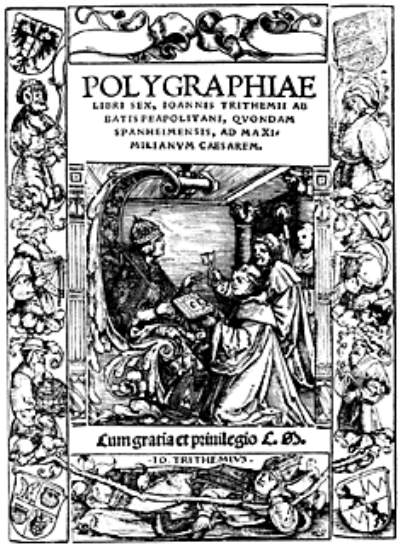
- Pesan dibaca setelah bola kecil dikeluarkan dari perut si pembawa pesan.

Renaissance Steganography

Johannes Trithemius (1404-1472)

Tahun 1499, Johannes Trithemius menulis buku **Steganographia**, yang menceritakan tentang metode steganografi berbasis karakter


P. GASPARIS SCHOTTI
E SOCIETATE JESU,
SCHOLA STEGANOGRAPHICA,
IN CLASSES OCTO DISTRIBUTA,
Quibus, prater alia multa, ac jucundissima, explicantur
ARTIFICIA NOVA,
AD SERENISSIMUM S.R.I. PRINCIPEM
FERDINANDUM MAXIMILIANUM MARCHIONEM BADENSEM.
Cum Figuris æri insitis, & Privilegio.
Excudebat JOSUS HERTZ, Typographus Herbipol.
Anno M. DC. LXXX.
Prostant Norimbergæ apud dictos Endteros.


POLYGRAPHIAE
LIBRI SEX, IOANNIS TRITHEMII AB BATIS FEAPOLITANI, QVONDAM SPANHEIMENSIS, AD MAXIMILIANVM CAESAREM.
Cum gratia et privilegio C. M.
IO. TRITHEMIVS.

Selanjutnya tahun 1518 dia menulis buku tentang steganografi dan kriptografi, Berjudul **Polygraphiae**

Giovanni Battista Porta
(1535-1615)

Giovanni Battista Porta menggambarkan cara menyembunyikan pesan di dalam telur rebus.

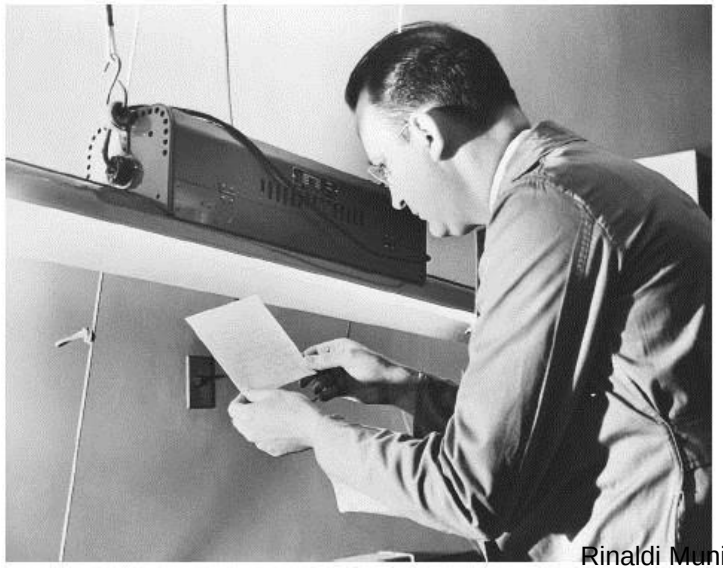
Caranya, pesan ditulis pada kulit telur yang dibuat dari tinta khusus yang dibuat dengan satu ons tawas dan setengah liter cuka.

Prinsipnya penyembunyiannya adalah tinta tersebut akan menembus kulit telur yang berpori, tanpa meninggalkan jejak yang terlihat.

Tulisan dari tinta akan membekas pada permukaan isi telur yang telah mengeras (karena sudah direbus sebelumnya). Pesan dibaca dengan membuang kulit telur

World War Steganography

- Penggunaan tinta tak-tampak (*invisible ink*) dalam spionase.
  - Pada Perang Dunia II, tinta tak-tampak digunakan untuk menulis pesan rahasia
  - Tinta terbuat dari campuran susu, sari buah, cuka, dan urine.
  - Cara membaca: Kertas dipanaskan sehingga tulisan dari tinta tak-tampak tersebut akan menghitam.



Seorang agen FBI sedan menggunakan sinar ultraviolet untuk membaca tulisan yang tersembunyi pada kertas yang dicurigai dari agen spionase.

- Steganografi dalam Perang Dunia II: *Null Cipher*

  Pesan berikut dikirim oleh Kedubes Jerman pada PD II:

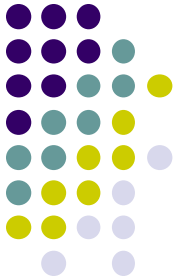
  Apparently neutral's protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretext for embargo on by-products, ejecting suets and vegetable oils.

  Ambil huruf kedua setiap kata, diperoleh pesan berikut:
  Pershing sails from NY June 1.

Contoh *Null Cipher* lainnya:

Big rumble in New Guinea.
The war on celebrity acts should end soon. Over four die ecstatic elephants replicated.

Bring two cases of deer.

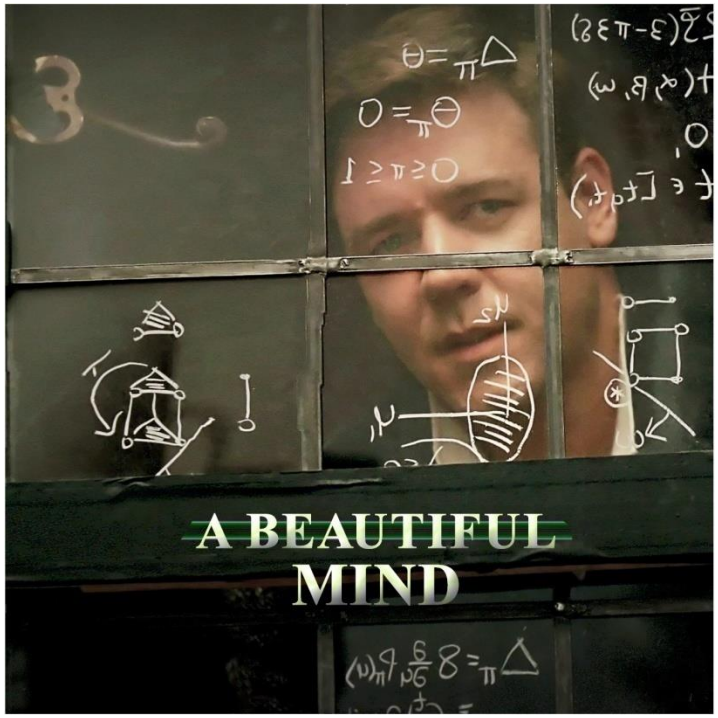


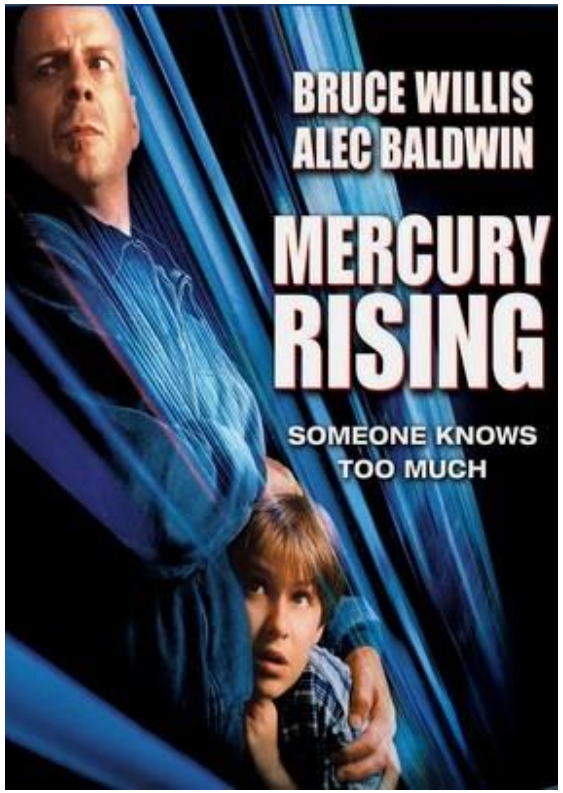
Fishing freshwater bends and saltwater coasts rewards anyone feeling stressed. Resourceful anglers usually find masterful leapers fun and admit swordfish rank overwhelming anyday.

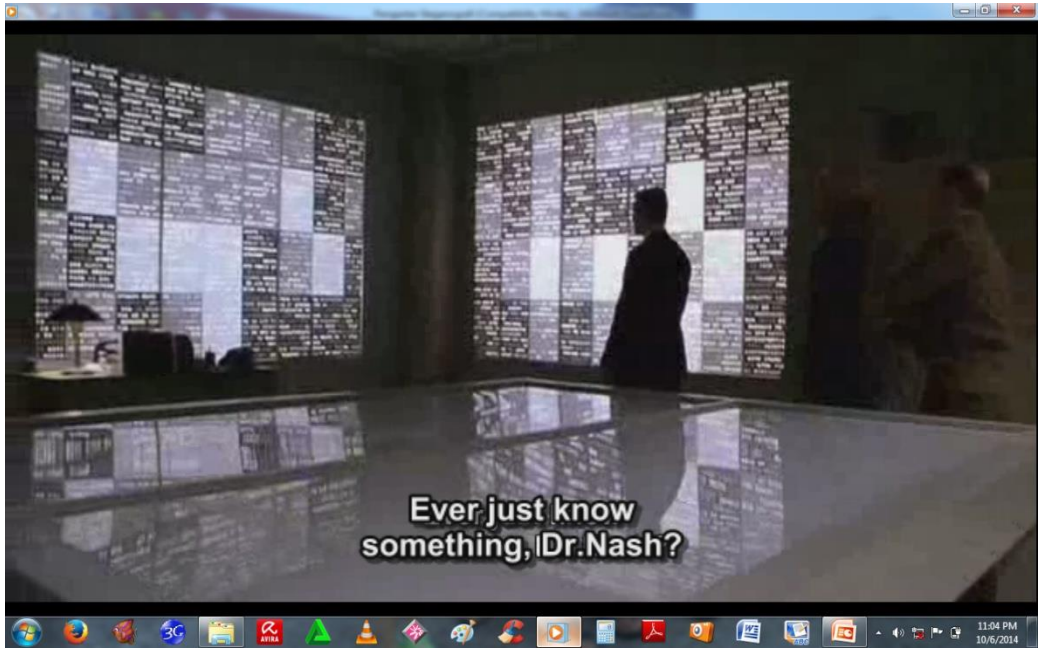
Dengan mengambil huruf ketiga pada setiap kata diperoleh pesan berikut:

Send Lawyers, Guns, and Money.

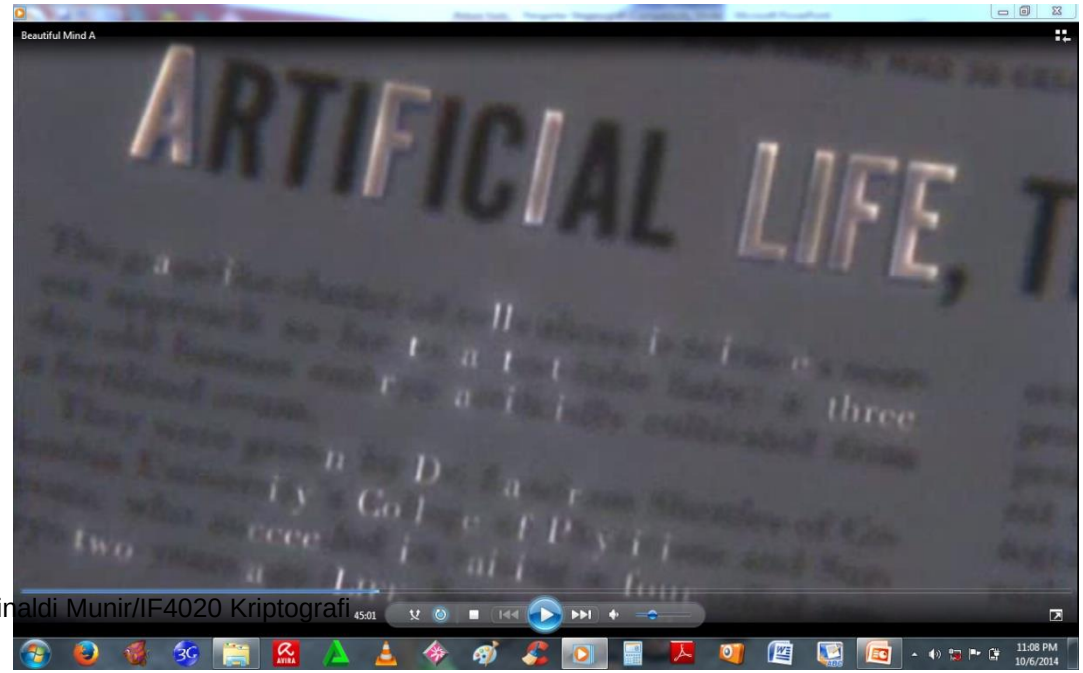
- Steganografi di dalam film *Mercury Rising* dan *Beautiful Mind*







Beberapa adegan film *Beautiful Mind* yang memperlihatkan steganografi



http://www.randomhouse.com/doubleday/davinci/

While in Paris on business, Harvard symbologist Robert Langdon receives an urgent late-night phone call. The elderly curator of the Louvre has been murdered inside the museum, a baffling cipher found near the body. As Langdon and a gifted French cryptologist, Sophie Neveu, sort through the bizarre riddles, they are stunned to discover a trail of clues hidden in the works of Da Vinci—clues visible for all to see and yet ingeniously disguised by the painter.

