# **KEAMANAN JARINGAN KOMPUTER**

# SIMULASI ALGORITMA STEGANOGRAFI DENGAN SOFTWARE OPEN PUFF

NAUFAL AHMAD FARAUQ

#### I. PENDAHULUAN

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, dimana disinilah fungsi dari teknik steganography yaitu sebagai teknik penyamaran menggunakan media lain yang berbeda sehingga informasi rahasia dalam media awal tidak terlihat secara jelas. Steganography juga berbeda dengan cryptography yaitu terletak padahasil keluarannya. Hasil dari cryptography biasanyaberupa 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. Selain itu steganography keberadaan informasi yang disembunyikan pada 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. Namun secara umum steganography dan cryptography mempunyai tujuan yang sama yakni mengamankan data, bagaimana supaya data tidak dapat dibaca, dimengerti atau diketahui secara langsung. Steganography memanfaatkan kekurangan kekurangan indra manusia seperti mata dan telinga. Dengan kekurangan inilah maka teknik ini dapat diterapkan dalam berbagai media digital. Media yang dimaksudkan adalah media dalam bentuk file digital dengan berbagai format, antara lain :Images (bmp, gif, jpeg, tif, dll), Audio (wav, Mp3, dll), Video (flv).

## II. LANGKAH PERCOBAAN

### HIDING INFORMATION

1. Buka software Open Puff v4.00

OpenPuff v4.00 - Steganography & Marking
Steganography Hide Unhide
Volatile marking & Carrier clean up SetMark CheckMark CleanUp
Help & Options Home Help Help Threads: 2

- 2. Pada gambar diatas merupakan tampilan dari Open Puff v4.00. Pilih menu steganography, menu hide untuk menyembunyikan informasi dan menu unhide untuk membuka informasi yang telah di-*hide*.
- 3. Buat file pesan yang akan disembunyikan dengan teknik steganografi. File pesan bisa berupa file .txt, .bmp, .jpg, .wav, .flv, dll. Dalam praktikum ini file pesan menggunakan file .txt

pesan rahasia - Notepad
<u>Eile E</u> dit F <u>o</u> rmat <u>V</u> iew <u>H</u> elp
Pesan Rahasia buat kamu. jangan bilang siapa-siapa saya powerranger hitam.

4. Klik hide pada Open Puff v4.00.

5. Masukkan *password* cryptography dengan minimal 8 karakter dan maksimal 32 karakter. Lihat seperti gambar dibawah ini.

OpenPuff v4.00 - Data Hiding			X
(1) Insert 3 uncorrelated data passwor Cryptography (A)      ********************************	ds (Min: 8, Max: 32) (B) Enable (B) (C) A = B = C tance (X) (Y) >= 25%	(2) Data (Max: 256Mb) Target Size (4) Bit selection options	Browse 0 bytes
(3) Carrier selection [Order sensitive]         (Name) Sort by name / (Bytes) Sort by by         Name         (-) Move up selected / (+) Move dow         Add       Selected / Total	ytes Shuffle Clear Bytes Chain Order n selected / (Del) Delete selected 0 / 0 bytes	• Support (stream)         • Aff (Audio)         • Fiv (Stream)         • Mp3 (Audio)         • Mp3 (File)         • Mp3 (File)         • Mp3 (File)         • Mp3 (Stream)         • Mp3 (Stream)         • Mp3 (Stream)         • Mp3 (Stream)         • Mp3 (Mage)         • Mp3 (Mage)         • Mp3 (Mage)         • Mp3 (Audio)         • Me4 (Stream)         • Mp3 (Audio)         • Me4 (Stream)         • Mp3 (Mage)         • Mage (Audio)	Hide Data!

