MULTI FRIENDLY SECRET IMAGE SHARING BERDASARKAN PADA OPERASI BOOLEAN DENGAN BASIS (N,N)

Ruby Abdullah
Department of Informatics Faculty of Mathematic and Natural Sciences
Sebelas Maret University
Surakarta
rubyabdullah@student.uns.ac.id

Heri Prasetyo
Department of Informatics Faculty of Mathematic and Natural Sciences
Sebelas Maret University
Surakarta
heri.prasetyo@staff.uns.ac.id

Bambang Harjito
Department of Informatics Faculty of Mathematic and Natural Sciences
Sebelas Maret University
Surakarta
bambang_harjito@staff.uns.ac.id

***Abstract*- In recent years, advancements in network bandwidth and rapid developments in digital communication have driven the growing trend for digital data transmission. Images are one of the most commonly used data types worldwide. The SIS (Secret Image Sharing) method was developed to overcome this problem. In previous studies, the MSIS (Multi Secret Image Sharing) method with a base (n, n) threshold with boolean operations has been developed, namely by inputting as many as n secret images and producing n shared images with hidden meanings. In this research there is a drawback that is the attacker can know that the data is hidden. To overcome this problem the MFSIS (Multi Friendly Secret Image Sharing) method is proposed. The method is a combination of MSIS and FSS (Friendly Secret Sharing) so as to produce a shared image that is familiar and difficult to identify by attackers.**

I. Pendahuluan

Dalam beberapa tahun terakhir, kemajuan dalam *bandwidth* jaringan dan perkembangan pesat dalam komunikasi digital telah mendorong tren yang berkembang untuk transmisi data digital. Gambar adalah salah satu tipe data yang paling sering digunakan secara mendunia. Namun, terdapat beberapa isu dalam mentransmisi gambar melalui internet. Beberapa gambar ada yang rahasia yaitu gambar yang hanya boleh diketahui oleh pihak-pihak tertentu saja. Tetapi jika hanya satu pihak yang diberikan akses ke data, maka kehilangan gambar yang disengaja atau tidak disengaja dapat terjadi. Di sisi lain, jika beberapa pihak mendapatkan bagian dari *secret image*, maka kerjasama mereka semua adalah kelemahan untuk rekontruksi secret image. Kekhawatiran ini telah menunjang adanya penelitian tentang skema *secret sharing* (SS).

Penelitian pertama tentang SIS (*secret image sharing*) dikenalkan oleh Naor dan Shamir, (1995) yang diketahui dengan nama VSS (*visual secret sharing*). Pada skema VSS *secret image* diubah menjadi beberapa bayangan yang dimana setiap partisipan dapat mem-fotokopi transparansi dan menumpuknya pada *overhead projector* untuk mendekode secara *visual secret image*. Pada penelitian (Kabirirad and Eslami, 2018) mengembangkan *multi secret image sharing* (MSIS) dengan konsep *(t,n)-threshold*. Pada penelitian tersebut *secret image* sebanyak t gambar didistribusikan sebanyak n gambar menjadi *shared image* dengan menggunakan operasi boolean. Gambar mampu dikembalikan dengan menggunakan sebanyak t atau lebih dari *shared image*. Pada penelitian Kabirirad dan Eslami, (2019) mengembangkan skema *multi secret image sharing* (MSIS) dengan konsep *(n,n)-threshold*. Pada skema MSIS dengan *(n,n)-threshold* telah meningkatkan performa sekuritas dari skema sebelumnya dengan basis yang sama karena setiap shared image memiliki ketergantungan linier. Sehingga apabila *shared image* berjumlah lebih sedikit dari n maka gambar tidak dapat dikembalikan.

Terdapat permasalahan dalam metode-metode sebelumnya yaitu *shared image* yang dihasilkan dapat diidentifikasi sebagai gambar yang disembunyikan oleh penyerang karena gambar tidak memiliki makna. Pada penelitian (Prasetyo dan Simatupang, 2019) mengembangkan skema Friendly-Progressive Secret Sharing, yaitu dengan menyisipkan sharing image ke dalam cover image. Skema tersebut merupakan pengembangan dari PSS(Progressive Secret Sharing) menjadi FSS(Friendly Secret Sharing) sehingga hasil sharing menjadi gambar bermakna sehingga mampu merubah persepsi dari penyerang. Hanya saja pada penelitian ini hanya mampu diimplementasikan untuk satu gambar.