The stakes are raised when Langdon uncovers a startling link: The late curator was involved in the Priory of Sion—an actual secret society whose members included Sir Isaac Newton, Botticelli, Victor Hugo, and Da Vinci, among others. Langdon suspects they are on the hunt for a breathtaking historical secret, one that has proven through the centuries to be as enlightening as it is dangerous. In a frantic race through Paris, and beyond,

(continued on back flap)

Langdon and Neveu find themselves matching wits with a faceless powerbroker who appears to anticipate their every move. Unless they can decipher the labyrinthine puzzle, the Priory's secret—and an explosive ancient truth—will be lost forever.

Breaking the mold of traditional suspense novels, *The Da Vinci Code* is simultaneously lightning-paced, intelligent, and intricately layered with remarkable research and detail. From the opening pages to the unpredictable and stunning conclusion, bestselling author Dan Brown proves himself a master storyteller.

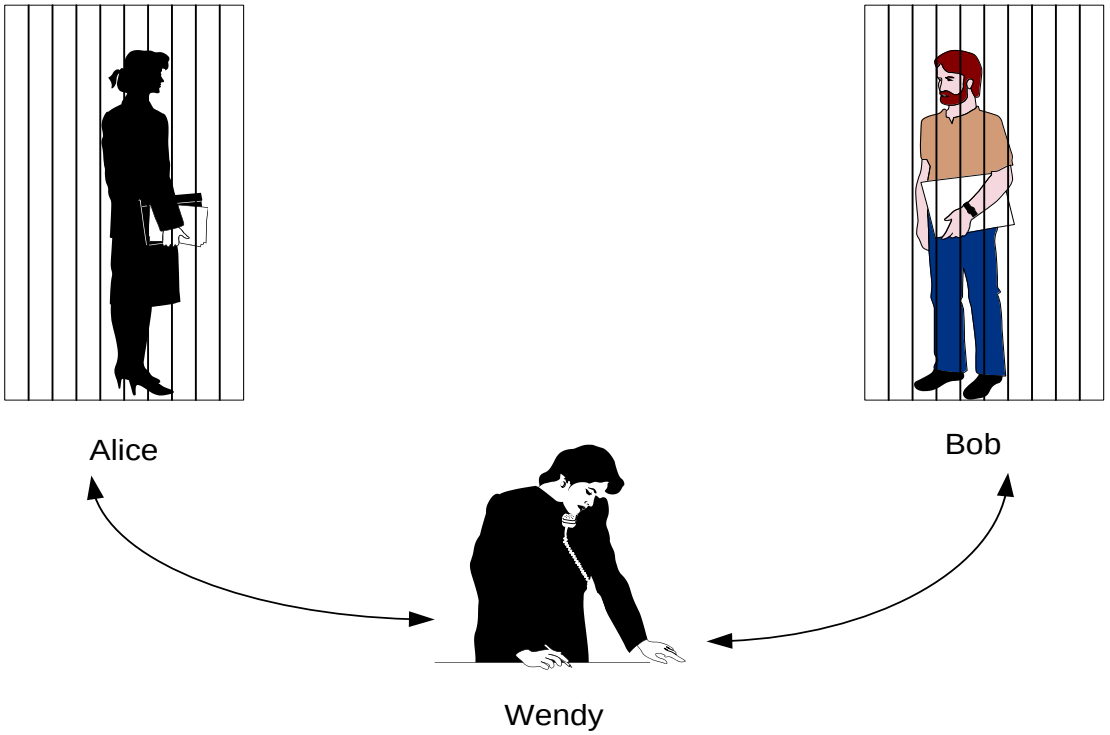
Sumber: http://budi.paume.itb.ac.id

Steganografi dan Terorisme

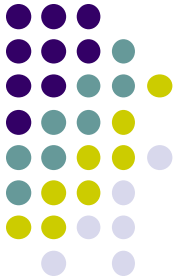
- Ilmu steganografi mendadak naik daun ketika pasca 11 September 2001 pihak FBI menuding *Al-Qaidah* menggunakan steganografi untuk menyisipkan pesan rahasia melalui video atau gambar yang mereka rilis secara teratur di Internet.

Steganografi Modern - *The Prisoner's Problem*

- Diperkenalkan oleh Simmons – 1983
- Dilakukan dalam konteks *USA – USSR nuclear non-proliferation treaty compliance checking*



Pesan rahasia: "malam ini kita kabur"

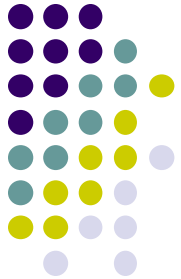


- Bagaimana cara Bob mengirim pesan rahasia kepada Alice tanpa diketahui oleh Wendy?

- Alternatif 1: mengenkripsinya

xjT#9uvmY!rc$7yt59hth@#

Wendy pasti curiga!



- Alternatif 2: menyembunyikannya di dalam tulisan lain

  masihkah ada lara apabila memoriku ingat nestapa itu. kita ingin tetap abadikan kisah asmara. bersamamu usiaku renta.

  Wendy tidak akan curiga!

Information hiding dengan steganografi!

Steganografi Digital

- Steganografi digital: penyembunyian pesan digital di dalam dokumen digital lainnya.
- *Carrier file*: dokumen digital yang digunakan sebagai media untuk menyembunyikan pesan.

1. Teks

"Kita semua bersaudara"

-Txt
- doc
- html

2. Audio

- wav
- mp3

3. Gambar (*image*)

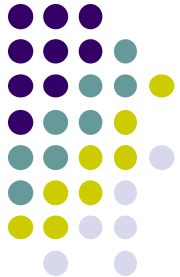
- bmp
- jpeg
- gif

4. Video

-Mpeg
- avi
- dll


myhiddenmessage.txt ...
File Edit Format Help
My hidden message.
Carrier File
Carrier File with
Hidden Message

Terminologi Steganografi



1. *Embedded message* (*hiddentext) atau secret message*: pesan yang disembunyikan .
   Bisa berupa teks, gambar, audio, video, dll

2. *Cover-object* (*covertext):* pesan yang digunakan untuk menyembunyikan *embedded message*.
   Bisa berupa teks, gambar, audio, video, dll

2. *Stego-object* (*stegotext):* pesan yang sudah berisi pesan *embedded message*.

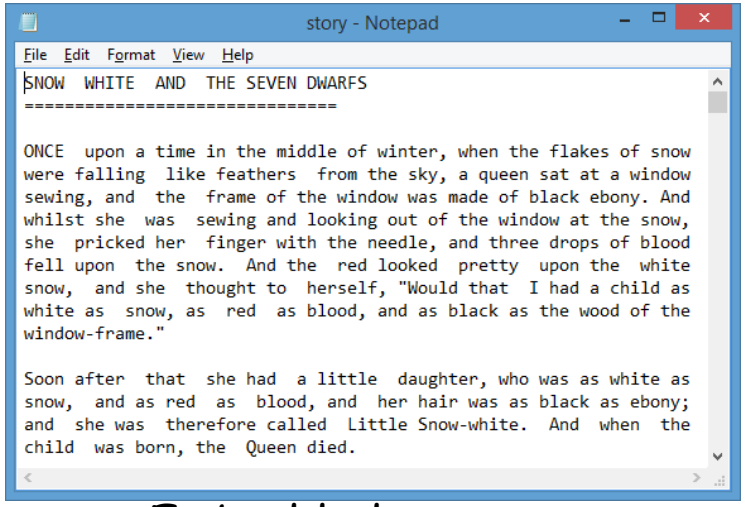
3. *Stego-key*: kunci yang digunakan untuk menyisipan pesan dan mengekstraksi pesan dari stegotext.

Istilah keilmuan serumpun terasa memberikan distorsi persepsi pada maksud sebenarnya. Persepsi yang segera terbentuk dengan istilah tesrebut adalah pertumbuhan dari akar-akar ilmu membentuk suatu rumpun, yang berarti bahwa nuansa historis organisasi/kelompok/unit yang mewadahinya.		
Embedded message	*Cover-image*	*Stego-image*