6. Pilih file dengan klik *browse* dengan maksimum data sebesar 256Mb, seperti gambar dibawah ini.

OpenPuff v4.00 - Data Hiding		
(1) Insert 3 uncorrelated data passwords (Min: 8, Max: 32 Cryptography (A)	2)	(2) Data (Max: 256Mb) Target [C:\Users\NAUFAL\Desktop\Percob Browse
Scrambling (C) Enab	ole (B) 🔲 (C) 🔲	Size 10 + name(17) + data(75) bytes
Passwords check         A = B = C           H (X,Y) = Hamming distance (X)(Y) >=	: 25%	(4) Bit selection options
[3] Carrier selection [Order sensitive]     [Name) Sort by name / (Bytes) Sort by bytes     Shuffle	Clear	Aiff (Audio) Aiff (Audio) Bitmap (Image) Flv (Stream)
Name Bytes	Chain Order	A Jpeg (Image)     Wp3 (Audio)     Mp3 (Stream)     Mp3 (J11 (Stream)     Mp3 (J11 (Stream)     Next/Sun (Audio)     A Pex (Image)     Pex (Image)
(-) Move un selected / (-) Move down selected / [Del] [	Delete selected	Image       Image <t< td=""></t<>
Add Selected / Total	0 / 102 bytes	Reset Options Add Decoy! Hide Data

7. Tentukan file yang dijadikan sebagai file pengangkut. Kriteria file pengangkut harus lebih besar dari file yang akan dibawa. Klik add seperti gambar dibawah ini. Dari keterangan gambar dibawah ini diketahui bahwa file pengangkut dapat mengangkut maksimum sebesar file 1.088 bytes dan file yang diangkut sebesar 102 bytes.

OpenPuff v4.00 - Data Hiding	And Address of States	×
OpenPuff v4.00 - Data Hiding          (1) Insert 3 uncorrelated data password         Cryptography       (A)         Scrambling       (C)         Passwords check       H         H       (X, Y)         H       (X, Y)         Name       Manual (Bytes) Sort by byte         Name       Koala.jpg         (-) Move up selected / (+) Move down	s (Min: 8, Max: 32) (B) Enable (B) (C) ( A = B = C ance (X) (Y) >= 25% es Shuffle Clear Bytes Chain Order 1.088 #00000 selected / (Del) Delete selected	(2) Data (Max: 256Mb)         Target       C:\Users\NAUFAL\Desktop\Percob       Browse         Size       10 + name(17) + data(75) bytes         (4) Bit selection options
Add Selected / Total	1.088 / 102 bytes	Reset Options Add Decoy! Hide Data!

#### 8. Klik *Hide Data*!

(1) Insert 3 uncorrelated data passv	vords (Min: 8, Max: 32)	(2) Data (Max: 256Mb)
Cryptography (A)	(B)	Target C:\Users\NAUFAL\Desktop\Percob Browse
Scrambling (C)	Enable (B) 🔲 (C) 🗐	Size 10 + name(17) + data(75) byt
Passwords check	A = B = C	- (A) Bit selection options
H(X,Y) = Hamming	distance(X)(Y) >= 25%	(4) bit selection options
(Name) Sort by name / (Bytes) Sort by Name	y bytes Shuttle Clear	■-■ Flv (Stream) ■-A Jpeg (Image)
👃 Koala.jpg	1.088 # 00000	e ⊷ Mp3 (Audio) e ■ Mp4 (Stream) e ■ Mpg I/II (Stream)
🔪 Koala ipg	1.088 # 00000	Mp3 (Audio)       Hp4 (Stream)       Hp5 (Xlream)       Hst/Sun (Audio)       Mst/Sun (Audio)
🏠 Koala jog	, 2,2,2,3, 2,2,2,00000 T.088 #00000	Mp3 (Audio)       Mp4 (Stream)       Mp4 (Stream)       Mp3 (J/l (Stream))
Koala jog (·) Move up selected / (+) Move di	T.088 # 00000	Image: Second

9. Pilih lokasi data akan disimpan.





OpenPuff v4.00 - Task Report	
**** Begin of Report ***	<u> </u>
Bit Selection: Jpeq (Image) <- 1/5 (20%) - Medium	
	E
#UUUUU <- Koala.pg	
t.	Done

10. Hasil dari langkah-langkah 1-9. Pada gambar dibawah ini dapat diketahui secara kasat mata bahwa file yang telah disisipi pesan dengan yang aslinya tampak sama.









