RC4 Technique in Visual CryptographyRGB Image Encryption

Andysah Putera Utama Siahaan
Faculty of Computer Science
Universitas Pembangunan Panca Budi
Jl. Jend. Gatot Subroto Km. 4,5 Sei Sikambing, 20122, Medan, Sumatera Utara, Indonesia

Abstract– *In this study, we are doing a cryptography scheme which can modify the visualization of pictures. The image protection is very critical. It does not alter the value of the header and the metadata. We are just trying to modify the color intensities. Every image consists of three color layers. There are red, green and blue (RGB). Each layer has numbers which represent the intensity. RC4 is used to change the color intensity in every layer. The encryption process it to manipulate the integer number and produce the encrypted value. We determine how many layers is going to be encrypted. The fuzziness of the encrypted image depends on how many layer are included. The visualization will be unrecognized after the encryption. It changes to a noisy picture. This method makes the image content secure and undetected.*

Keywords *- Cryptography, Image Encryption, Steganography*

I. INTRODUCTION

In the modern era, the picture can be used as an evidence of the crime. It proves what someone has done. It contains the confidential information. If it fell into the perpetrator, it will be a big problem. We must hide the information from being intercepted. There are various techniques should be used to protect the confidential image data from unauthorized access [2]. Visual Cryptography can be combined with Steganography, but, in Steganography, it usually changes the bit pattern for increasing the security of the image [4]. Our target is to manipulate the image visualization. The image consists of integer number that represents the color intensity. The method proposes is the encryption that transforms the original image into to coded image. So it is hard to understand. In the other word, someone might take it, but they do not understand, or they cannot decrypt the content of the picture without having the key.

II. RELATED WORK

A. *Visual Cryptography*

Visual Cryptography is one of the cryptography methods to hide the information. It usually uses pictures and other multimedia [2][6]. The example is when we subscribe to the TV station. If we late to pay the subscription fee, they will encrypt the broadcast. We still can see on TV, but we do not know what is the meaning of the pattern showed. Yes, of course, they scramble the broadcast. The previous example is one of the visual cryptography implementation. The encryption in visual cryptography does not use hard mathematical computations to perform encryption and decryption. The original information going to encrypt is a secret message. After encryption, ciphers are generated and referred as shares. The part of a secret in scrambled form is known as a share. The basic idea in visual cryptography is to share the secret among participants. It is divided into several pieces of images. They are shares. These are distributed among the participants. To reveal the first secret, each participant provides his share. There is a various scheme of visual cryptographic available. There is 2 out of 2 visual cryptography where the message is split into two images. These two shares must participate to retrieve the secret message [1].

Adi Shamir invented the secret sharing concept of visual cryptography in 1979. He stated that it is divided into several pieces and easily reconstructive from any pieces. He claimed that data is protected by encryption, but the key used for encryption could not be covered. He wrote that secret sharing aimed to protect the keys used to encryption. Depending on Shamir, the scheme is k out of n secret sharing scheme [3]. Figure 1 shows the example of visual cryptography scheme.

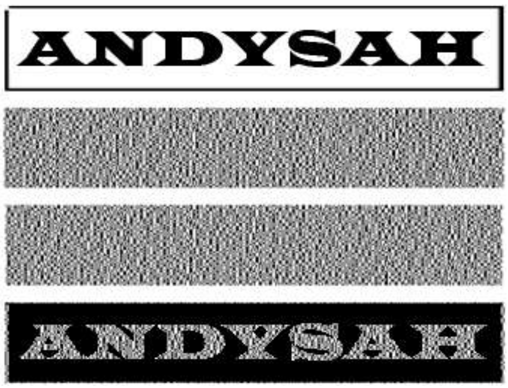

Figure 1 : Example of visual cryptography

In figure 1, the graying effect is noted in the background. Additional black pixels in the background are forming some pattern and giving rise to gray effect. Owing to this effect, the meaning of secret remains undisturbed. Hence, the graying effect has no impact on the first secret. Another scheme knew as 2 out of n scheme. The secret is structured in exactly n shares. At least, two shares must participate. The third scheme of VC is formally known as K out of n scheme in which the secret is structured in exactly n parts. Multiple shares are generated out of visual cryptography. An extended version of the third visual cryptography scheme is n out of n where in secret is split into several n shares. They must participate while revealing the secret. Hierarchical visual cryptography abbreviated is the specialization of visual cryptographic schemes. It is based upon basic 2 out of 2 visual cryptographic schemes. The secrecy increase as the secret message has been encrypted. There are numerous authentication systems available based upon biometric, passwords, but each authentication system is having pitfalls related to the confidentiality of data.