GIF image

Cover image

Stego-image



Embedded message

Cover image

Stego-image

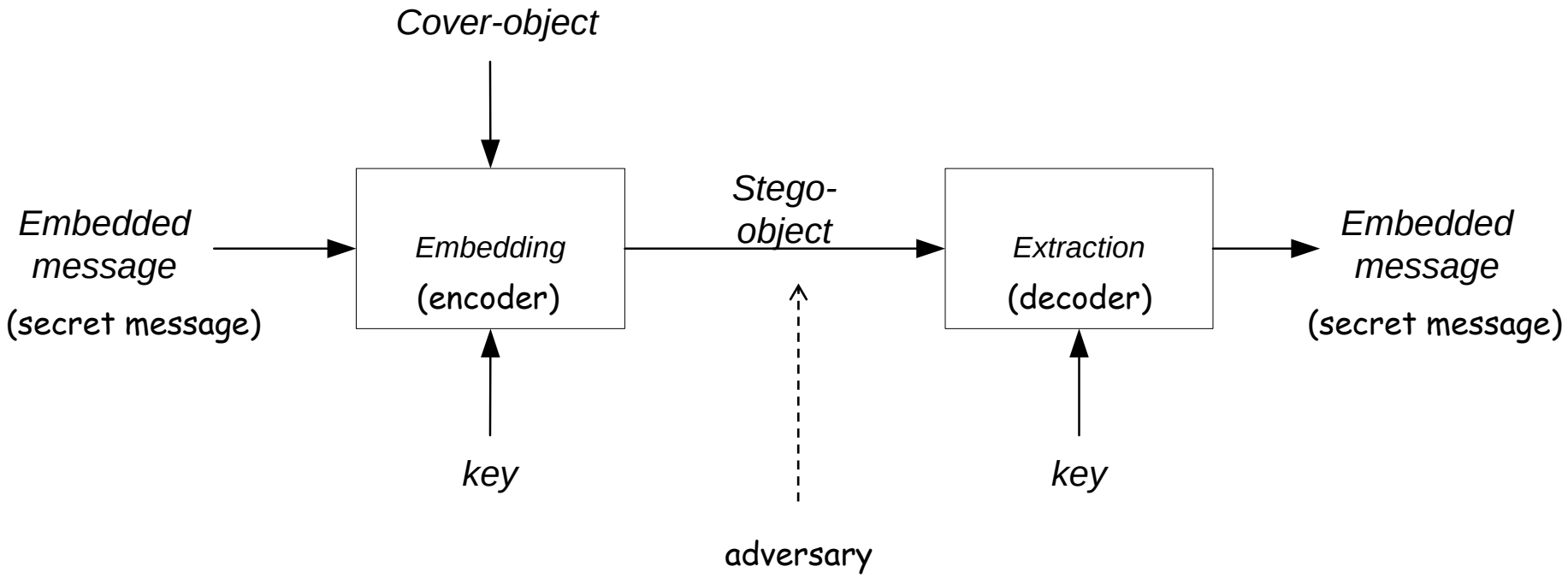
Cover image

Embedded image

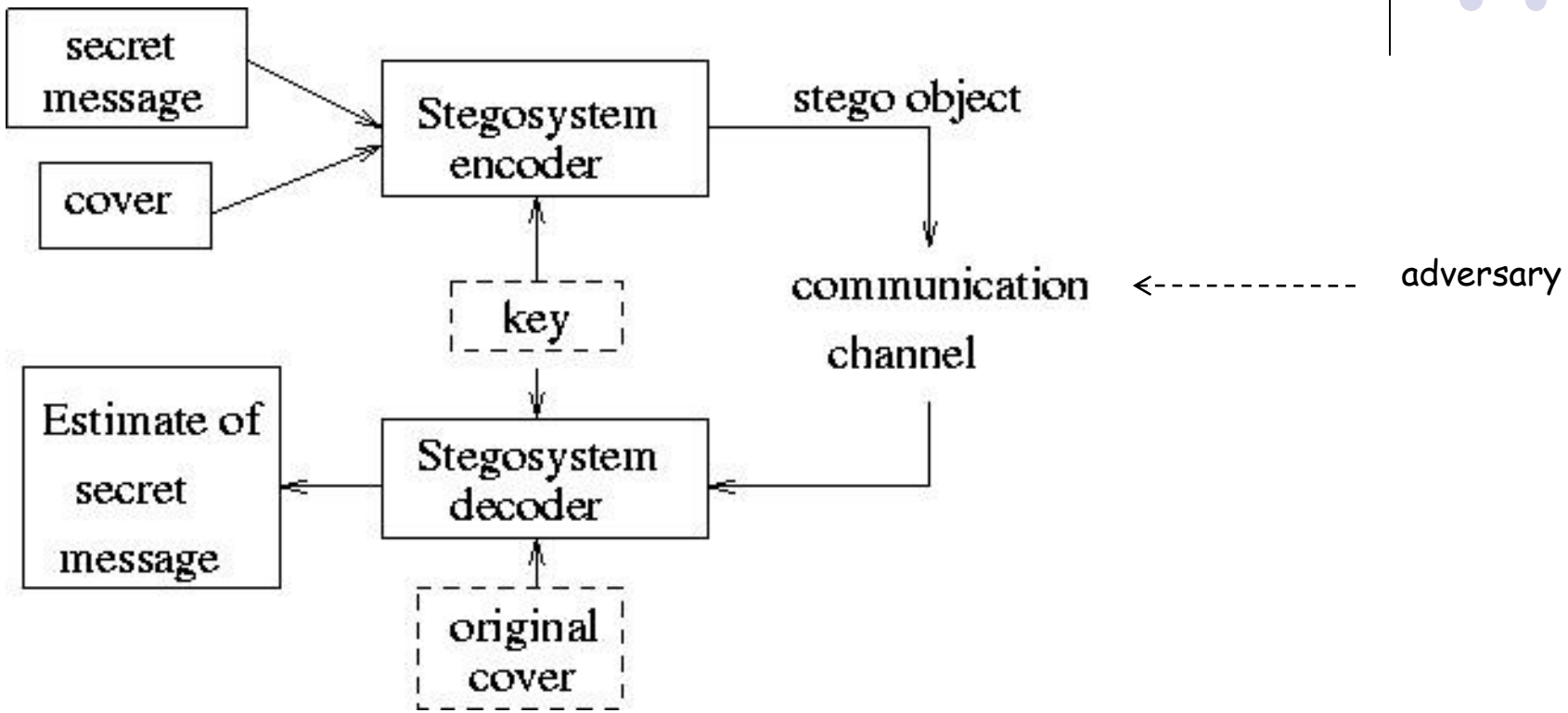
Stego-image

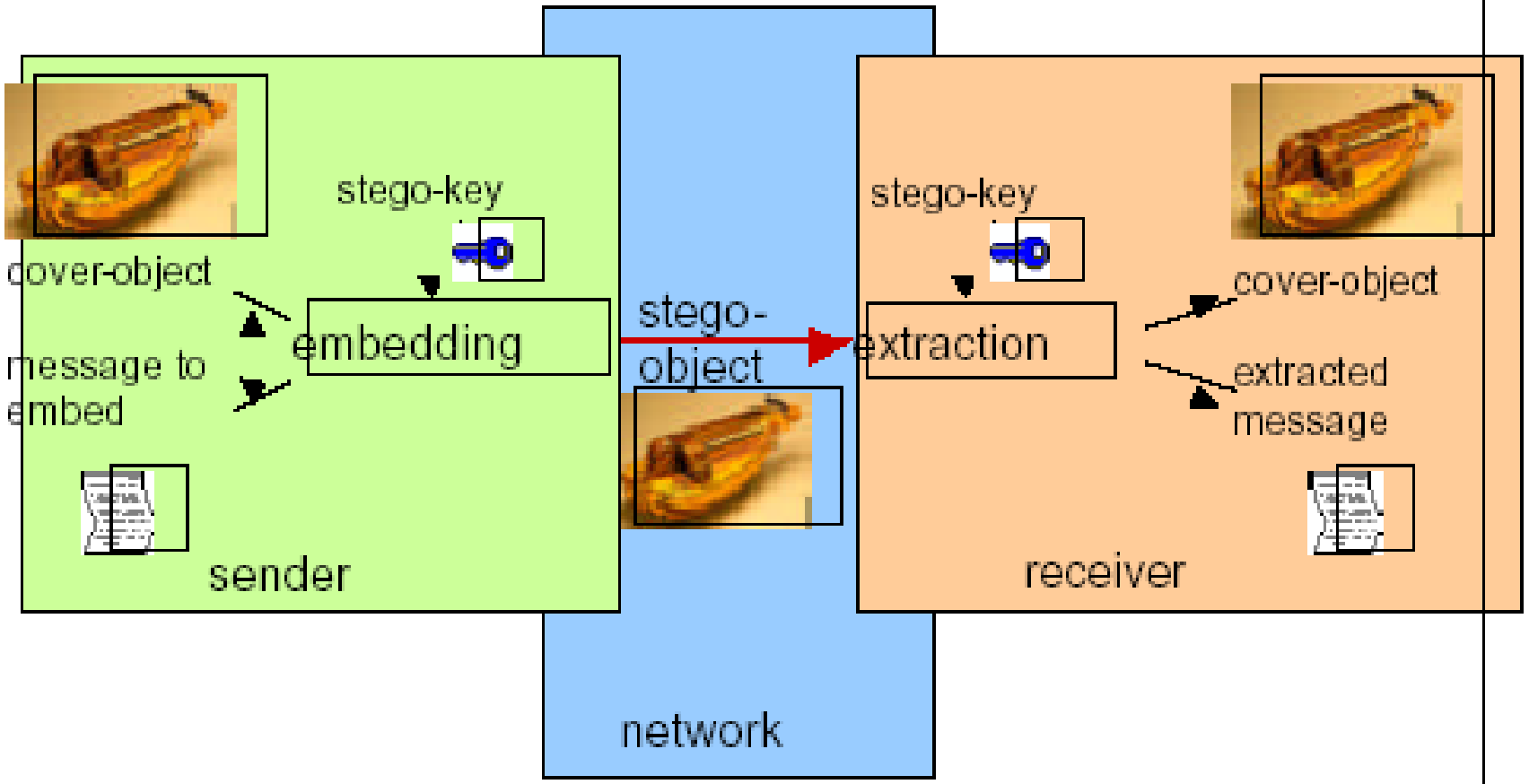
Extracted image

Diagram Proses Steganografi



Stego-object umumnya dikirim melalui saluran komunikasi:




stego-key
cover-object
embedding
message to
embed
sender
stego-
object
network
stego-key
extraction
cover-object
extracted
message
receiver

Kriteria Steganografi yang Bagus

1. *Imperceptible*

   Keberadaan pesan rahasia tidak dapat dipersepsi secara visual atau secara audio (untuk *stego-audio*).

2. *Fidelity*.

   Kualitas *cover-object* tidak jauh berubah akibat penyisipan pesan rahasia.

3. *Recovery*.

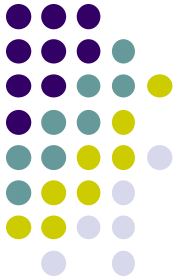
   Pesan yang disembunyikan harus dapat diekstraksi kembali.

4. *Capacity*

   Ukuran pesan yang disembunyikan sedapat mungkin besar

Catatan: *Robustness* bukan isu penting di dalam steganografi

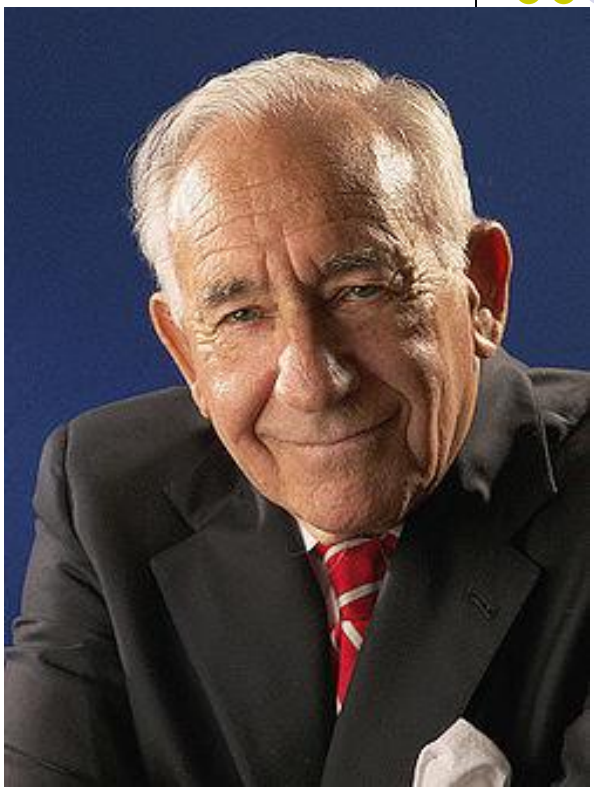
Kombinasi Kriptografi dan Steganografi



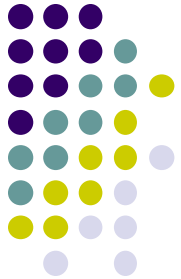
- Steganografi bukan pengganti kriptografi, tetapi keduanya saling melengkapi.
- Keamanan pesan rahasia dapat ditingkatkan dengan menggabungkan kriptografi dan steganografi.
- Mula-mula pesan dienkripsi dengan algoritma I kriptografi.
- Selanjutnya pesan terenkripsi disembunyikan di dalam media lain (citra, video, audio, dll).

Steganography has its place in security. It is not intended to replace cryptography but supplement it. Hiding a message with steganography methods reduces the chance of a message being detected. However, if that message is also encrypted, if discovered, it must also be cracked (yet another layer of protection).

(David Kahn, penulis buku *The Codebreakers - The Story of Secret Writing*)



Tiga Tipe Steganografi



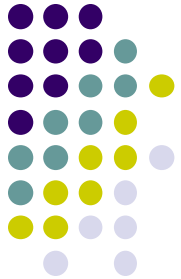
1. ***Pure steganography***

   Tidak membutuhkan kunci sama sekali. Keamanan steganografi seluruhnya bergantung pada algoritmanya.

   Contoh: *Null Cipher*

   Prinsip Kerkhoff juga seharusnya pada steganografi, bahwa keamanan sistem seharusnya tidak didasarkan pada kerahasaiaan algoritma embedding, tetapi pada kuncinya.

   Pure steganography →tidak disukai



2. ***Secret (or symmetric) key Steganography***

   Menggunakan kunci yang sama untuk *embedding* dan *extraction*.

   Contoh: - kunci untuk pembangkitan bilangan acak

   - kunci untuk mengenkripsi pesan dengan algoritma kriptografi simetri (DES, AES, dll)

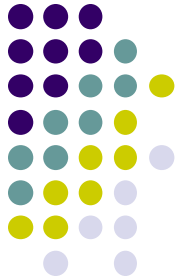
3. ***Public-key Steganography***

   Menggunakan dua kunci: kunci publik untuk *embedding* dan kunci privat untuk *extraction.*

   Contoh: kunci publik RSA untuk mengenkripsi *hidden message*

   kunci privat RSA untuk mendekripsi *hidden message*

Ranah Steganografi



Berdasarkan ranah operasinya, metode-metode steganografi dapat dibagi menjadi dua kelompok

- *Spatial (time) domain methods*

  Memodifikasi langsung nilai *byte* dari *cover-object* (nilai *byte* dapat merepresentasikan intensitas/warna *pixel* atau amplitudo)

  Contoh: Metode modifikasi *LSB*

- *Tranform domain methods*

  Memodifikasi hasil transformasi sinyal dalam ranah transform (hasil trnasformasi dari ranah spasial ke ranah lain (misalnya ranah frekuensi).