## UNHIDE INFORMATION

1. Buka software Open Puff v4.00. Klik Unhide

📑 OpenPuff v4.00 - Steganography & Marking
Steganography Hide Unhide
Volatile marking & Carrier clean up
Help & Options Home Help Help Threads: 2

2. Masukkan password yang sama sesuai dengan saat hidding Information.

OpenPuff v4.00 - Data Unhiding  (1) Insert 3 uncorrelated passwords (h Cuyptography (A)  Scrambling (C)	Min: 8, Max: 32) (B) Enable (B) (C)	(3) Bit selection options (3) ≣it selection options (3) ≣it selection options (3) ∰ (3) (1) (1) (1) (1) (1) (1) (1) (1) (1) (1	×
Passwords check H (X,Y) = Hamming di (2) Carrier selection [Order sensitive] (Name) Sort by name / [Bytes) Sort by	A = B = C stance (X) (Y) >= 25% bytes Clear	A Jpeg (Image)     Mp3 (Audio)     Mp4 (Stream)     Mp4 (Stream)	
Name	Bytes   Chain Order	Image: Series     Series       Image: Series     Series <td></td>	
(-) Move up selected / (+) Move dov Add Carriers Selected	vn selected / (Del) Delete selected O bytes	Reset Options	Unhidel

3. Selanjutnya *upload* file yang menjadi file pengangkut/pembawa. Klik *add carrier* dan cari file pembawa tadi.





4. Setelah itu maka tampilan akan seperti gambar dibawah ini. Dan dapat diketahui juga file tersebut masih sama terdapat informasi file tersebut dapat mengangkut file sebesar 1088 bytes.

OpenPuff v4.00 - Data Unhiding	100 100		×
(1) Insert 3 uncorrelated passwords (N	vlin: 8, Max: 32)	(3) Bit selection options	]
Cryptography (A)	(B)	⊞ <b>≣</b> 3gpp (Stream)	
Scrambling (C)	Enable (B) 🔲 (C) 🗌	Bitmap (Image)	
Passwords check	A = B = C	·····································	
H (X,Y) = Hamming di	stance (X) (Y) >= 25%	Harris (Audio) Harris Mp4 (Stream)	
<ul> <li>(2) Carrier selection [Order sensitive]</li> <li>(Name) Sort by name / (Bytes) Sort by</li> </ul>	bytes Clear	H → Mpg I/II (Stream) Next/Sun (Audio) → ♪ Pcx (Image) → ■ Pdf (File)	
Name	Bytes Chain Order	E S Par (Fine) 	
🔥 Koala A-1.jpg	1.088 <b>#</b> 00000	e-∎ Swf (Stream) e-// Tga (Image) e-∎ Vob (Stream) e ⊠ Wave (Audio)	
(·) Move up selected / (+) Move dow	vn selected / (Del) Delete selected		
Add Carriers Selected	1.088 bytes	Reset Options	Unhide!

5. Sebelum *unhide* pastikan bahwa *password*, file pembawa, dan *bit selection levels* nya sama dengan saat *hiding information*. Jika dari kesemuanya itu tidak sama maka tidak akan bisa melakukan *unhide*. Seperti tampilan gambar dibawah ini.



6. Kemudian klik *unhide*.

CopenPuff v4.00 - Data Unhiding				
Image: Second state of the second state of the second state stat	<ul> <li>(3) Bit selection options</li> <li>3gpp (Stream)</li> <li>Aiff (Audio)</li> <li>Flv (Stream)</li> <li>Flv (Stream)</li> <li>Mp3 (Audio)</li> <li>Mp4 (Stream)</li> <li>Mp1/II (Stream)</li> <li>Next/Sun (Audio)</li> <li>PCx (Image)</li> <li>Pdf (File)</li> <li>Pdf (File)</li> <li>Swf (Stream)</li> <li>Swf (Stream)</li> <li>Tga (Image)</li> <li>Yob (Stream)</li> <li>Wave (Audio)</li> </ul>			
(·) Move up selected / (+) Move down selected / (Del) Delete selected				
Add Carriers Selected 1.088 bytes	Reset Options Unhide!			