B. RC4

The RC4 algorithm is one of symmetric cryptography algorithms that combines the plaintext with the keystream which is normally used for encryption and decryption. RC4 is derived from the RSA Data Security, Inc. In symmetric ciphers, both the encryption and decryption use the same keys. It is designed to be easily performed even in large amounts of data. Symmetric ciphers can operate in block or stream mode. In block mode, the message will be split into several fixed size blocks and each block will be encrypted one by one while in stream mode the message will be encrypted from the first character to the end of the message [5].

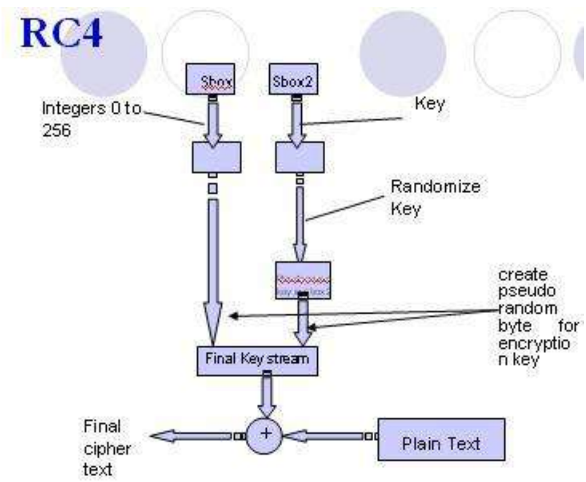

Figure 2 : RC4 ciphertext production

Figure 2 shows the step of RC4 algorithm. The key is produced by permutation of the two SBOXes. RC4 emphasize the key establishment.

III. PROPOSED WORK

The previous works used the grayscale image as the secret picture and share the key to doing the encryption. This study is not resulting in the shared key to decrypt the picture, but we combine the RC4 technique. The key is still alphabetic key, and it can be generated by a pseudorandom generator or even type them on a keyboard. Yes, of course, we are required to save the key from being stolen. Every picture is composed of three layers, such as red, green and blue. The pixel data in every is encrypted by the key created (Figure 3)

R/G/B	Key	Cipher
P1	K1	C1
P2	K2	C2
P3	K3	C3
P4	K4	C4
P5	K5	C5
P6	K6	C6
...	...	...
Pn	Kn	Cn

Figure 3 : Encryption Process

The pixel P1 to Pn are encrypted by using K1 to Kn. They are encrypted respectively. The key K1 to Kn are obtained from the pseudorandom number generated by the SBOX. There three parts of the encryption process. The first is for the red composition while the second and the last are green and blue. Figure 4 shows the flow chart of overall process. The original image must be extracted to Red, Green and Blue data. Once after the extraction is complete, the pixels in each layer is encrypted with the key generated before.After the encryption the layer must be combine again to produce the visualization.

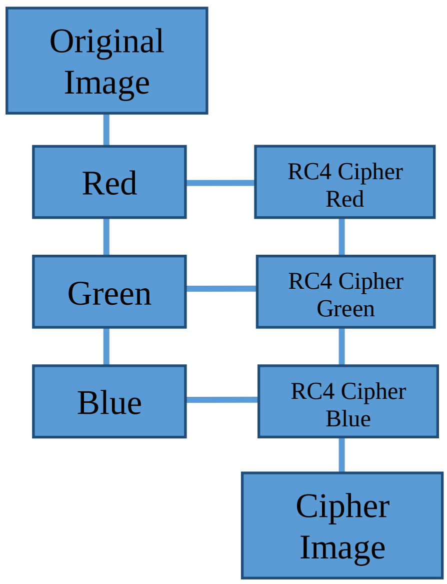

Figure 4 : The flow of the encryption

TESTING AND IMPLEMENTATION

In this implementation, we do the test with a 10x10 pixel image. Figure 5 shows the original image before the encryption.