  Contoh: Metode *Spread Spectrum*

Teknik Dasar dalam Steganografi

- Substitution techniques : mengganti bagian yang redundan dari cover-object dengan pesan rahasia.

  Contoh: metode modifikasi LSB

- Transform domain techniques : menyisipkan pesan rahasia ke dalam sinyal dalam ranah *transform* (misalnya dalam ranah frekuensi).

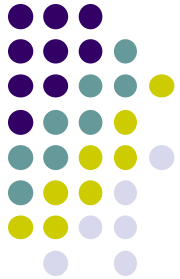
- Spread spectrum techniques : menyisipkan pesan rahasia dengan mengadopsi ide komunikasi *spread spectrum.*

- Statistical techniques : menyisipkan pesan dengan mengubah beberapa properti statistik dari *cover-object* dan menggunakan metode uji hipotesis pada proses ekstraksi pesan.

- Distortion techniques : menyimpan pesan rahasia dengan distorsi sinyal dan mengukur deviasinya dari *cover-object* pada proses ekstraksi pesan.

- Cover generation techniques : tidak menyisipkan pesan pada *cover-object* yang dipilih secara acak, tetapi membangkitkan *cover* yang cocok untuk pesan yang disembunyikan.

Program Stegano *shareware*



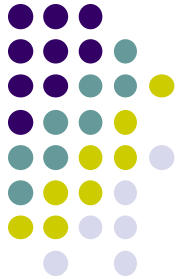
1. InPlainView:
   http://www.simtel.net/product.php%5Bid%5D12796%5BSiteID%5Dsimtel.net

   Keterangan: hanya untuk citra .bmp

2. S-tools

   http://digitalforensics.champlain.edu/download/s-tools4.zip

   Keterangan: untuk citra GIF dan BMP.



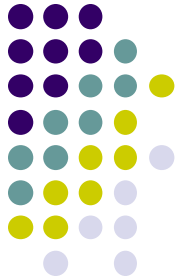
- Daftar 100 kakas steganografi lainnya:

  http://www.jjtc.com/Steganography/toolmatrix.htm

- Beberapa diantaranta berjalan di Linux:
  1. **JPHS (JPHide JPSeek, JP hide and seek)**

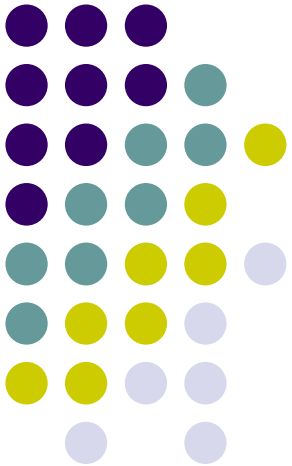
     http://linux01.gwdg.de/~alatham/stego.html

  2. Steghide
  3. Outguess
  4. Blindside
  5. Gifshuffle
  6. GzSteg
  7. dll



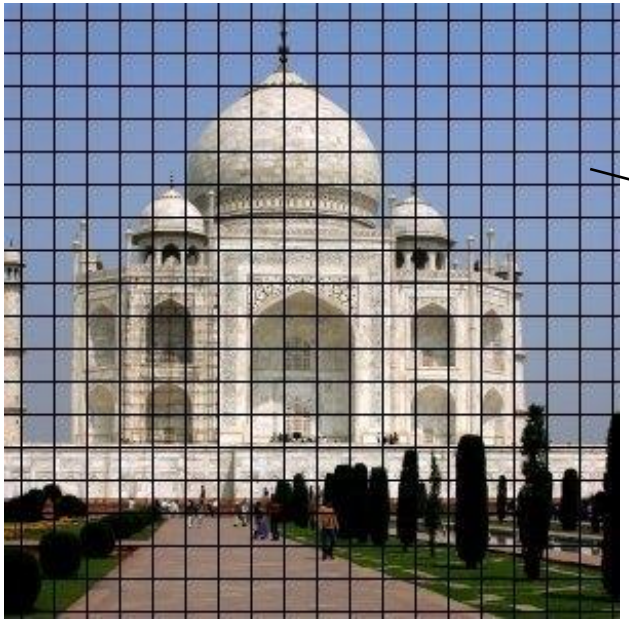
- Situs populer untuk informasi steganografi
  - **http://www.ise.gmu.edu/~njohnson/Steganography**
  - **http://www.rhetoric.umn.edu/Rhetoric/misc/dfrank/stegsoft.html**
  - **http://www.topology.org/crypto.html**
  - http://mozaiq.org/

Metode LSB



Citra Digital

- Citra terdiri dari sejumlah *pixel.* Citra 1200 x 1500 berarti memiliki 1200 x 1500 pixel = 1.800.000 pixel



- Setiap *pixel* panjangnya *n*-bit.

  Citra biner → 1 bit/pixel

  Citra *grayscale* → 8 bit/pixel

  Citra *true color* → 24 bit/pixel

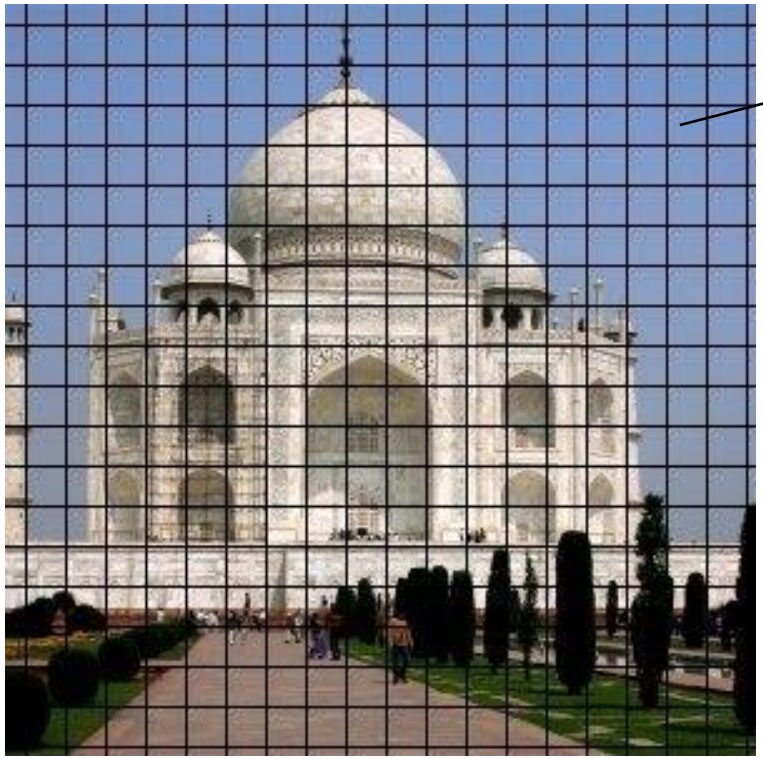
Citra Lenna

True color image
(24-bit)

Grayscale image
(8-bit)

Bimary image
(1-bit)

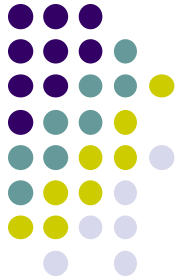
Pada citra 24-bit (*real image*), 1 pixel = 24 bit, terdiri dari komponen RGB (Red-Green-Blue)



100100111001010010001010

R G B

Bitplane pada Citra Digital



- Nilai *pixel* pada koordinat (*x*, *y*) menyatakan intensitas nilai keabuan pada posisi tersebut.

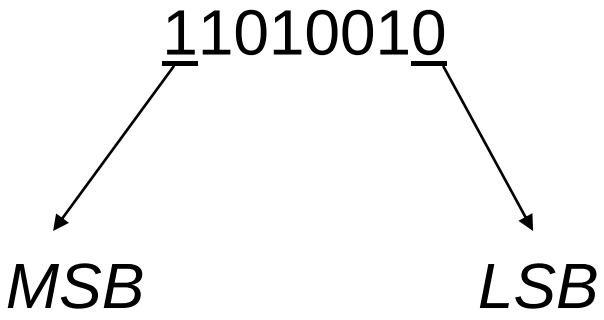
- Pada citra *grayscale* nilai keabuan itu dinyatakan dalam *integer* berukuran 1 *byte* sehingga rentang nilainya antara 0 sampai 255.

- Pada citra berwarna 24-bit setiap *pixel* tediri atas kanal *red*, *green*, dan *blue* (*RGB*) sehingga setiap *pixel* berukuran 3 *byte* (24 bit).

- Di dalam setiap *byte* bit-bitnya tersusun dari kiri ke kanan dalam urutan yang kurang berarti (*least significant bits* atau *LSB*) hingga bit-bit yang berarti (*most significant bits* atau *MSB*).

- Susunan bit pada setiap *byte* adalah $b_8b_7b_6b_5b_4b_3b_2b_1$.

  Contoh:



LSB = Least Significant Bit
MSB = Most Significant Bit

- Jika setiap bit ke-*i* dari *MSB* ke *LSB* pada setiap *pixel* diekstrak dan diplot ke dalam setiap *bitplane image* maka diperoleh delapan buah citra biner.

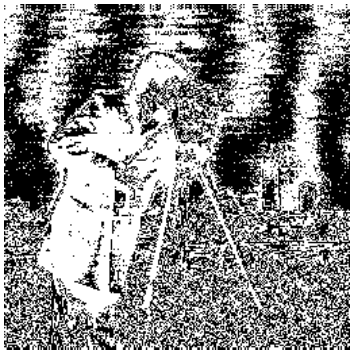
Original image

Bitplane 7

Bitplane 6

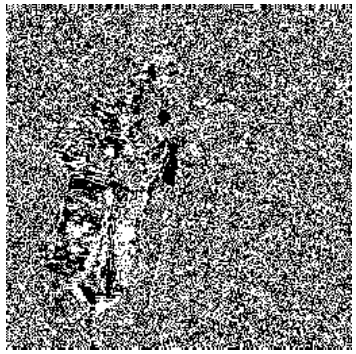
Bitplane 5

Bitplane 4

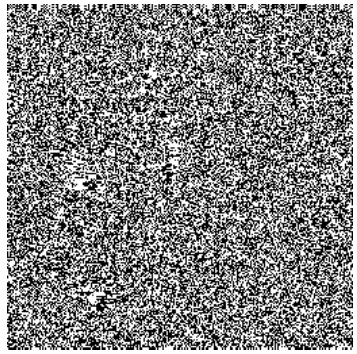


Bitplane 3

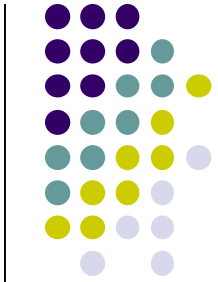
Bitplane 2



Bitplane 1

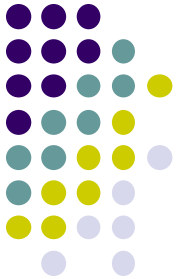


Bitplane 0



- *Bitpline* LSB, yaitu *bitplane* 0, terlihat seperti citra acak (*random image*).
- Bitpline LSB merupakan bagian yang redundan pada citra.
- Artinya, perubahan nilai bit pada bagian tersebut tidak mengubah persepsi citra secara secara keseluruhan.
- Inilah yang mendasari metode steganografi yang paling sederhana, yaitu **metode modifikasi LSB**.

Metode Modifikasi *LSB*



- Merupakan metode steganografi yang paling populer.
- Memanfaatkan kelemahan indra visual manusia dalam mengamati perubahan sedikit pada gambar
- Caranya: Mengganti bit *LSB* dari *pixel* dengan bit pesan.

  Mengubah bit *LSB* hanya mengubah nilai *byte* satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya → tidak berpengaruh terhadap persepsi visual/auditori.

