Pengamanan Citra Digital Menggunakan Algoritma Pohlig Hellman

Wasit Ginting

* Corresponding author : wasitginting74@gmail.com
Fakultas Ilmu Komputer Universitas Katolik Santo Thomas Sumatera Utara
Jln. Setia Budi No.479-F, Tanjung Sari, Medan

Abstract-- Image encryption is a technique to protect the confidentiality of images from illegal access. With image encryption, people who do not know how the image is hidden will not know how to open it. Encryption is needed because in today's digital era digital images are easily stored or transmitted through public channels such as the internet. Image transmission through public channels is prone to tapping and storing images in storage media prone to access by parties without authority. For this reason, it is necessary to do a cryptographic method to secure confidential images from parties who do not have the authority to understand the actual image. This study presents an image encryption algorithm using Pohlig Hellman's Algorithm. The Pohlig Hellman algorithm is an algorithm used to compute discrete logarithms in a multiplicative group whose rank is not a safe prime number.

***Keywords**: Cryptography, Image, Pohlig-Hellman*

Abstrak-- *Enkripsi citra merupakan teknik untuk melindungi kerahasiaan citra dari pengaksesan ilegal. Dengan adanya enkripsi citra, orang-orang yang tidak mengetahui bagaimana citra tersebut disembunyikan tidak akan mengetahui bagaimana cara untuk membuka. Enkripsi diperlukan karena dalam era digital sekarang ini citra digital mudah disimpan atau ditransmisikan melalui saluran publik seperti internet. Pengiriman citra melalui saluran publik rawan terhadap penyadapan dan penyimpanan citra didalam media storage rawan terhadap pengaksesan oleh pihak-pihak yang tidak memiliki otoritas. Untuk itu maka perlu dilakukan metode kriptografi untuk mengamankan gambar yang bersifat rahasia dari pihak-pihak yang tidak memiliki otoritas untuk memahami gambar yang sebenarnya. Penelitian ini menyajikan algoritma enkripsi gambar menggunakan Algoritma Pohlig Hellman. Algoritma Pohlig Hellman merupakan sebuah algoritma yang digunakan untuk mengkomputasi logaritma diskrit dalam sebuah kelompok multiplikatif yang pangkatnya bukan merupakan bilangan prima yang aman.*

Kata Kunci : Kriptografi, Citra, Pohlig-Hellman

PENDAHULUAN

Fenomena pada era masyarakat informasi saat ini dengan mudahnya kita mendapatkan banyak informasi yang tersebar dari beragam bentuk khususnya dalam bentuk citra. Padahal informasi tersebut tanpa disadari memiliki nilai yang sangat tinggi/berharga bagi pribadi, institusi atau organisasi. Sehingga sangat rentan akan dimanfaatkan oleh pihak-pihak yang tidak bertanggung jawab bagi kepentingan pribadi atau kelompoknya. Apalagi dengan tersedianya program aplikasi yang sangat mudah dioperasikan sehingga para pelaku dapat memanipulasi citra sesuai dengan niat jahatnya.

Citra digital yang bersifat pribadi dan rahasia sangat rentang terhadap penyadapan oleh pihak-pihak lain, terutama bila citra tersebut

didistribusikan melalui internet. Tindakan penyadapan dan penyalahgunaan terhadap citra yang sifatnya rahasia tentu saja dapat merugikan pihak pemilik citra. Untuk melindungi kerahasiaan citra tersebut maka perlu enkripsi gambar untuk melindungi kerahasiaan citra dari pengaksesan ilegal. Dengan adanya enkripsi citra, orang- orang yang tidak mengetahui bagaimana citra tersebut disembunyikan tidak akan mengetahui bagaimana cara untuk membuka. Enkripsi diperlukan karena sekarang ini citra digital mudah disimpan atau ditransmisikan melalui saluran publik seperti internet. Pengiriman citra melalui saluran publik rawan terhadap penyadapan dan penyimpanan citra didalam media *storage* rawan terhadap pengaksesan oleh pihak-pihak yang tidak memiliki otoritas.