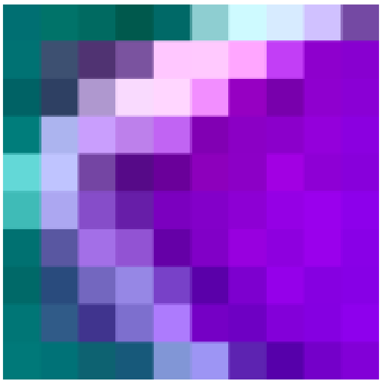

Figure 5 : A 10 x 10 pixel

The original image needs to be split into three components. The following tables show the extraction of the pixels.

Table 1 : The red layer color intensities

RED PIXEL									
25	26	15	4	36	154	213	218	204	109
25	63	74	115	252	252	255	177	125	119
12	48	170	244	255	228	135	105	126	120
37	171	192	178	179	114	122	122	130	121
126	189	107	76	95	126	123	144	124	118
96	168	122	89	109	114	123	129	134	121
24	85	152	134	87	112	133	123	134	117
13	44	109	144	110	76	108	130	114	115
29	54	58	119	163	102	93	112	113	122
29	38	39	38	129	152	81	72	99	113

Table 2 : The green layer color intensities

GREEN PIXEL									
108	112	102	85	102	203	248	234	199	85
110	82	60	93	208	209	182	102	48	54
95	67	158	223	221	161	64	28	49	57
121	184	169	143	124	40	36	33	41	49
212	200	83	43	41	48	37	59	35	46
182	173	97	60	58	43	34	34	38	38
110	94	127	102	36	38	46	28	36	34
101	78	112	145	86	32	40	49	32	34
115	93	64	121	139	58	26	33	32	41
117	111	97	89	154	158	63	36	44	51

Table 3 : The blue layer color intensities

BLUE PIXEL									
112	103	94	76	101	207	254	255	253	161
113	112	111	158	255	254	255	244	204	206
99	99	206	254	255	254	192	170	205	210
121	237	251	233	241	177	195	201	217	221
213	254	161	134	153	186	196	225	211	216
183	239	199	168	189	199	210	226	234	232
111	159	229	209	165	197	221	220	233	228
102	123	189	227	198	167	205	230	222	227
116	136	140	204	251	195	193	213	222	234
119	117	111	120	211	242	175	168	198	212

The data above is the raw data which are derived from the image. Colors are converted to integer numbers to represent the intensities. The numbers inside the cells are the value after conversion. Each layer must have its key as a password to perform the encryption. There are three keys used in this encryption.

Key 1 : HIDE
Key 2 : THIS
Key 3 : IMAGE

The SBOX have been generatedby the calculation the key and the plain image which has been converted to anarray of byte. Table 4 until Table 6 show the values of all SBOX.

Table 4 : The red layer SBOX

SBOX RED, KEY = "HIDE"							
94	7	216	32	6	247	108	80
16	126	65	73	228	141	41	46
95	1	232	10	158	250	84	113
57	19	251	55	148	218	245	192
130	25	219	67	79	21	135	44
99	75	35	179	243	127	36	131
230	96	122	136	202	147	194	14
150	90	105	83	109	39	110	206
77	184	111	185	169	182	40	125
121	3	102	181	238	180	11	151
115	143	168	233	2	26	209	60
164	47	124	217	9	17	4	24
89	78	167	38	69	42	153	162
166	71	170	56	72	101	48	98
59	176	116	171	117	128	104	146
193	68	253	138	129	178	120	205

239	161	144	28	33	152	195	177
235	174	76	225	187	23	106	43
231	160	196	198	172	139	201	87
224	12	54	132	212	214	156	199
165	107	246	254	234	66	222	91
220	88	100	226	123	22	190	200
81	112	191	58	114	213	118	215
183	204	85	5	18	155	197	252
119	154	242	142	70	173	0	134
82	103	227	61	8	157	221	203
62	244	64	149	188	140	137	210
74	255	29	229	163	53	240	248
86	34	31	175	20	50	63	133
236	52	249	97	159	15	189	49
27	208	223	93	241	92	145	186
211	30	37	237	207	45	51	13