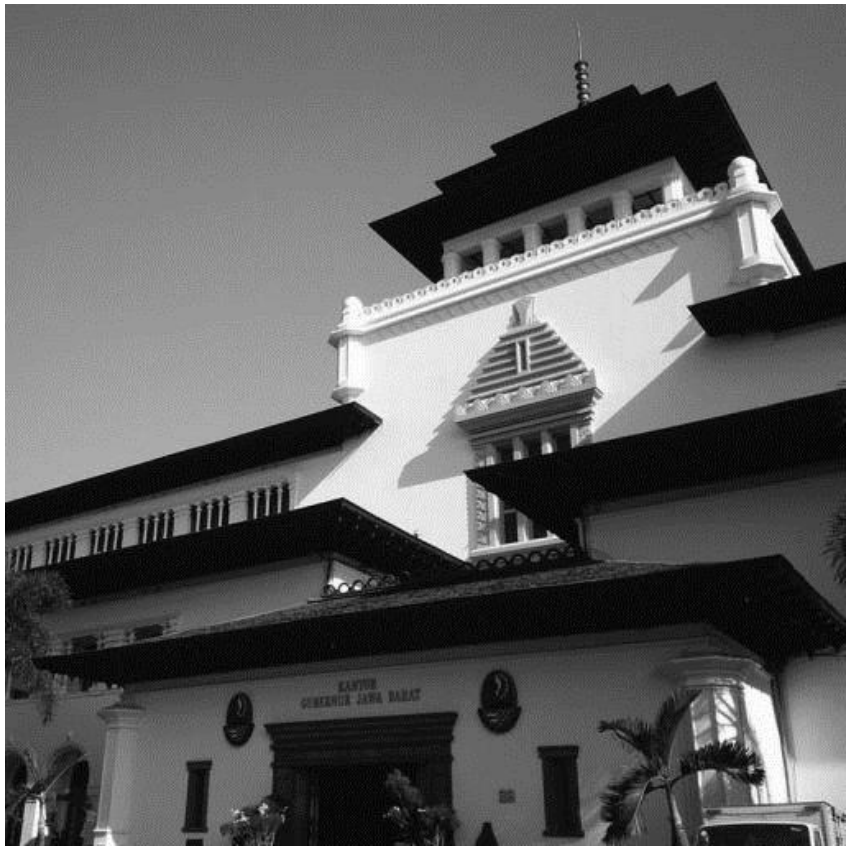
Misalkan semua bit LSB pada citra berwarna dibalikkan
Dari semula 0 menjadi 1; dari semula 1 menjadi 0

Sebelum

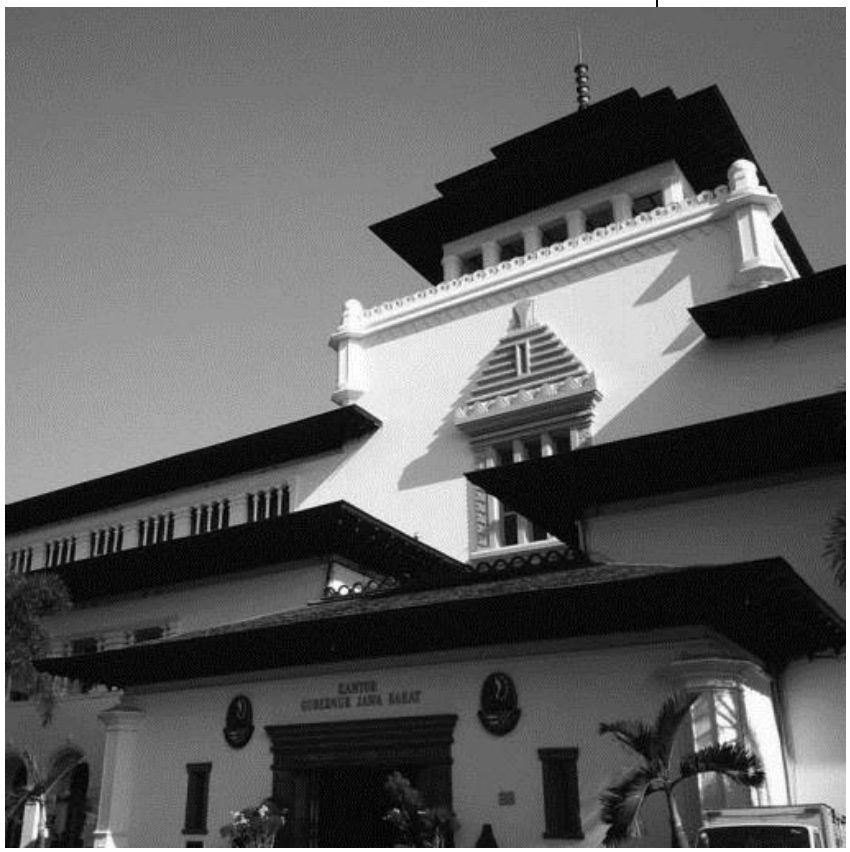
Sesudah

Adakah perbedaaanya?

Misalkan semua bit LSB pada citra *grayscale* dibalikkan
Dari semula 0 menjadi 1; dari semula 1 menjadi 0



Sebelum



Sesudah

Adakah perbedaaanya?

Contoh 1:

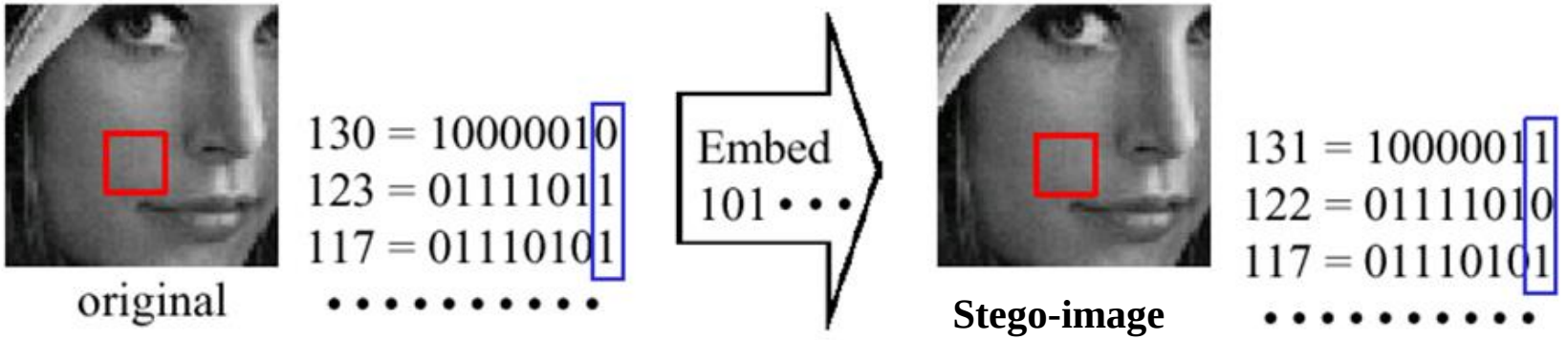
- Tinjau 1 buah *pixel* dari citra 24-bit (3 x 8 bit):

  10000010 01111011 01110101

  (130) (123) (117)

- Bit bit-bit *embedded message*: 101

- *Embed*: 00110011 10100010 11100011



Original: misalkan *pixel* [130, 123, 117] berwarna "ungu"
Stego-image: *pixel* [131, 122, 117] tetap "ungu" tapi berubah sangat sedikit.
Mata manusia tidak dapat membedakan perubahan warna yang sangat kecil.

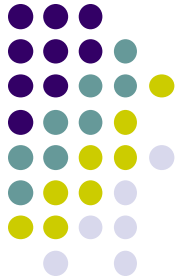
Pergeseran warna sebesar 1 dari 256 warna tidak dapat dilihat oleh manusia

PESAN RAHASIA :
KETEMUAN Di STASIUN KA MALAM JAM 13.00

01101111

Sumber: TA Yulie Anneria Sinaga 13504085

Contoh 2:



- Jika pesan = 10 bit, maka jumlah *byte* yang digunakan = 10 *byte*

- Contoh susunan *byte* yang lebih panjang:

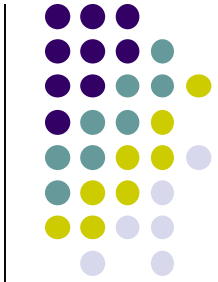
  00110011 10100010 11100010 10101011 00100110
  10010110 11001001 11111001 10001000 10100011

- Pesan: 1110010111

- Hasil penyisipan pada bit *LSB*:

  00110011 10100011 11100011 10101010 00100110
  10010111 11001000 11111001 10001001 10100011

Ekstraksi Pesan dari *Stego-object*



- Bit-bit pesan yang disembunyikan di dalam citra harus dapat diekstraksi kembali.

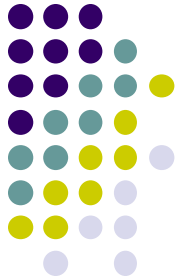
- Caranya adalah dengan membaca *byte-byte* di dalam citra, mengambil bit LSB-nya, dan merangkainya kembali menjadi bit-bit pesan.

- Contoh: Misalkan *stego-object* adalah sbb

  00110011 10100011 11100011 10101010 00100110
  10010111 11001000 11111001 10001001 10100011

  Ekstrak bit-bit LSB: 1110010111

Menghitung Ukuran Pesan yang dapat Disembunyikan



- Ukuran pesan yang akan disembunyikan bergantung pada ukuran *cover-object*.

- Misalkan pada citra *grayscale* (1 *byte/pixel*) 256 x 256 *pixel* :
  - jumlah *pixel* = jumlah byte = 256 x 256 = 65536
  - setiap byte dapat menyembunyikan 1 bit pesan di LSB-nya
  - jadi ukuran maksimal pesan = 65536 bit = 8192 *byte* = 8 KB

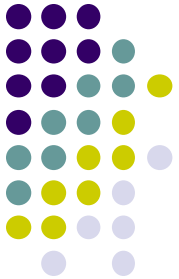
- Pada citra berwarna 24-bit berukuran 256 $\times$ 256 *pixel*:
  - jumlah *pixel* 256 x 256 = 65536
  - setiap pixel = 3 byte, berarti ada 65536 $\times$ 3 = 196608 *byte*.
  - setiap *byte* dapat menyembunyikan 1 bit pesan
  - jadi ukuran maksimal pesan = 196608 bit = 24576 byte = 24KB

Beberapa Varian Metode LSB

1. *Sequential*

- Bit-bit pesan disembunyikan secara sekuensial pada *pixel-pixel* citra.
- Misalkan ukuran pesan = 15 bit, maka urutan *pixel-pixel* yang digunakan untuk penyembunyian bit adalah:

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	
						=	



Ekstraksi pesan dari *Stego-image*

- Pada proses ekstraksi pesan, *pixel-pixel* dibaca secara sekuensial mulai dari *pixel* pertama sampai *pixel* yang menyimpan bit pesan terakhir

- Ambil setiap *byte* dari *pixel*, ekstraksi bit LSB-nya.

- Rangkailah bit-bit LSB menjadi bit-bit pesan semula.

2. Acak

- Untuk membuat penyembunyian pesan lebih aman, bit-bit pesan tidak disimpan pada pixel-pixel yang berurutan, namun dipilih secara acak.

- Pembangkit bilangan acak-semu (*PRNG*: *pseudo-random number generator*) digunakan untuk membangkitkan bilangan acak.

- Umpan (*seed*) untuk pembangkit bilangan acak berlaku sebagai kunci (*stego-key*).

- Misalnya jika terdapat 64 *byte* dan 15 bit pesan yang akan disembunyikan. *Pixel-pixel* dipilih secara acak, seperti pada gambar berikut.

				5			8
	10					4	
			13		2		
7							9
		1			12		
		15					
11						3	
			6				14

Ekstraksi pesan dari *Stego-image*

- Posisi *pixel* yang menyimpan bit pesan dapat diketahui dari bilangan acak yang dibangkitkan oleh *PRNG*.

- Jika kunci yang digunakan pada waktu ekstraksi sama dengan kunci pada waktu penyisipan, maka bilangan acak yang dibangkitkan juga sama.

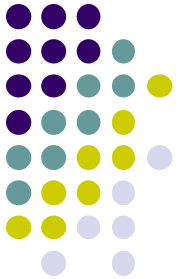
- Dengan demikian, bit-bit pesan yang bertaburan di dalam citra dapat dikumpulkan kembali.

3. *m*-bit LSB

- Untuk meningkatkan ukuran pesan yang disembunyikan, maka digunakan lebih dari 1 bit LSB untuk setiap *byte*.
- Susunan bit pada setiap *byte* adalah $b_7b_6b_5b_4b_3b_2b_1b_0$. Jika diambil 2-bit LSB, maka bit yang digunakan adalah bit b_1 dan bit b_0

  Contoh: 11010010 → 2 bit LSB terakhir dipakai untuk menyembunyikian pesan.
- *Trade-off*: Semakin banyak bit LSB yang digunakan, semakin besar ukuran pesan yang dapat disembunyikan, tetapi semakin turun kualitas *stego-image*.

- Pesan dapat disembunyikan secara sekuensial atau secara acak pada *pixel-pixe*l di dalam citra.



4. Enkripsi XOR

- Pesan dapat dienkripsi terlebih dahulu sebelum disembunyikan ke dalam citra.
- Teknik enkripsi yang sederhana adalah dengan meng-XOR-kan bit-bit pesan dengan bit-bit kunci. Jumlah bit-bit kunci sama dengan jumlah bit pesan.
- Bit-bit kunci dibangkitkan secara acak.
- Kunci untuk pembangkitan bit-bit kunci menjadi *stego-key*.
- Jika dipakai teknik acak dalam memilih *pixel-pixel*, maka ada dua *stego-key*: satu untuk pembangkitan bit-bit kunci, satu lagi untuk pembangkitan posisi *pixel* yang dipilih untuk menyembunyikan pesan.