Skema MFSIS (Multi Friendly Secret Image Sharing) diajukan untuk menjawab masalah diatas dan mampu diimplementasikan pada gambar sejumlah n gambar. Skema MFSIS yang diajukan menggabungkan metode MSIS(Kabirirad dan

Eslami, 2019) dan metode Friendly Progressive Secret Sharing(Prasetyo dan Simatupang, 2019).

II. Penelitian Terkait

Telah banyak penelitian sebelumnya yang bertujuan untuk menjawab masalah tersebut. Pada penelitian Kabirirad dan Eslami, (2018), skema model MSIS (*Multi Secret Image Sharing)* dengan basis *(t,n)* dan menggunakan operasi boolean digunakan untuk mengembangkan metode-metode yang sudah ada yaitu yang hanya berbasis *(2,n)* atau *(n,n)*. Pada skema ini dibagi menjadi dua yaitu *generate sharing image* dan *recovery secret image.* Pada fase *generate sharing image* mengambil input gambar yang ingin dienkripsi dengan banyak t untuk menghasilkan *shared image* sebanyak n dengan menggunakan *XOR operation* dan *pseudo random function*. Pada fase *recovery secret image* mengambil input gambar yang terenkripsi sebanyak n untuk menghasilkan gambar asli. Pada penelitian Kabirirad dan Eslami, (2019), skema model MSIS (*Multi Secret Image Sharing)* dengan basis *(n,n)* dan menggunakan operasi boolean adalah pengembangan dari metode sebelumnya yang berbasis *(n,n).* Pada skema ini dibagi menjadi dua yaitu *generate sharing image* dan *recovery secret image.* Pada fase *generate sharing image* mengambil input gambar yang ingin dienkripsi dengan banyak n untuk menghasilkan *shared image* sebanyak n dengan menggunakan *XOR operation*, *pseudo random function, block chipper*. Pada fase *recovery secret image* mengambil input gambar yang terenkripsi sebanyak n untuk menghasilkan n gambar asli, dapat dilihat pada Algoritma 2.1 untuk *sharing procedure* dan Algoritma 2.2 untuk *recovery procedure.*

Algoritma 2.1: Kabirirad dan Eslami,(2019) *Sharing Procedure*

Input: sejumlah *n secret image* $I_0, I_1, \ldots, I_{n-1}$ dengan ukuran $M \times N \times C$

Output: sejumlah *n shared image* $S_0, S_1, \ldots, S_{n-1}$ dengan ukuran $M \times N \times C$

Proses:

1. Membuat *pseudo random* citra *R* dengan perhitungan sebagai berikut:
   $K = H(I_0 \oplus I_1 \oplus \ldots \oplus I_{n-1})$,
   $R = chiper_k(I_0 \oplus I_1 \oplus \ldots \oplus I_{n-1})$
   dimana H adalah fungsi hash dan cipher adalah fungsi *block cipher* yang melakukan generasi sebuah output *pseudo random* dengan $I_0 \oplus I_1 \oplus \ldots \oplus I_{n-1}$ sebagai input dan K sebagai key. Output dari fungsi cipher adalah matriks berukuran $M \times N \times C$
2. Citra dari *pseudo random* R_i, $i = 1, 2, \ldots, n$ dibangkitkan dengan operasi boolean(XOR dan *circular shift)* sebagai berikut:

$$R_i(x,y) = R(x - d_1 i(\mathrm{mod}M), y - d_2 i(\mathrm{mod}N), z - d_3 i(\mathrm{mod}C)) \oplus R(x + d_1 i(\mathrm{mod}M), y + d_2 i(\mathrm{mod}N), z + d_3 i(\mathrm{mod}C))$$

dimana $1 \le d_1 \le \frac{M-1}{2n}$, $1 \le d_2 \le \frac{N-1}{2n}$ dan $1 \le d_3 \le \frac{C-1}{2n}$