Table 5 : The green layer SBOX

SBOX GREEN, KEY = "THIS"							
92	157	232	17	99	227	50	34
22	57	7	198	225	169	246	88
188	21	112	111	3	236	61	96
114	45	144	82	168	211	58	202
35	137	244	106	93	108	190	56
233	19	182	181	4	247	177	240
89	185	60	31	77	68	70	173
136	156	62	101	235	212	23	95
132	54	125	37	199	120	124	219
119	15	133	255	13	148	98	1
91	46	158	215	151	178	128	63
94	200	162	231	129	113	253	176
250	52	48	8	5	147	189	210
131	145	167	67	41	218	86	196
164	191	102	222	71	172	115	121
51	213	42	204	11	192	135	216
141	53	146	155	242	97	139	84
118	161	47	12	171	27	49	20
104	64	39	107	134	154	76	30
209	153	152	80	228	230	254	179
197	28	44	100	220	103	206	32
110	90	252	127	24	83	184	186
38	183	85	122	33	14	149	59
116	9	79	217	163	123	143	165
78	208	160	241	214	238	234	223
207	229	117	140	194	72	170	65

193	245	187	0	18	6	175	55
130	226	109	138	43	74	249	126
26	36	201	16	224	221	248	142
25	180	87	195	29	237	166	66
203	251	69	40	239	159	73	150
105	75	205	10	81	2	243	174

Table 6 : The blue layer SBOX

SBOX BLUE, KEY = "IMAGE"							
34	0	7	214	20	146	11	86
165	243	246	158	99	63	138	78
200	185	167	24	159	151	121	137
16	101	220	56	128	253	100	75
49	250	37	215	127	73	201	65
148	15	181	82	169	166	153	119
84	202	114	123	217	67	31	80
90	228	178	30	27	19	175	66
229	251	177	203	192	74	64	206
173	152	103	161	131	92	197	4
204	106	89	18	109	236	227	207
247	141	184	224	116	46	213	164
104	22	51	235	6	2	69	190
43	96	98	117	134	60	254	194
115	122	62	225	125	23	163	17
135	25	191	59	81	33	199	188
133	61	244	79	145	221	168	210
241	124	126	156	112	10	14	52
53	143	87	38	180	29	71	209
170	136	239	162	238	149	171	222
248	219	195	28	40	176	72	234
223	205	144	110	9	231	26	3
132	211	55	208	35	45	50	130
226	8	77	242	193	157	120	107
160	41	47	39	91	230	172	245
198	111	150	85	54	48	105	154
212	142	240	174	179	232	36	42
97	155	68	58	94	12	249	83
21	118	57	218	88	189	76	216
196	252	1	147	129	32	13	233
139	108	186	102	182	255	70	5
237	44	140	93	187	113	95	183

After all the SBOX are fully generated, we do the calculation to get the pseudorandom key. It is for the encryption and decryption later. Soon after we get it, the key can be used to encrypt the array of the pixel in every layer.

```
Red Pixel[0] = 25

i     =  0
j     =  0

i     =  (i + 1) mod 256
      =  0 + 1
      =  1

j     =  (j + S[i]) mod 256
      =  0 + 7
      =  7

S[i]  =  S[1] = 7
S[j]  =  S[7] = 80 then swap
S[i]  =  S[1] = 80
S[j]  =  S[7] = 7

t     =  (S[i] + S[j]) mod 256
      =  (80 + 7) mod 256
      =  87

K     =  S[t]
      =  S[87]
      =  60

CT    =  PT ⊕ K
      =  25⊕60
      =  37
```

We see that K=60 is the pseudorandom number generated by the SBOX. The value is used to mate the plain pixel. And the cipher pixel is the result of the process. The cipher pixel after encryption is 37. This calculation continues until all the pixel in the layer are covered. Table 7 to Table 9 are the encrypted numbers. The process of the encryption must include three layers.

Table 7 : The red layer after encryption

ENCRYPTED RED PIXEL									
37	36	112	41	179	169	24	39	107	49
40	75	234	110	241	209	18	176	237	164
242	31	239	210	67	58	3	145	137	102
52	106	60	30	33	151	203	116	122	227
29	187	68	223	27	169	18	0	186	97
1	190	88	33	248	146	161	136	110	183
49	28	78	160	135	214	123	58	135	49
218	76	56	64	5	113	105	244	163	1
157	40	149	89	151	198	202	105	151	109
77	129	25	192	190	12	250	188	252	90

Table 8 : The green layer after encryption

ENCRYPTED GREEN PIXEL									
247	127	23	237	144	34	52	205	210	114
110	208	98	229	78	212	28	205	80	73
199	170	89	121	250	29	44	226	104	113
54	104	170	203	138	126	108	214	89	158
184	157	55	115	92	178	218	97	174	169
88	135	161	17	57	153	230	133	50	191
236	84	196	145	95	231	52	68	79	102
9	136	152	188	31	227	6	90	216	11
80	132	171	236	242	169	192	189	224	69
123	57	236	16	235	143	129	90	182	182