OpenPuff v4.00 - Task Report	and the second
*** Begin of Report ***	<u> </u>
Hidden file: Name ≺- nesan rahasia txt	
Size <- 75 byte(s) CRC32 <- 0x2CB526D8	
*** End of Report ***	-
	Done

7. Setelah berhasil, file pesan yang tadinya terlebur dengan file pembawa dengan melakukan proses *unhide information* maka file pesan tersebut akan berpisah dengan file pembawanya. Tampilan dibawah ini merupakan file pesan setelah dipisahkan dari file pembawanya dan masih dalam keadaan dengan file format dan file name yang sama.



### III. HASIL PERCOBAAN DAN ANALISA

Hasil percobaan seperti tabel dibawah ini.

	pesan msgs	yang harus diangkut (byte)	Jumlah byte yang dapat diangkut oleh pesan carrier (byte) dengan bit selection levels medium								
n o			JPG (581 KB)	JPG (836 MB)	JPG (3.45 MB)	MP3 (3.04 MB)	MP3 (4.75 MB)	MP3 (5.24 MB)	FLV (3.19 MB)	FLV (123 MB)	FLV (204 MB)
1	pesan-1.txt	26									
	pesan 1 -										
2	encrypted.txt	59									
3	pesan-2.txt	39									
	pesan 2 -										
4	encrypted.txt	79									
5	pesan-3.txt	51									
	pesan 3 -		448	1136	3488	4048	5792	8848	1360	35680	69008
6	encrypted.txt	99									
7	pesan-4.txt	75									
	pesan 4 -										
8	encrypted.txt	123									
9	pesan-5.txt	146									
	pesan 5 -										
10	encrypted.txt	207									

### Tabel. Hasil Percobaan

Dari data tabel diatas dapat diketahui bahwa korelasi antara besar file message dengan besar file carrier yaitu seberapa besar file pesan yang dapat diangkut file penggankut dengan ukuran file yang sama, maka jumlah maksimum byte yang dapat diangkut besarnya sama.

Untuk jenis file *carrier* yang dapat mengangkut dengan daya angkut yang lebih besar dari data diatas adalah file MP3. Dari data percobaan diatas untuk file *carrier* dengan besar file lebih dari kisaran 3Mb dari jenis file *carrier* yang memiliki daya angkut yang lebih besar yakni MP3.

Kemudian untuk parameter pengaruh pesan yang telah terenkripsi pada proses steganografi, pengaruhnya hanya terletak pada besar file yang akan diangkut. Karena dapat dilihat dari tabel hasil percobaan diatas file pesan yang terenkripsi lebih besar dari pesan yang tidak terenkripsi. Dan juga lebih besarnya file yang harus diangkut dari file terenkripsi lebih besar dari file yang harus diangkut dari file terenkripsi. Perbedaan file yang harus diangkut itulah yang merupakan perbedaan dari file yang terenkripsi dan tidak terenkripsi.

#### IV. KESIMPULAN

- 1. Software Open Puff digunakan untuk membuat keamanan pesan dengan teknik steganografi.
- 2. Agar dapat terjadi steganografi, besar file carriernya dari pengangkut harus lebih besar dengan file message yang harus diangkut.
- 3. Besar file message semakin besar maka semakin besar pula yang harus diangkut.
- 4. Besar jenis file carrier dengan daya angkut yang lebih besar adalah jenis .MP3.
- 5. Pengaruh file enkripsi pada file message dalam proses steganografi terletak pada besar ukuran yang harus diangkut lebih besar dari besar file message yang tidak terenkripsi.

# V. **REFERENSI**

- 1. <u>http://id.wikipedia.org/wiki/Steganografi</u>
- 2. <u>http://elista.akprind.ac.id/staff/catur/Multimedia/12-Keamanan%20Multimedia.pdf</u>