3. *Shared image* S_{ti}, $i = 0, 1, \ldots, n-1$ dibangkitkan dengan cara sebagai berikut:

$$S_{ti} = \begin{cases} I_0 \oplus R_1 \oplus R_2 & i = 0 \\ I_1 \oplus R_2 \oplus R_3 & i = 1 \\ \vdots & \vdots \\ I_{n-2} \oplus R_{n-1} \oplus R_n & i = n-2 \\ I_{n-1} \oplus R_n \oplus R_1 & i = n-1 \end{cases}$$

Algoritma 2.2: Kabirirad dan Eslami,(2019) *Recovery Procedure*

Input: sejumlah *n secret image* $I_0, I_1, \ldots, I_{n-1}$ dengan ukuran $M \times N \times C$

Output: sejumlah *n shared image* $S_0, S_1, \ldots, S_{n-1}$ dengan ukuran $M \times N \times C$

Proses:

1. Membuat *pseudo random* citra *R* dengan perhitungan sebagai berikut:
   $K = H(I_0 \oplus I_1 \oplus \ldots \oplus I_{n-1})$,
   $R = chiper_k(I_0 \oplus I_1 \oplus \ldots \oplus I_{n-1})$
   dimana H adalah fungsi hash dan cipher adalah fungsi *block cipher* yang melakukan generasi sebuah output *pseudo random* dengan $I_0 \oplus I_1 \oplus \ldots \oplus I_{n-1}$ sebagai input dan K sebagai key. Output dari fungsi cipher adalah matriks berukuran $M \times N \times C$
2. Citra dari *pseudo random* R_i, $i = 1, 2, \ldots, n$ dibangkitkan dengan operasi boolean(XOR dan *circular shift)* sebagai berikut:

$$R_i(x,y) = R(x - d_1 i(\mathrm{mod}M), y - d_2 i(\mathrm{mod}N), z - d_3 i(\mathrm{mod}C)) \oplus R(x + d_1 i(\mathrm{mod}M), y + d_2 i(\mathrm{mod}N), z + d_3 i(\mathrm{mod}C))$$

dimana $1 \le d_1 \le \frac{M-1}{2n}$, $1 \le d_2 \le \frac{N-1}{2n}$ dan $1 \le d_3 \le \frac{C-1}{2n}$

3. *Shared image* S_{ti}, $i = 0, 1, \ldots, n-1$ dibangkitkan dengan cara sebagai berikut:

$$S_{ti}=\begin{cases} I_0 \oplus R_1 \oplus R_2 & i=0 \\ I_1 \oplus R_2 \oplus R_3 & i=1 \\ \vdots & \vdots \\ I_{n-2} \oplus R_{n-1} \oplus R_n & i=n-2 \\ I_{n-1} \oplus R_n \oplus R_1 & i=n-1 \end{cases}$$

Pada penelitian Prasetyo dan Simatupang, (2019), skema model FSS (Friendly-Progressive Secret Sharing) adalah merupakan pengembangan metode dari PSS (Progressive Secret Sharing). PSS menghasilkan sharing image berupa gambar yang noisy sedangkan pada FSS menghasilkan gambar yang lebih familiar. Skema tersebut digunakan untuk menjawab permasalahan dimana seorang penyerang apabila mendapatkan sebuah data gambar tersebut dalam bentuk noisy akan menyadari bahwa itu rahasia, sedangkan apabila gambar tersebut lebih familiar maka penyerang terdapat kemungkinan tidak menyadari bahwa gambar tersebut rahasia.

III. Metodologi

Diagram skematis penelitian pada tugas akhir ini dapat dilihat pada Gambar 3.1. Tahap penelitian akan meliputi tiga bagian yaitu persiapan data input, perancangan model skema MFSIS, dan testing model.

Gambar 3.1. Diagram skematis penelitian

A. Persiapan Data Input

Data yang digunakan sebagai data input pada skema MFSIS (Multi Friendly Secret Image Sharing) berupa image digital yang berekstensi .tif.

B. Perancangan Model Skema MFSIS

Pada perancangan model skema MFSIS (Multi Friendly Secret Image Sharing) dibagi menjadi dua yaitu sharing procedure dan recovery procedure.