PSNR

- PSNR = *Peak-Signal-to-Noise Ratio*
- Merupakan metrik untuk mengukur kualitas (*fidelity*) citra setelah proses manipulasi.
- Selalu dibandingkan dengan citra semula (yang belum dimanipulasi).
- Misalkan *I* = *cover-image* dan $\hat{I}$ = *stego-image*, ukuran citra *M* x *N*, maka

$$PSNR = 20 \times \log_{10}\left(\frac{256}{rms}\right) \quad \text{dengan} \quad rms = \sqrt{\frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}(I_{ij} - \hat{I}_{ij})^2}$$

rms = root mean square

- Satuan *PSNR* adalah desibel (dB).

- *PSNR* menyatakan visibilitas derau di dalam citra.
- Nilai PSNR berbanding terbalik dengan *rms*.

- *PSNR* yang besar mengindikasikan nilai *rms* yang kecil; *rms* kecil berarti dua buah citra mempunyai hanya sedikit perbedaan.

- PSNR yang kecil mengindikasikan nilai *rms* yang besar; *rms* besar berarti kedua citra memiliki perbedaan yang besar (degradasi).

- PSNR yang dapat diterima/ditoleransi adalah jika > 30

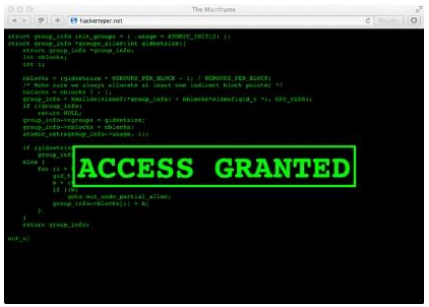
How to Hack a Computer Using Just An Image

Monday, June 01, 2015 Swati Khandelwal

Next time when someone sends you a photo of a cute cat or a hot chick than be careful before you CLICK on the image to view — it might hack your machine.

Yes, the normal looking images could hack your computers — thanks to a technique discovered by security researcher *Saumil Shah* from India.

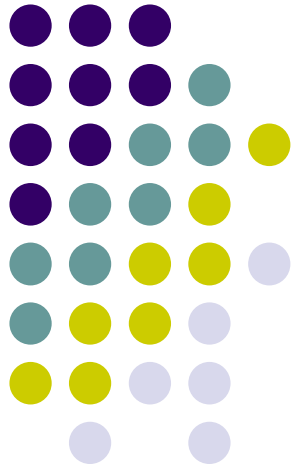
Dubbed "***Stegosploit***," the technique lets hackers hide malicious code inside the pixels of an image, hiding a malware exploit in plain sight to infect target victims.



Just look at the image and you are HACKED!

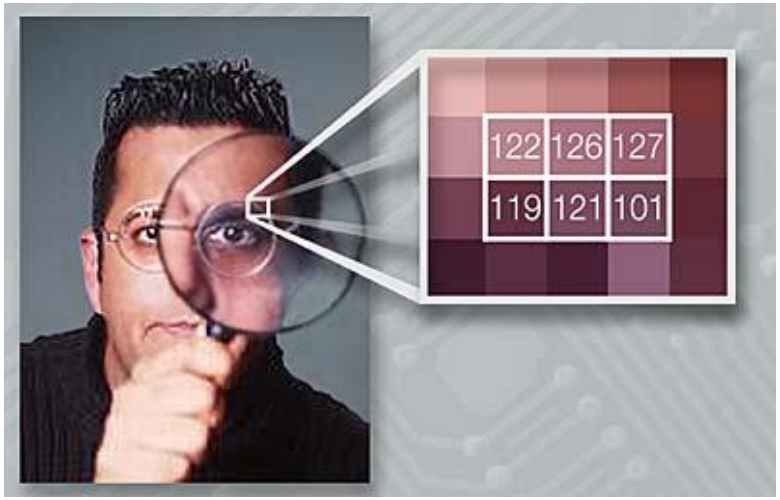
http://thehackernews.com/2015/06/Stegosploit-malware.html

Pengantar Steganalisis

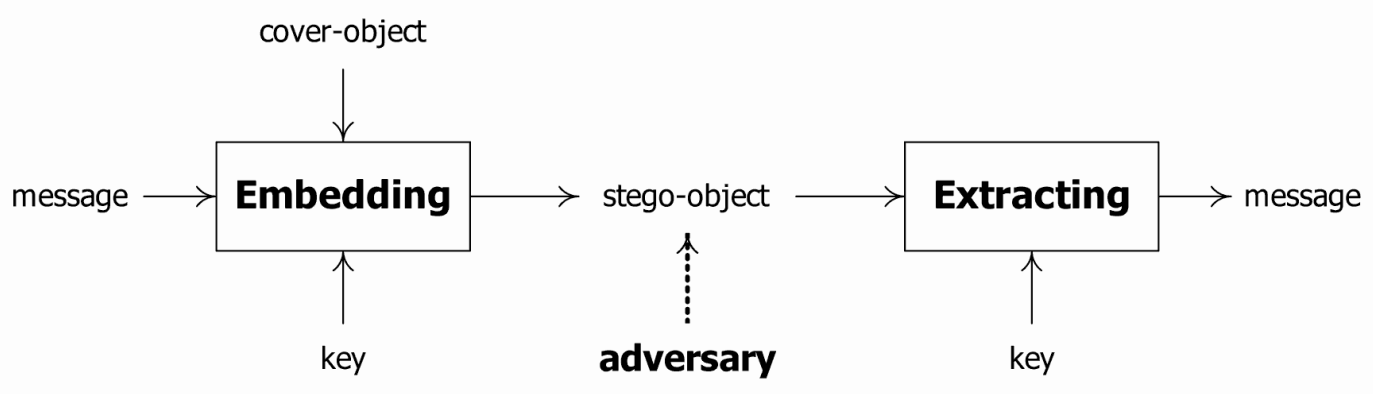


Steganalysis

- Tujuan: menentukan apakah sebuah media *suspect* mengandung pesan tersembunyi



- Steganografi

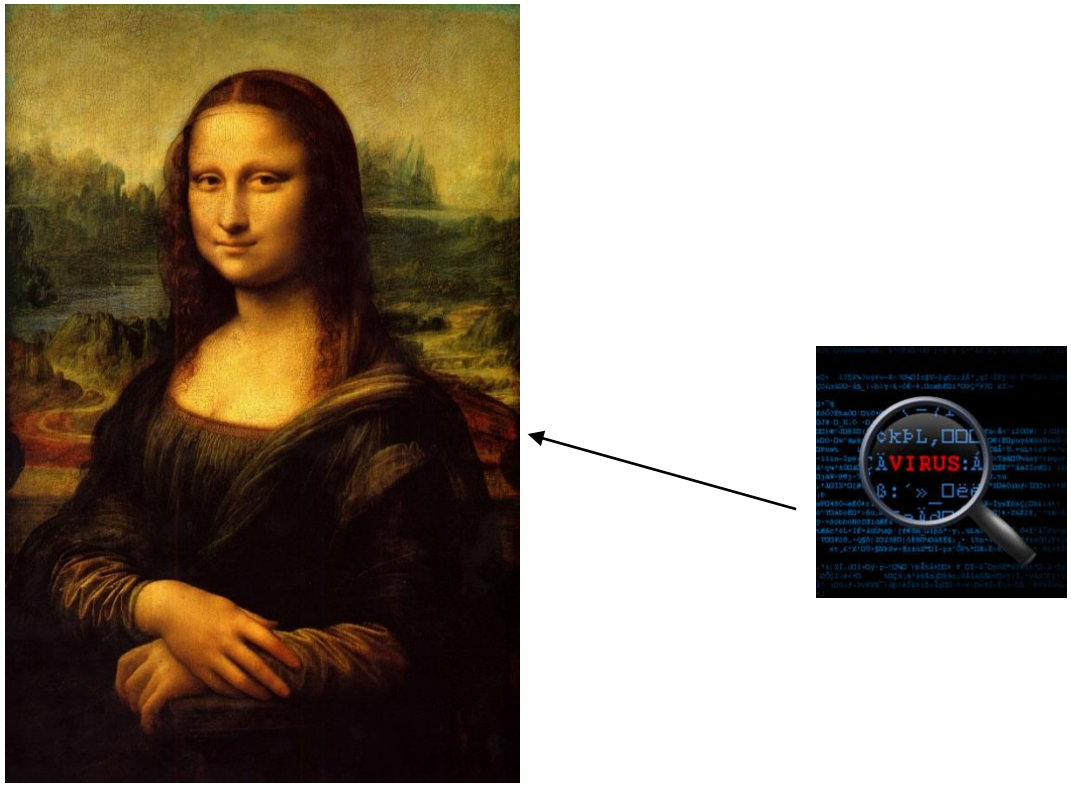


- Steganalisis

*) Keterangan: 1 jika ada pesan tersembunyi, 0 jika tidak

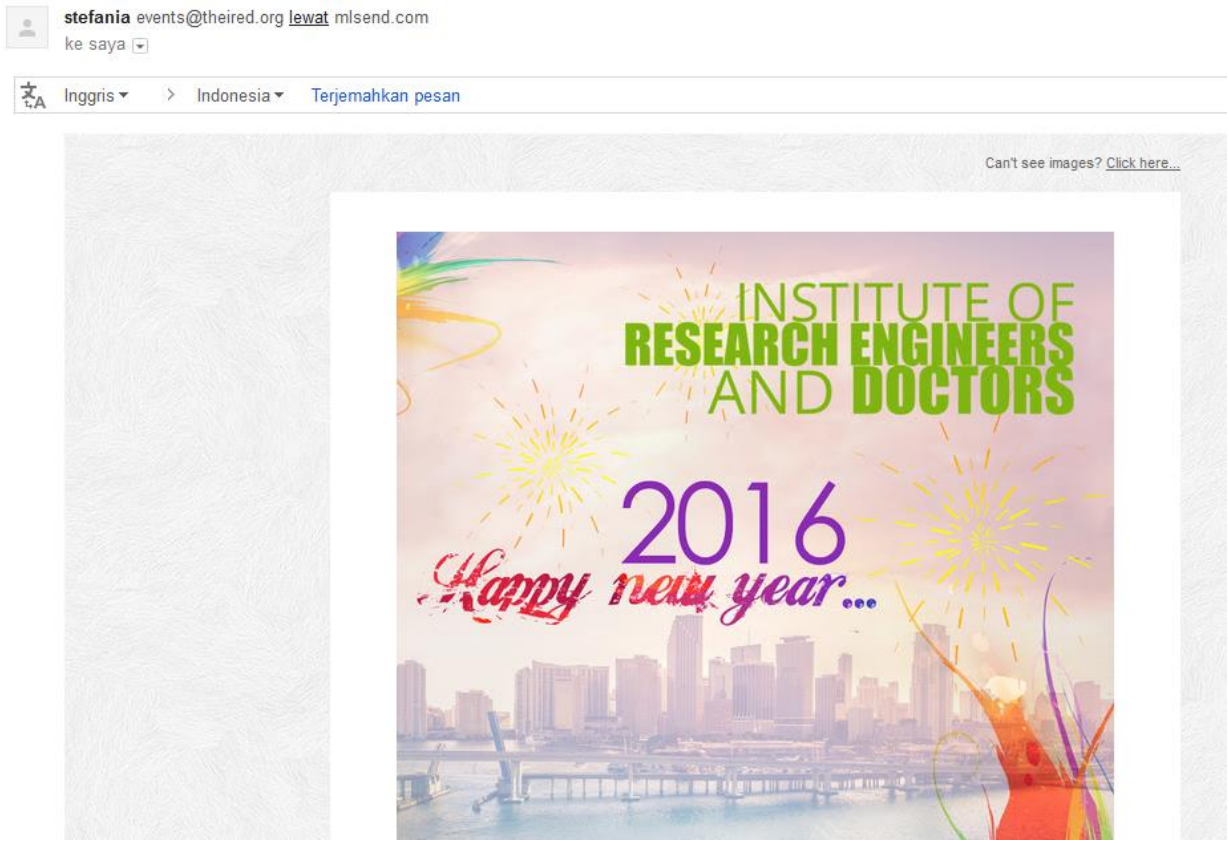
Fakta: Gambar-gambar bertebaran di internet (*website*, *social media*, *social networking*)