Penelitian ini menyajikan algoritma enkripsi gambar menggunakan Algoritma Vernam Cipher dan Pohlig Hellman. Algoritma Vernam Cipher merupakan algoritma kriptografi berjenis *symmetric key*. Kunci yang digunakan untuk melakukan enkripsi dan dekripsi menggunakan kunci yang sama. Dalam melakukan proses enkripsi, algoritma Vernam Cipher menggunakan cara *stream cipher* dimana *cipher* berasal dari hasil operasi XOR antara bit *plainteks* dan bit *key*. Sedangkan algoritma Pohlig Hellman merupakan sebuah algoritma yang digunakan untuk mengkomputasi logaritma diskrit dalam sebuah kelompok multiplikatif yang pangkatnya bukan merupakan bilangan prima yang aman. Algoritma ini didasari oleh teorema *Chinese Remainder* dan berjalan pada waktu polynomial yang akan digunakan untuk mengenkripsi nilai (Red Green Blue) RGB dari setiap susunan *pixel* pada gambar dan akan menghasilkan sebuah gambar acak. Misalnya lukisan digital ataupun citra seni fotografi karya photographer professional yang akan dikirim melalui internet untuk mengikuti suatu perlombaan, agar tidak diketahui oleh pesaingnya yang memungkinkan mereka untuk mencontek atau mengambil hak paten lukisan dan citra seni fotografi tersebut perlu diamankan agar tidak diketahui atau ditiru. Dengan demikian akan menyulitkan pihak-pihak yang tidak memiliki hak otoritas untuk memahami pola gambar yang sebenarnya.

Algoritma Kriptografi

Algoritma kriptografi adalah suatu algoritma yang berfungsi untuk melakukan konfusi data sehingga data yang dikirimkan tidak dapat diartikan secara langsung tanpa menggunakan algoritma deskripsinya. Selain itu algoritma kriptografi juga bias melakukan difisi sehingga bisa menghilangkan karakteristik dari data tersebut, sehingga dapat digunakan untuk mengamankan informasi. Pada implementasinya sebuah algoritma kriptografi harus memperhatikan kualitas layanan dari keseluruhan sistem dimana algoritma tersebut diimplementasikan. Algoritma kriptografi yang handal adalah algoritma kriptografi yang kekuatannya terletak pada kunci, bukan pada kerahasiaan algoritma itu sendiri.

Algoritma Pohlig Hellman

Pada algoritma Pohlig Hellman terdapat 3 (tiga) buah proses, yang pertama adalah proses pembentukan kunci, kedua proses enkripsi, dan yang ketiga adalah proses dekripsi. Proses pembentukan kunci dilakukan untuk mendapatkan pasangan kunci, yaitu kunci publik dan kunci privat. Setelah didapatkan pasangan kunci tersebut, maka selanjutnya kunci publik digunakan untuk proses enkripsi, sedangkan kunci privat digunakan untuk proses dekripsi [5keanggotaan data, u_{ij} adalah nilai keanggotaan objek ke-j pada cluster ke-i; $u_{ij} \in [0,1]$, c adalah jumlah cluster, n adalah jumlah objek pada data,m adalah tingkat kekaburan atau fuzzifier, dan $d^2(y_j, z_i)$ adalah jarak antara objek y_j dengan pusat cluster z_i.

METODOLOGI PENELITIAN

Pembentukan Kunci

Langkah-langkah untuk pembentukan pasangan kunci pada algoritma Pohlig-Hellman adalah sebagai berikut:

a. Pilih bilangan prima *p*

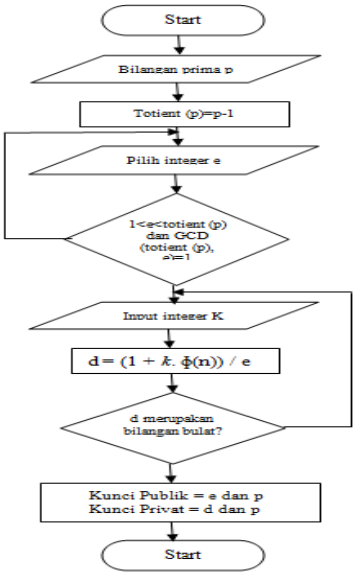
b. Hitung nilai *totient (p)* = *p* – 1

c. Pilih kunci *enkripsi* e, dengan syarat :

1<e<totient*(p)* dan GCD (totient (*p*), e) = 1

d. Hitung kunci *dekripsi* d dengan syarat, d = e-1 mod totient (p) atau dengan persamaan = (1 + k *totient *(p)*)/e, dimana k merupakan merupakan nilai 1,2,3, …, sehingga sehingga dengan demikian akan diperoleh nilai d yang bulat. Sehingga dengan demikian : Kunci Publik = e dan p dan Kunci Private= d dan p.

Proses pembentukan kunci pada algoritma Pohlig Hellman dapat dilihat selengkapnya pada Gambar 1.