1. Sharing Procedure

Sharing procedure adalah prosedur untuk melakukan generasi shared image. Algoritma sharing procedure bisa dilihat pada Algoritma 3.1.

Algoritma 3.1: *Sharing Procedure*

Input: *n secret image* $I_0,\ I_1,\ \ldots,\ I_{n-1}$ dan *cover image* C_o
Output: *n shared image* $S_0,\ S_1,\ \ldots,\ S_{n-1}$
Proses:

1. Membuat *pseudo random* citra *R* dengan perhitungan sebagai berikut:
   $K = H(I_0 \oplus I_1 \oplus \ldots \oplus I_{n-1})$,
   $R = chiper_k(I_0 \oplus I_1 \oplus \ldots \oplus I_{n-1})$
   dimana $H(\cdot)$ adalah fungsi *hash* dan cipher adalah fungsi *block cipher* yang melakukan generasi sebuah output *pseudo random* dengan $I_0 \oplus I_1 \oplus \ldots \oplus I_{n-1}$ sebagai input dan K sebagai key. Output dari fungsi cipher adalah matriks berukuran $M \times N \times C$
2. Citra dari *pseudo random*
3. $R_i,\ i = 1, 2,\ \ldots,\ n$ dibangkitkan dengan operasi boolean(XOR dan *circular shift)* sebagai berikut:

$$R_i(x,y,z) = R(x - d_1 i(modM), y - d_2 i(modN), z - d_3 i(modC)) \oplus R(x + d_1 i(modM), y + d_2 i(modN), z + d_3 i(modC))$$

   dimana $1 \le d_1 \le \frac{M-1}{2n}$, $1 \le d_2 \le \frac{N-1}{2n}$ dan $1 \le d_3 \le \frac{C-1}{2n}$
4. *Shared image* $S_{ti}\ i = 0, 1,\ \ldots,\ n-1$ digenerasi dengan cara sebagai berikut:

$$S_{ti}=\begin{cases} I_0 \oplus R_1 \oplus R_2 & i=0 \\ I_1 \oplus R_2 \oplus R_3 & i=1 \\ \vdots & \vdots \\ I_{n-2} \oplus R_{n-1} \oplus R_n & i=n-2 \\ I_{n-1} \oplus R_n \oplus R_1 & i=n-1 \end{cases}$$

5. Menyisipkan citra shared kedalam gambar bermakna dengan cara sebagai berikut:
   $S_i = S_{ti} + C_o$
   Dengan mengonversi gambar S_i dari 8 bit menjadi 4 bit

2. Recovery Procedure

Recovery procedure adalah prosedur untuk melakukan pemulihan *shared image* menjadi *secret image*. Algoritma sharing procedure bisa dilihat pada Algoritma 3.2.

Algoritma 3.2: *Recovery Procedure*

Input: *n shared image* $S_0,\ S_1,\ \ldots,\ S_{n-1}$
Output: *n secret image* $I_0,\ I_1,\ \ldots,\ I_{n-1}$
Proses:

1. Mengambil *shared image* S_{ti} didalam S_i kemudian merubahnya menjadi 16 *bit*.
   $S_i \rightarrow S_{ti}$
2. Membuat *pseudo random* citra *R* dengan perhitungan sebagai berikut:
   $K = H(S_{t0} \oplus S_{t1} \oplus \ldots \oplus S_{tn-1})$,
   $R = chiper_k(S_{t0} \oplus S_{t1} \oplus \ldots \oplus S_{tn-1})$

3. Citra dari *pseudo random* R_i, $i = 1, 2, \ldots, n$ digenerasi dengan operasi boolean(XOR dan *circular shift)* sebagai berikut:

$$R_i(x,y,z) = R(x - d_1 i(modM), y - d_2 i(modN), z - d_3 i$$
$$\oplus R(x + d_1 i(modM), y + d_2 i(modN), z + d_3 i(m$$

4. *Secret image* I_i, $i = 0, 1, \ldots, n-1$ digenerasi dengan cara sebagai berikut:

$$I_i = \begin{cases} S_{t0} \oplus R_1 \oplus R_2 & i = 0 \\ S_{t1} \oplus R_2 \oplus R_3 & i = 1 \\ \vdots & \vdots \\ S_{tn-2} \oplus R_{n-1} \oplus R_n & i = n-2 \\ S_{tn-1} \oplus R_n \oplus R_1 & i = n-1 \end{cases}$$

C. Testing Modul

Testing dilakukan dengan membandingkan hasil dari skema MFSIS (*Multi Friendly Secret Image Sharing*) dengan metode MSIS (*Multi Secret Image Sharing*) sebelumnya(Kabirirad dan Eslami, 2019).