Table 9 : The blue layer after encryption

ENCRYPTED BLUE PIXEL									
85	73	103	201	19	170	129	48	117	145
132	73	218	14	121	77	64	184	222	214
1	250	67	174	211	74	41	177	8	203
236	93	45	250	130	4	44	26	45	53
66	18	72	102	85	230	60	63	85	229
252	192	158	212	25	155	9	136	69	88
84	168	31	119	217	117	156	178	125	37
178	250	205	147	167	240	52	136	63	77
128	28	138	130	26	48	9	80	73	90
249	95	94	29	153	19	166	128	92	127

Now there are all set. The three layers have been encrypted using the SBOX key. Figure 6 shows the image after reconstruction. The image in hundred times magnification, if we see in normal range, it looks like a noisy dotted-picture.

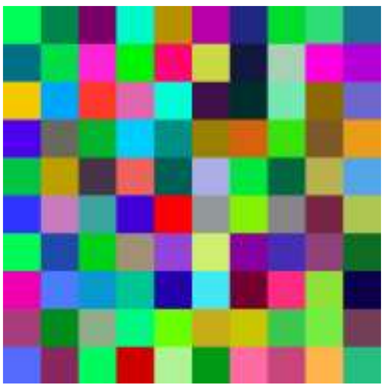

Figure 6 : The encrypted image in magnification

Figure 7 shows the encrypted image in a normal view. The image size is 1024 x 768 pixel.

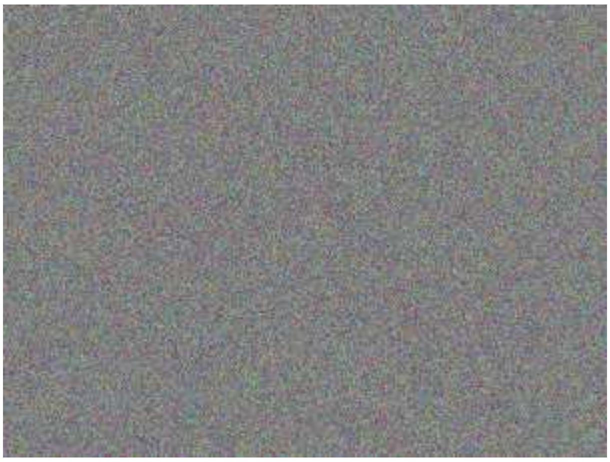

Figure 7 : The encrypted image in normal view

From the discussion, RC4 can be applied to an image. After doing the encryption, the image pattern is entirely different. In Figure 6, there is a various color from the close view while in Figure 7, the image looks like noisy.

IV. CONCLUSION

In the digital era, the image security is of great importance since the network all over the world is vulnerable. We proposed the encryption of the RGB image in visual cryptography. The crucial problem when sending an image to other is that it can be stolen. It requires a good method to hide or manipulate it from being analyzed. RC4 offers the right method to cover up the content of the picture. RC4 has three levels option to allow us to select which layer is encrypted. This approach provides high security.

V. REFERENCES

[1] K. D. Patel dan S. Belani, "Image Encryption Using Different Techinques," *International Journal of Emerging Technology and Advanced Engineering,* vol. 1, no. 1, pp. 30-34, 2011.

[2] P. V. Chavan, M. Atique dan L. Malik, "Design and Implementation of Hierarchical Visual Cryptography with Expansionless Shares," *International Journal of Network Security,* vol. 6, no. 1, pp. 91-102, 2014.

[3] A. Shamir, "How to share a secret," *Communications of ACM,* vol. 22, no. 11, pp. 612-613, 1979.

[4] A. P. U. Siahaan, "Vernam Conjugated Manipulation of Bit-Plane Complexity Segmentation," *International Conference of Computer, Environment, Social Science, Engineering & Technology*, Medan, 2016.

[5] A. P. U. Siahaan, "Blum Blum Shub in Generating Key in RC4," *KNSI,* Batam, 2016.

[6] A. P. U. Siahaan, "BPCS Steganography Noise-For Region Security Improvisation," *The International Journal of Science & Technoledge,* 2016.