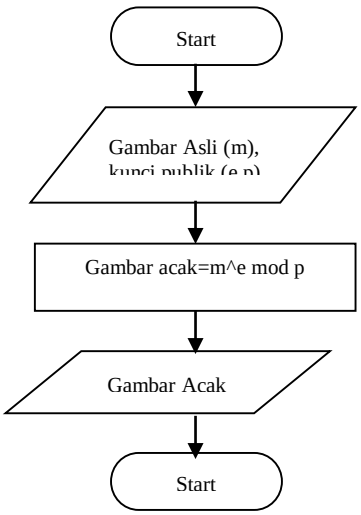
Gambar 1. *Pembentukan Kunci Pohlig Hellman*

Proses Enkripsi

Langkah-langkah untuk proses enkripsi gambar asli menjadi gambar acak menggunakan algoritma Pohlig Hellman adalah sebagai berikut:

1. Input gambar asli dan kunci publik (*e* dan *p*).
2. Setiap nilai RGB dienkripsi menggunakan kunci publik untuk menghasilkan gambar acak.

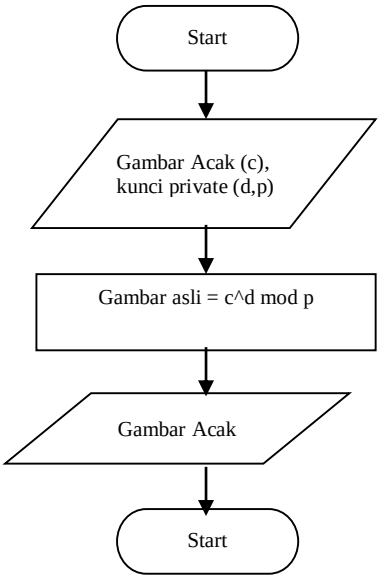
Seluruh proses enkripsi algoritma Pohlig Hellman dapat dilihat selengkapnya pada gambar berikut :



Gambar 2. *Enkripsi Algoritma Pohlig Hellman*

Langkah-langkah untuk proses dekripsi gambar acak menjadi gambar asli menggunakan algoritma Pohlig Hellman adalah sebagai berikut:

1. Input gambar acak dan kunci privat (*d* dan *p*).
2. Setiap nilai gambar acak didekripsi menggunakan kunci privat untuk menghasilkan gambar asli seluruh proses dekripsi algoritma Pohlig Hellman.



Gambar 3. *Dekripsi Algoritma Pohlig Hellman*

HASIL DAN PEMBAHASAN

Misalkan citra (gambar) yang akan di enkripsi adalah sebuah citra berukuran 4 x 3 piksel, dengan nilai RGB seperti pada tabel berikut:

Tabel 1. *Nilai RGB pada ukuran 4x3*

R = 65 G = 66 B = 67	R = 15 G = 12 B = 15	R = 47 G = 111 B = 117	R = 233 G = 217 B = 73
R = 196 G = 145 B = 10	R = 134 G = 178 B = 105	R = 32 G = 12 B = 188	R = 43 G = 78 B = 52
R = 9 G = 45 B = 140	R = 123 G = 89 B = 37	R = 123 G = 56 B = 31	R = 97 G = 98 B = 99

Maka proses enkripsinya adalah sebagai berikut:

Proses Pembentukan Kunci Pohlig Hellman

a. Input bilangan prima *p* (*p* = 347)
b. Hitug nilai *totient (p)* = p – 1 = 347 – 1 = 346
c. Pilih kunci *public* e, dengan syarat :
   1<e< *totient (p)* dan GCD (totient (p), e) = 1

Dalam hal ini nilai e yang dipilih adalah e = 337, karena GCD (346, 337) = 1

d. Hitung nilai d dengan d = e-1 mod totient (p) atau dengan persamaan d = (1 + k * *totient (p)*)/e, dimana k merupakan nilai 1,2,3, …, sehingga dengan demikian akan diperoleh nilai d yg bulat. Dengan mencoba 1,2,3,.. diperoleh nilai *d* yang bulat adalah 269, dengan menggunakan k = 262. *d* = (1 + *k*totient(p)*)/*e* = (1 + 262*346) / 337 = 269

Dengan demikian maka diperoleh kunci publik dan kunci privat sebagai berikut: Kunci Publik (p dan e) = (347, 337) dan Kunci Private (p dan d) = (347, 269)

Proses Enkripsi Pohlig Hellman

Berdasarkan nilai RGB dari citra diatas dan dengan menggunakan persamaan C = Me mod P, maka proses enkripsinya adalah sebagai berikut