IV. Hasil dan Analisis Percobaan

A. Proses *Sharing Procedure* MFSIS

Implementasi *sharing procedure* membutuhkan input sejumlah *n secret image* serta *cover image*. Lihat Gambar 4.1. Proses *sharing procedure* dibagi menjadi beberapa langkah yaitu:


(1)


(2)


(3)


(4)

Gambar 4.1. Input *Image I(i)* : (1)(2)(3) *Secret Image,* (4) *Cover Image*

Pertama adalah melakukan generasi *pseudo random image R* dengan menggunakan algoritma *Block Cipher* dengan key berupa operasi XOR antar *secret image.* Diketahui matriks I_1, I_2, I_3 merupakan matriks *secret image* berukuran $512 \times 512 \times 3$ sebagai berikut:

$$I_1 = \begin{bmatrix} [71\ 150\ 164] & \cdots & [118\ 188\ 179] \\ \vdots & \ddots & \vdots \\ [12\ 11\ 9] & \cdots & [2\ 5\ 4] \end{bmatrix}$$

$$I_2 = \begin{bmatrix} [0\ 0\ 101] & \cdots & [107\ 0\ 120] \\ \vdots & \ddots & \vdots \\ [0\ 0\ 93] & \cdots & [202\ 190\ 100] \end{bmatrix}$$

$$I_3 = \begin{bmatrix} [0\ 0\ 101] & \cdots & [181\ 0\ 120] \\ \vdots & \ddots & \vdots \\ [0\ 0\ 93] & \cdots & [171\ 200\ 198] \end{bmatrix}$$

Maka untuk mendapatkan nilai keynya menggunakan:

$$K = H\left(\begin{bmatrix} [71\ 150\ 164] & \cdots & [168\ 188\ 179] \\ \vdots & \ddots & \vdots \\ [12\ 11\ 9] & \cdots & [157\ 77\ 17] \end{bmatrix}\right)$$

$$K = \begin{bmatrix} 12 & 216 & 244 & 35 & 252 & 39 & \ldots & 241 \end{bmatrix}$$

Untuk memperkecil komputasi maka input hasil dari XOR tersebut diubah ke grayscale menggunakan fungsi:

$$I_{result} = 0.299 \times x + 0.587 \times y + 0.114 \times z$$

$$R \approx chiper_k\left(\begin{bmatrix} 128 & \cdots & 181 \\ \vdots & \ddots & \vdots \\ 11 & \cdots & 94 \end{bmatrix}\right)$$

$$R = \begin{bmatrix} 4 & \cdots & 21 \\ \vdots & \ddots & \vdots \\ 163 & \cdots & 94 \end{bmatrix}$$

Kemudian melakukan generasi *Image* R_i dengan $i = 1, 2, 3, \ldots, n$ menggunakan fungsi pada Algoritma 3.1 pada proses yang kedua. Diketahui matriks R berukuran $512 \times 512 \times 3$ yang akan digunakan untuk generasi *image* R_1, R_2, R_3 :

$$R_i(x,y,z) = R(x - d_1 i(modM), y - d_2 i(modN), z - d_3 i(modC))$$
$$\oplus R(x + d_1 i(modM), y + d_2 i(modN), z + d_3 i(modC))$$

$$R_1 = \begin{bmatrix} [0\ 0\ 0] & \cdots & [0\ 0\ 0] \\ \vdots & \ddots & \vdots \\ [0\ 0\ 0] & \cdots & [0\ 0\ 0] \end{bmatrix}$$