Namun, dibalik sebuah gambar dapat tersembunyi informasi rahasia

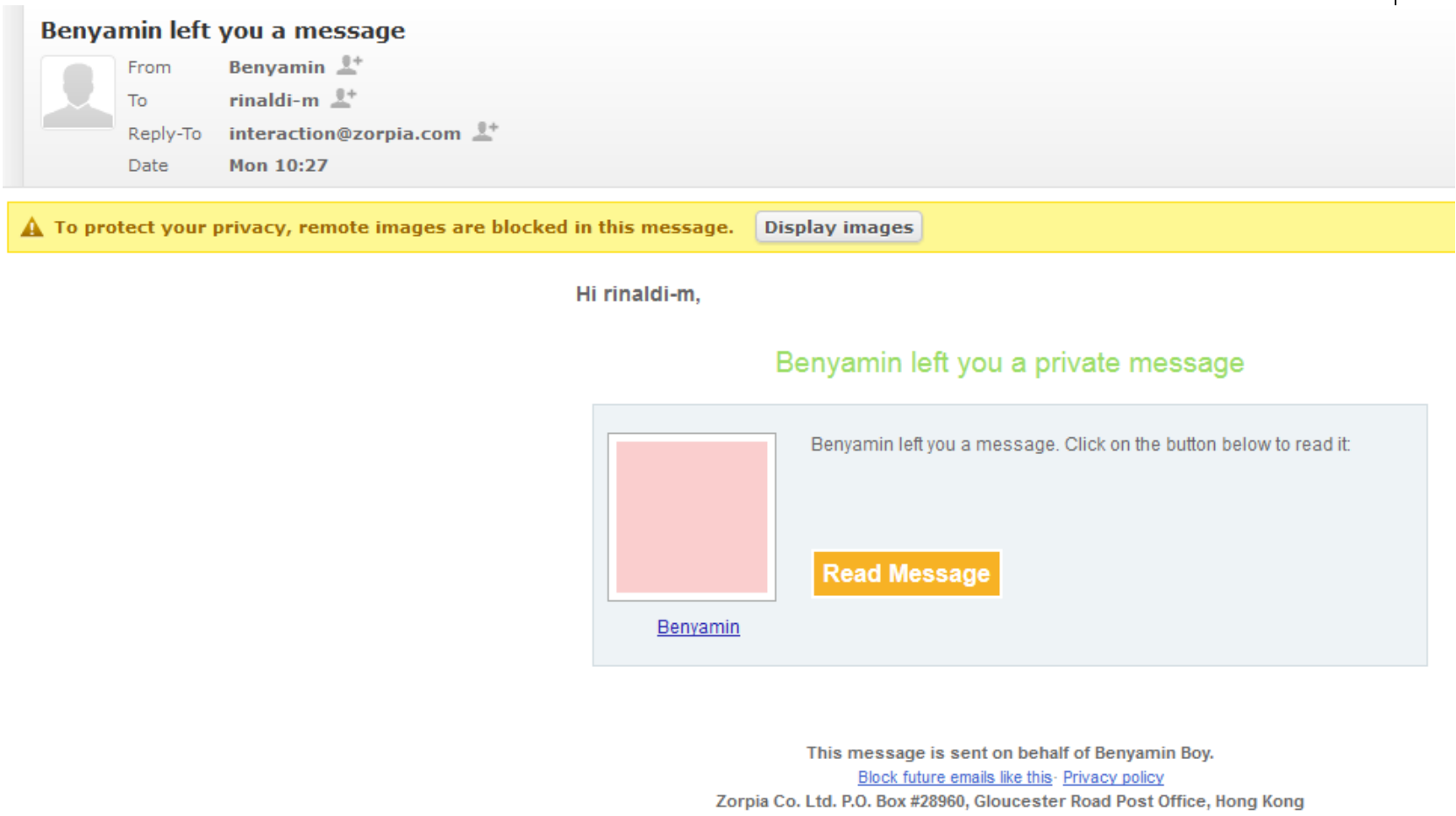


Informasi rahasia tersebut dapat berupa pesan biasa, pesan kejahatan, program jahat, bahkan virus komputer!

Pernah terima surel (*e-mail*) dari orang tak dikenal dan mengandung *file attachmet* berupa gambar seperti di bawah ini?



HATI-HATI!!!!!!!!!



HATI-HATI! Jangan langsung klik jika anda tidak yakin!

Ingat kembali stegosploit!!!

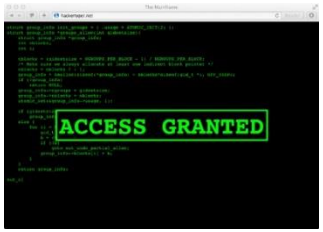
How to Hack a Computer Using Just An Image

Monday, June 01, 2015 Swati Khandelwal

Next time when someone sends you a photo of a cute cat or a hot chick than be careful before you CLICK on the image to view — it might hack your machine.

Yes, the normal looking images could hack your computers — thanks to a technique discovered by security researcher *Saumil Shah* from India.

Dubbed "***Stegosploit***," the technique lets hackers hide malicious code inside the pixels of an image, hiding a malware exploit in plain sight to infect target victims.



Just look at the image and you are HACKED!

http://thehackernews.com/2015/06/Stegosploit-malware.html

- Steganalisis diperlukan di dalam *forensic image analysis*

- ***Forensic Image Analysis*** *is the application of image science and domain expertise to interpret the content of an image and/or the image itself in legal matters.*

- Subdisiplin dari *Forensic Image Analysis:*

  (1) Photogrammetry

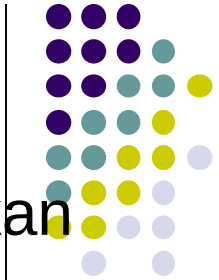
  (2) Photographic Comparison

  (3) Content Analysis

  (4) Image Authentication

- Salah satu pekerjaan di dalam *content analysis* adalah mendeteksi apakah ada pesan tersembunyi di dalam sebuah gambar.

- Contoh sebuah skenario: Mr. Abdul, seorang investigator forensik, diminta Lab Forensik Polri untuk menginvestigasi sebuah *cybercrime* berupa foto. Sebagai investigator forensik yang ahli, dia menganalisis foto untuk menemukan pesan tersembunyi di dalamnya dengan kakas steganalisis.



- Tujuan utama steganalisis adalah untuk membedakan apakah sebuah media mengandung pesan rahasia atau tidak.

- Steganalisis dianggap berhasil jika ia dapat menentukan apakah sebuah media mengandung pesan tersembunyi dengan peluang lebih tinggi daripada menerka secara acak.

- Selain tujuan utama di atas, terdapat beberapa tujuan minor steganalisis:

  - menentukan panjang pesan
  - menentukan tipe algoritma penyisipan
  - kunci yang digunakan

Jenis-jenis steganalisis

1. ***Targeted steganalysis***

- Teknik steganalisis yang bekerja pada algoritma steganografi spesifik, dan kadang-kadang dibatasi hanya pada format media tertentu saja.

- Teknik ini mempelajari dan menganalisis algoritma penyisipan, lalu menemukan statistik yang berubah setelah penyisipan.

- Hasil steganalisis sangat akurat, tetapi tidak fleksibel karena tidak dapat diperluas untuk algoritma steganografi yang lain atau format media yang berbeda.

2. ***Blind steganalysis***

- Teknik steganalisis yang bekerja pada sembarang algoritma steganografi dan sembarang format media.

- Teknik ini mempelajari perbedaan antara statistik *cover-object* dan *stego-object* dan membedakannya. Proses pembelajaran (*learning*) dilakukan dengan melatih (*training*) mesin pada sekumpulan database media. Model *machine learning* yang digunakan misalnya jaringan syaraf tiruan.

- Hasil steganalisis kurang akurat dibandingkan dengan teknik *targeted steganalysis*, tetapi kelebihannya adalah dapat diperluas untuk algoritma yang lain.

Metode Steganalisis

1 . **Serangan berbasis visual (*visual attacks*)**

- Khusus untuk *stego-object* berupa citra
- Bersifat subjektif, karena melakukan pengamatan secara kasat mata dengan melihat artefak yang mencurigakan di dalam *stego-image*, lalu membandingkannya dengan citra asli (cover *image*)
- Digunakan pada masa-masa awal riset steganalisis
- Contoh serangan visual:

  a. *LSB plane attack*

  b. *Filtered visual attack* (E*nhanced LSB*)

2. Serangan berbasis statistik (*statistical attack*)

- Menggunakan analisis matematik pada citra untuk menemukan perbedaan antara *cover image* dengan *stego image*.
- Didasarkan pada fakta bahwa penyembunyian pesan ke dalam media menimbulkan artefak yang dapat dideteksi secara statistik sehingga dapat mengungkap penyembunyian pesan atau pesan yang disembunyikan itu sendiri.
- Contoh serangan statistik:
  a. *histogram analysis*
  b. *Regular-singular* (RS) *analysis*
  c. *Chi-square analysis*
  d. *Sample pair* (SP) *analysis*

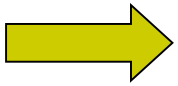
Visual Attack

- Memanfaatkan indera penglihatan → inspeksi kerusakan pada gambar akibat penyisipan [WES99]

- Ide dasar :

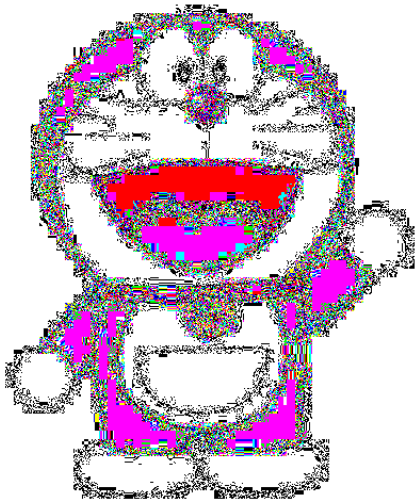
Metode Enhanced LSB

BLUE	GREEN	RED
1010010**1**	1001110**0**	1110011**1**



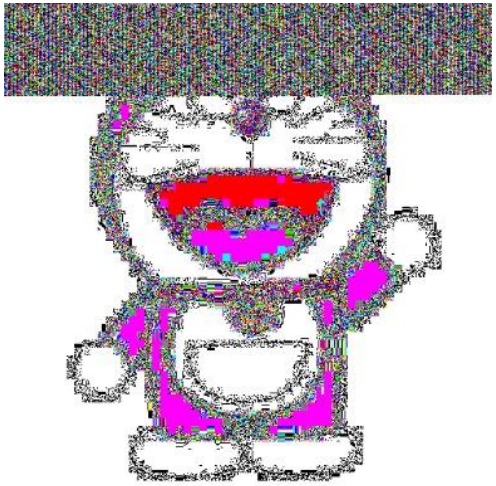
BLUE	GREEN	RED
11111111	**00000000**	**11111111**

(a) Citra orisinal



(b) Citra hasil *enhanced LSB*

(c) Citra stego



(b) Citra hasil *enhanced LSB*

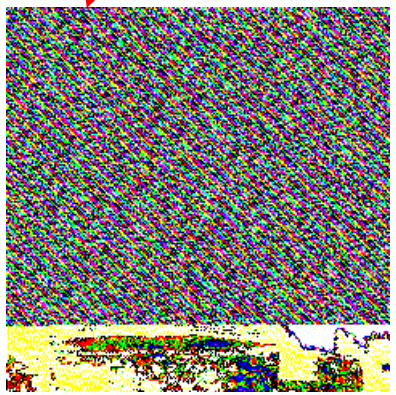
Teknik Steganalisis: *Visual Attack*

Artefak mencurigakan

Gambar asli

Hasil penapisan (asli)

Terdeteksi ada pesan



Terdeteksi ada pesan

Terdeteksi ada pesan

Terdeteksi ada pesan

Metode *enhanced-LSB* bagus untuk citra dengan kontras tinggi, yaitu citra yang memiliki warna latar yang jelas atau memiliki perbedaan warna yang kontras antara latar dengan gambar utama

Gambar III-1 Gambar yang mengandung pesan rahasia dan hasil *enhanced LSB*-nya [PAU07]

Untuk citra dengan kontras rendah (seperti citra hasil fotografi), metode *enhanced LSB* seringkali menyulitkan steganalis. Karena steganalis akan kesulitan membedakan antara gambar yang seharusnya muncul dengan pesan rahasia.

Gambar III-3 Gambar dengan kontras rendah dan hasil *enhaced LSB*-nya

Metode *Chi-square*

- *Chi-square attack* merupakan serangan berbasis statistik yang menganalisis histogram dari *PoV* (*Pairs of Value*).