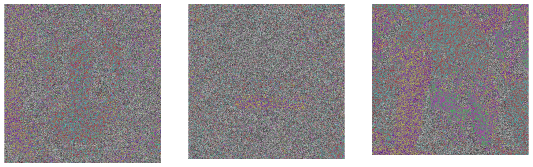
$$R_2 = \begin{bmatrix} [96\ 96\ 96] & \cdots & [116\ 116\ 116] \\ \vdots & \ddots & \vdots \\ [195\ 195\ 195] & \cdots & [3\ 3\ 3] \end{bmatrix}$$

$$R_3 = \begin{bmatrix} [20\ 20\ 20] & \cdots & [203\ 203\ 203] \\ \vdots & \ddots & \vdots \\ [145\ 145\ 145] & \cdots & [141\ 141\ 141] \end{bmatrix}$$

Kemudian melakukan generasi *sharing image* S_{ti} dengan $i = 1, 2, 3, \ldots, n$ menggunakan fungsi pada Algoritma 3.1 pada proses yang ketiga. Diketahui matriks R_1, R_2, R_3 berukuran

$(512 \times 512 \times 3)$ yang akan digunakan untuk generasi *shared image* S_{t1}, S_{t2}, S_{t3} :

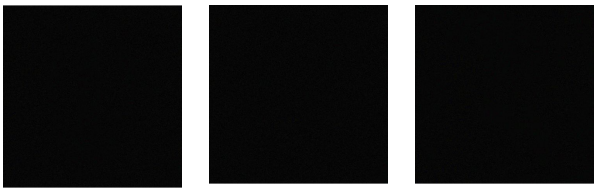
$$S_{ti} = \begin{cases} I_0 \oplus R_1 \oplus R_2 & i = 0 \\ I_1 \oplus R_2 \oplus R_3 & i = 1 \\ \vdots & \vdots \\ I_{n-2} \oplus R_{n-1} \oplus R_n & i = n-2 \\ I_{n-1} \oplus R_n \oplus R_1 & i = n-1 \end{cases}$$



(1) (2) (3)

Gambar 4.2. *St(i)*(1)(2)(3)*Shared Image* S_{ti}

Kemudian melakukan penyisipan *sharing image* S_{ti} dengan $i = 1, 2, 3, \ldots, n$ ke dalam *cover image*. Penyisipan dilakukan dengan merubah S_{ti} yang semula 16 bit menjadi 4 bit kemudian disisipkan ke *Cover Image* Gambar 4.1 bagian (4). Hasil pada tahap ini dapat dilihat pada Gambar 4.3 dan Gambar 4.4.



(1) (2) (3)

Gambar 4.3. *St(i)*(1)(2)(3) Hasil Konversi 16 bit menjadi 4 bit

(1) (2) (3)

Gambar 4.3. *S(i)*(1)(2)(3) Hasil Konversi 16 bit menjadi 4 bit

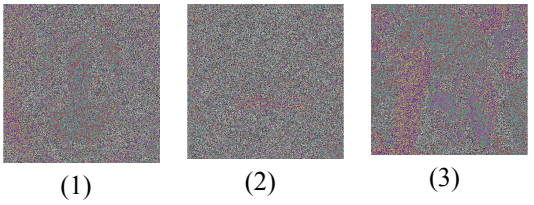
B. Proses *Recovery Procedure MFSIS*

Implementasi *recovery procedure* membutuhkan input berupa sejumlah n *secret image* serta cover image. Lihat Gambar 4.5. Proses sharing procedure dibagi menjadi beberapa langkah yaitu:

(1) (2) (3)

Gambar 4.5.Input *Image*:*S(i)*(1)(2)(3) *Sharing Image*

Pertama melakukan ekstraksi *shared image* dari *cover image* dengan mengambil nilai *shared image* yang tersimpan di dalam Gambar 4.5. Kemudian merubahnya menjadi 16 bit yang semula 4 bit. asil pada tahap ini dapat dilihat pada Gambar 4.6.



(1) (2) (3)

Gambar 4.6.*St(i)* (1)(2)(3)*Shared Image* S_{ti}

Kemudian melakukan generasi *pseudo random image R* dengan menggunakan algoritma *Block Cipher* dengan key berupa operasi XOR antar *secret image.* Diketahui matriks S_{t1}, S_{t2}, S_{t3} merupakan matriks *secret image*

berukuran $512 \times 512 \times 3$ sebagai berikut:

$$S_{t1} = \begin{bmatrix} \begin{bmatrix} 39 & 246 & 196 \end{bmatrix} & \cdots & \begin{bmatrix} 2 & 200 & 199 \end{bmatrix} \\ \vdots & \ddots & \vdots \\ \begin{bmatrix} 207 & 200 & 202 \end{bmatrix} & \cdots & \begin{bmatrix} 1 & 6 & 7 \end{bmatrix} \end{bmatrix}$$

$$S_{t2} = \begin{bmatrix} \begin{bmatrix} 116 & 116 & 17 \end{bmatrix} & \cdots & \begin{bmatrix} 212 & 191 & 199 \end{bmatrix} \\ \vdots & \ddots & \vdots \\ \begin{bmatrix} 82 & 82 & 15 \end{bmatrix} & \cdots & \begin{bmatrix} 186 & 14 & 93 \end{bmatrix} \end{bmatrix}$$

$$S_{t3} = \begin{bmatrix} \begin{bmatrix} 20 & 20 & 113 \end{bmatrix} & \cdots & \begin{bmatrix} 126 & 203 & 179 \end{bmatrix} \\ \vdots & \ddots & \vdots \\ \begin{bmatrix} 145 & 145 & 204 \end{bmatrix} & \cdots & \begin{bmatrix} 38 & 69 & 75 \end{bmatrix} \end{bmatrix}$$

Maka untuk mendapatkan nilai keynya menggunakan:

$$K = H(S_{t1} \oplus S_{t2} \oplus S_{t3})$$

$$K = \begin{bmatrix} 12 & 216 & 244 & 35 & 252 & 39 & \cdots & 241 \end{bmatrix}$$

Untuk menghasilkan *pseudo random image* R maka:

$$R = chiper_k(I_1 \oplus I_2 \oplus I_3)$$

$$R \approx chiper_k\left(\begin{bmatrix} 128 & \cdots & 181 \\ \vdots & \ddots & \vdots \\ 11 & \cdots & 94 \end{bmatrix}\right)$$

$$R = \begin{bmatrix} 4 & \cdots & 21 \\ \vdots & \ddots & \vdots \\ 163 & \cdots & 94 \end{bmatrix}$$

Kemudian melakukan generasi *Image* R_i dengan $i = 1, 2, 3, \ldots, n$ menggunakan fungsi pada Algoritma 3.1 pada proses yang kedua. Hasil R_i sama dengan hasil R_i pada *sharing procedure.*

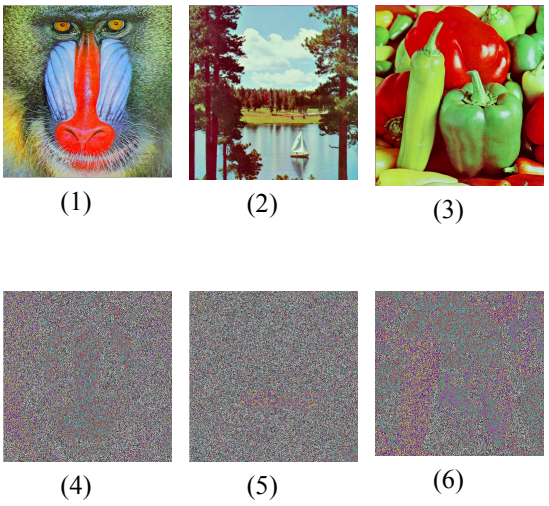
Kemudian melakukan generasi *sharing image* S_{ti} dengan $i = 1, 2, 3, \ldots, n$ menggunakan fungsi pada Algoritma 3.1 pada proses yang ketiga. Hasilnya dapat dilihat pada Gambar 4.7.