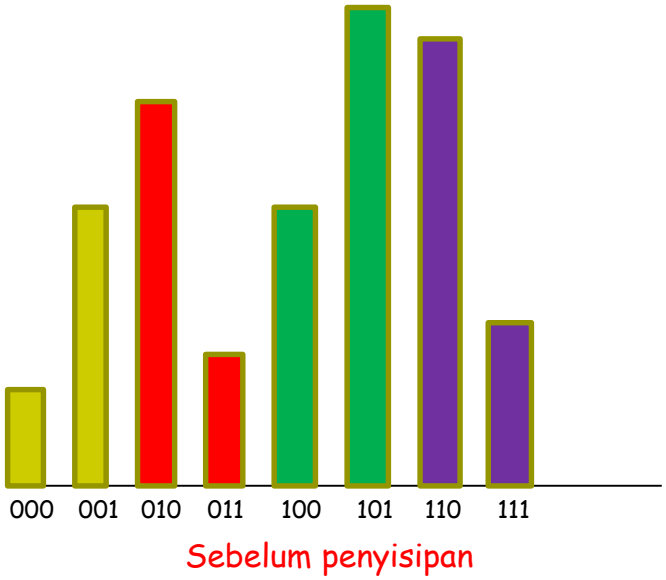
- PoV adalah pasangan nilai yang hanya berbeda pada bit LSB-nya saja.

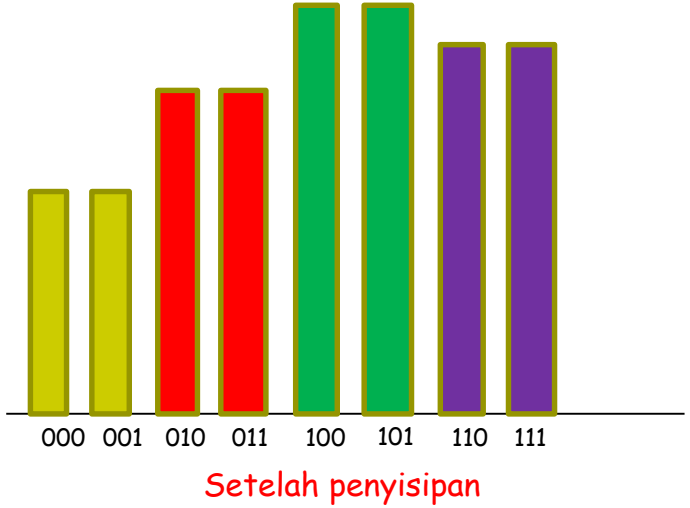
  Contoh: 10 dan 11 (0000101<u>0</u> dan 0000101<u>1</u>)

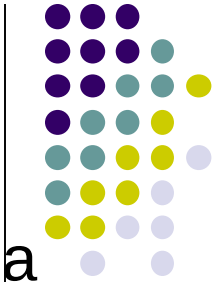
  128 dan 129 (1000000<u>0</u> dan 1000000<u>1</u>)

- Penyisipan pesan dengan metode LSB pada dasarnya mengganti bit LSB dengan bit pesan.
- Penggantian bit LSB tersebut hanya mengubah nilai bit LSB dari 0 menjadi 1 atau dari 1 menjadi 0 (*flip embedding*).
- Ini berarti nilai-nilai di dalam setiap *PoV* hanya mengalami *swapping* (saling dipertukarkan)
- *Chi-square attack* menganalisis histogram dari *PoV* yang nilai-nilainya saling dipertukarkan (*swapping*) selama proses penyisipan pesan.

Histogram nilai-nilai *pixel* sebelum dan setelah penyisipan:







- Ide *chi-square attack* adalah menguji frekuensi kedua nilai pada setiap PoV adalah sama.

- Jadi, misalkan *M* adalah *suspect image* (dicurigai mengandung pesan). Maka:
  - hitung distribusi frekuensi pada citra *M*
  - hitung distribusi frekuensi yang dipredikisi akan dimiliki jika citra orisinal dari *M* disisipi pesan.

- Jika kedua distribusi tersebut sama,berarti kemungkinan besar terdapat pesan di dalam citra *M*.

- Peluang bahwa kedua distribusi yersebut sama dihitung dengan persamaan:

$$p = 1 - \frac{1}{2^{\frac{k-1}{2}}\Gamma(\frac{k-1}{2})} \int_0^{\chi^2_{k-1}} e^{-\frac{u}{2}} u^{\frac{k-1}{2}-1} du$$

Γ adalah fungsi *Gamma Euler*, $\Gamma(u) = \int_0^{\infty} e^{-t} t^{u-1} dt$

Nilai *Chi-square* (χ^2) dengan derajat kebebasan $k - 1$ dihitung dengan persamaan $\chi^2_{k-1} = \sum_{i=1}^{k} \frac{(x_i - z_i)^2}{z_i}$

x_i = jumlah *pixel* dengan nilai warna $2i$

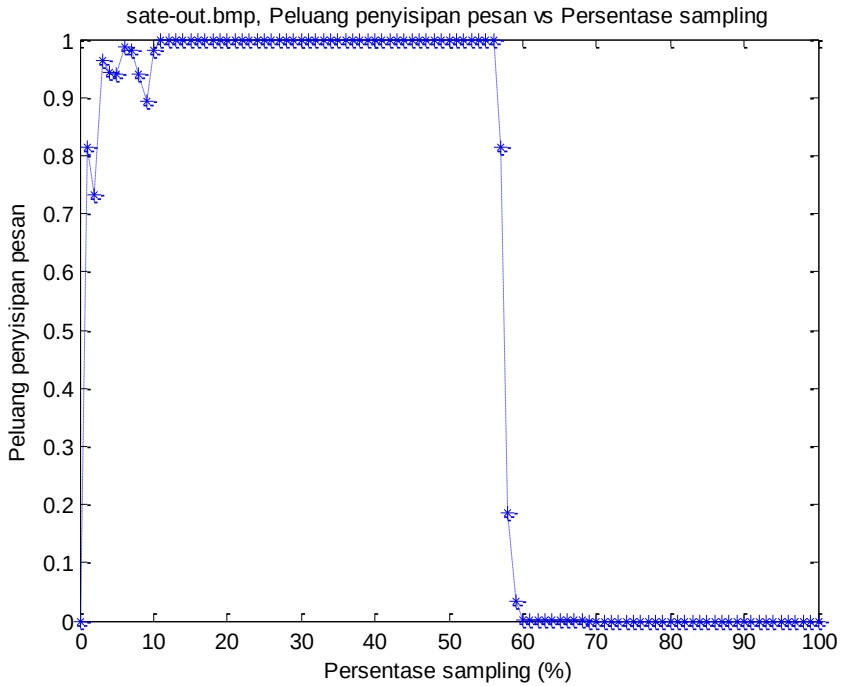
y_i = jumlah *pixel* dengan nilai warna $2i + 1$

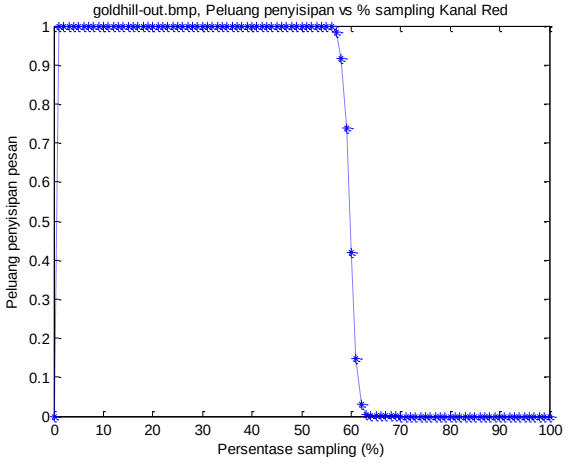
$z_i = \frac{x_i + y_i}{2}$

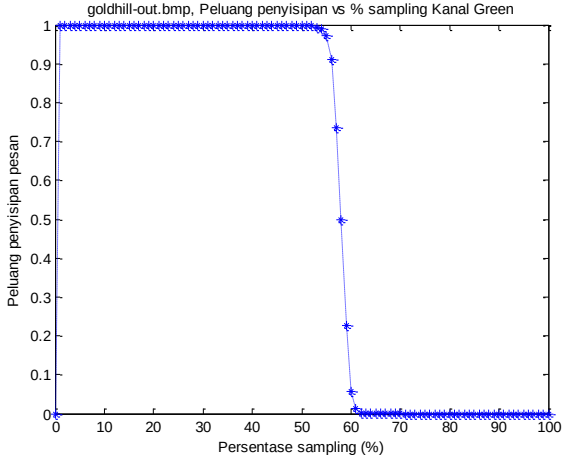
- Jika nilai *p* semakin mendekati 1, maka semakin besar kemungkinan citra disisipi pesan.

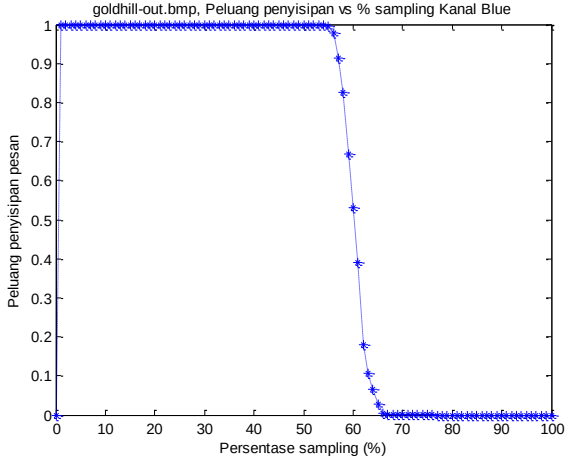
- Sebaliknya, jika nilai *p* mendekati 0, maka semakin kecil kemungkinan citra disisipi pesan.

- Jadi, nilai *p* dapat digunakan untuk mengindikasikan ada atau tidak pesan tersembunyi di dalam citra


sate-out.bmp, Peluang penyisipan pesan vs Persentase sampling
Peluang penyisipan pesan
Persentase sampling (%)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
10
20
30
40
50
60
70
80
90
100


goldhill-out.bmp, Peluang penyisipan vs % sampling Kanal Red
Peluang penyisipan pesan
Persentase sampling (%)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
10
20
30
40
50
60
70
80
90
100


goldhill-out.bmp, Peluang penyisipan vs % sampling Kanal Green
Peluang penyisipan pesan
Persentase sampling (%)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
10
20
30
40
50
60
70
80
90
100


goldhill-out.bmp, Peluang penyisipan vs % sampling Kanal Blue
Peluang penyisipan pesan
Persentase sampling (%)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
10
20
30
40
50
60
70
80
90
100

- Sekali citra diketahaui mengandung pesan rahasia, maka pesan tsb bisa dihancurkan dengan mengganti seluruh bit-bit LSB

Contoh penghancuran pesan pada citra adalah sebagai berikut :

1. Terdapat citra yang bit *LSB*-nya telah disisipi pesan rahasia, sbb:

(00100111 11101000 11001000)

(00100110 11001000 11101000)

(11001000 00100110 11101001)

2. Maka bit pesan rahasia tersebut adalah : **100000001**
3. Dilakukan penggantian bit *LSB* citra, menjadi : **000000000**

4. Bit tersebut kembali disisipkan pada citra, menjadi :

(00100110 11101000 11001000)

(00100110 11001000 11101000)

(11001000 00100110 11101000)

Penggantian seluruh bit *LSB* menjadi 0, tidak akan merusak tampilan citra, karena mata manusia tidak dapat membedakan perubahan yang terjadi pada bit *LSB*. Contoh citra yang memiliki pesan dan citra setelah pesan dihancurkan dapat dilihat pada Gambar III-5

Gambar III-5 Citra dengan pesan rahasia (kanan) dan citra setelah pesan dihancurkan (kiri)

Ada pertanyaan?

Referensi

- Li, F., *The art and science of writing hidden messages: Steganography*
- Khan, M. M. , *Steganography*
- Wohlgemuth, S. (2002), IT-Security: Theory and Practice : *Steganography and Watermarking,* University of Freiburg, Denmakr, 2002.
- Wong, P.W. (1997). *A Watermark for Image Integrity and Ownership Verification*. Prosiding *IS&T PIC Conference*.
- *Tawalbeh, L.* (2006), *Watermarking,* Information System Security AABFS-Jordan.
- Bae, S.H. (2006), *Copyright Protection of Digital Image*, Tongmyong University of information technology
- Yuli Anneria Sinaga, *Steganalisis dengan Metode Chi-square dan RS-analysis*, Tugas Akhir Informatika, IT
- Wikipedia