(1) (2) (3)

Gambar 4.7. *I(i)*(1)(2)(3)*Secret Image I*

C. *Analisa Percobaan*

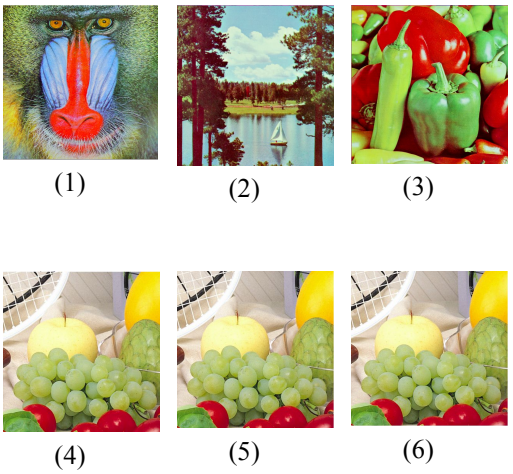
Terdapat perbedaan yang sangat signifikan dari metode sebelumnya yaitu penelitian dari (Kabirirad and Eslami, 2018) yang mengembangkan *multi secret image sharing* (MSIS) dengan konsep *(n,n)-threshold,* dengan Metode skema yang saya ajukan yaitu MFSIS (*Multi Friendly Secret Image Sharing*). Dapat dilihat dari segi visual bahwa metode yang diajukan MFSIS Gambar 4.9 menghasilkan *shared image* yang lebih familiar dan tidak mudah dideteksi oleh penyerang daripada metode sebelumnya Gambar 4.8.



(1) (2) (3)

(4) (5) (6)

Gambar 4.8. *I(i)*(1)(2)(3)*Secret Image S(i)*(4)(5)(6) *shared image*(Kabirirad dan Eslami,2019)



(1) (2) (3)

(4) (5) (6)

Gambar 4.9. *I(i)*(1)(2)(3) *Secret Image S(i)*(4)(5)(6) *Sharing Image,* MFSIS

V. Kesimpulan

Skema MFSIS(Multi Friendly *Secret Image* Sharing) dengan operasi boolean untuk masalah (n,n) threshold dihasilkan pada penelitian ini. Skema MFSIS terdiri dari sharing procedure yaitu proses untuk menghasilkan sharing image dengan input secret image dengan menggunakan operasi boolean XOR kemudian menyisipkannya ke dalam cover image dan recovery procedure yaitu proses untuk mengembalikan ulang gambar hasil sharing procedure menggunakan operasi boolean XOR. Skema MFSIS dapat mengatasi masalah yang terdapat pada penelitian sebelumnya yang dimana hasil dari sharing mampu menyembunyikan makna sebuah image tersembunyi.

VI. Daftar Pustaka

Gonzalez, R.C., 2009. Digital Image Processing. Pearson Education.

Kabirirad, S., Eslami, Z., 2019. Improvement of (n, n)-multi-secret image sharing schemes based on Boolean operations. J. Inf. Secur. Appl. 47, 16–27. https://doi.org/10.1016/j.jisa.2019.03.018

Kabirirad, S., Eslami, Z., 2018. A (t,n)-multi secret image sharing scheme based on Boolean operations. J. Vis. Commun. Image Represent. 57, 39–47. https://doi.org/10.1016/j.jvcir.2018.10.014

Liu, Y.-X., Yang, C.-N., Chou, Y.-S., Wu, S.-Y., Sun, Q.-D., 2018. Progressive (k,n) secret image sharing Scheme with meaningful shadow images by GEMD and RGEMD. J. Vis. Commun. Image Represent. 55, 766–777. https://doi.org/10.1016/j.jvcir.2018.08.003

Naor, M., Shamir, A., 1995. Visual cryptography, in: De Santis, A. (Ed.), Advances in Cryptology — EUROCRYPT'94, Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 1–12.

Prasetyo, H., Simatupang, J.W., 2019. XOR-ed Based Friendly-Progressive Secret Sharing, in: 2019 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS). Presented at the 2019 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), pp. 1–2. https://doi.org/10.1109/ISPACS48206.2019.8986406

Yan, X., Lu, Y., Liu, L., Wang, S., 2018. Partial secret image sharing for (k,n) threshold based on image inpainting. J. Vis. Commun. Image Represent. 50, 135–144. https://doi.org/10.1016/j.jvcir.2017.11.012