Tabel 2. *Nilai RGB pada proses enkripsi Pohlig Hellman*

Nilai M	C = Me mod P
20	20337 mod 347 = 317
12	12337 mod 347 = 87
10	10337 mod 347 = 255
68	68337 mod 347= 334
77	77337 mod 347 = 153
90	90337 mod 347 = 114
97	97337 mod 347 = 57
38	38337 mod 347 = 278
62	62337 mod 347 = 234
168	168337 mod 347= 48
140	140337 mod 347 = 33
7	7337 mod 347 = 103
141	141337 mod 347 = 78
218	218337 mod 347 = 298
75	75337 mod 347 = 35
211	211337 mod 347= 223
252	252337 mod 347 = 91
32	32337 mod 347 = 218
107	107337 mod 347 = 181
77	77337 mod 347 = 153
233	233337 mod 347 = 227
101	101337 mod 347= 216
7	7337 mod 347 = 103
127	127337 mod 347 = 149
72	72337 mod 347 = 32
120	120337 mod 347 = 324
194	194337 mod 347 = 177
50	50337 mod 347= 232
18	18337 mod 347 = 230
100	100337 mod 347 = 136
46	46337 mod 347 = 100
118	118337 mod 347 = 175
86	86337 mod 347 = 186
42	42337 mod 347= 345
35	35337 mod 347 = 42
54	54337 mod 347 = 294

Kemudian nilai cipherteks dibentuk kembali menjadi sebuah citra sehingga akan menghasilkan sebuah citra acak. Karena nilai cipherteks yang dihasilkan pada proses enkripsi citra tersebut ada yang melebihi angka 255 (batas range warna pada RGB), maka dengan demikian nilai cipherteks tersebut harus dipecah (dibagi dua) supaya menghasilkan nilai RGB dengan range antara 0 – 255. Selanjutnya hasil cipherteks di konversi menjadi bilangan biner 16 bit, lalu dibagi menjadi 2 bagian (masing-masing 8 bit). Selanjutnya setiap bagian diubah menjadi bilangan desimal, seperti berikut:

C1 = 317 = 0000000100111101 → R1 = 00000001 = 1 R2 = 00111101 = 61

C2 = 87 = 00000000001010111 → G1 = 00000000 = 0 G2 = 01010111 = 87

C3 = 255 = 0000000011111111 → B1 = 00000000 = 0 B2 = 11111111 = 255

C4 = 334 = 0000000101001110 → R1 = 00000001 = 1 R2 = 01001110 = 78

C5 = 153 = 0000000010011001 → G1 = 00000000 = 0 G2 = 10011001 = 153

C6 = 114 = 0000000001110010 → B1 = 00000000 = 0 B2 = 01110010 = 114

C7 = 57 = 0000000000111001 → R1 = 00000000 = 0 R2 = 00111001 = 57

C8 = 278 = 0000000100010110 → G1 = 00000001 = 1 G2 = 00010110 = 22

C9 = 234 = 0000000011101010 → B1 = 00000000 = 0 B2 = 11101010 = 234

C10 = 48 = 0000000000110000 → R1 = 00000000 = 0 R2 = 00110000 = 48
C11 = 33 = 0000000000100001 → G1 = 00000000 = 0 G2 = 00100001 = 33
C12 = 104 = 0000000001101000 → B1 = 00000000 = 0 B2 = 01101000 = 104
C13 = 78 = 0000000001001110 → R1 = 00000000 = 0 R2 = 01001110 = 78
C14 = 298 = 00000000100101010 → G1 = 00000001 = 1 G2 = 00101010 = 42
C15 = 35 = 0000000000100011 → B1 = 00000000 = 0 B2 = 00100011 = 35
C16 = 223 = 0000000011011111 → R1 = 00000000 = 0 R2 = 11011111 = 223
C17 = 91 = 0000000001011011 → G1 = 00000000 = 0 G2 = 01011011 = 91
C18 = 218 = 0000000011011010 → B1 = 00000000 = 0 B2 = 11011010 = 218
C19 = 181 = 0000000010110101 → R1 = 00000000 = 0 R2 = 10110101 = 181
C20 = 153 = 00000000010011001 → G1 = 00000000 = 0 G2 = 10011001 = 153
C21 = 227 = 0000000011100011 → B1 = 00000000 = 0 B2 = 11100011 = 227
C22 = 216 = 0000000011011000 → R1 = 00000000 = 0 R2 = 11011000 = 216
C23 = 103 = 0000000001100111 → G1 = 00000000 = 0 G2 = 01100111 = 103
C24 = 149 = 0000000010010101 → B1 = 00000000 = 0 B2 = 10010101 = 14
C25 = 32 = 0000000000100000 → R1 = 00000000 = 0 R2 = 00100000 = 32
C26 = 324 = 00000000101000100 → G1 = 00000001 = 1 G2 = 01000100 = 68
C27 = 177 = 0000000010110001 → B1 = 00000000 = 0 B2 = 10110001 = 177
C28 = 232 = 0000000011101000 → R1 = 00000000 = 0 R2 = 11101000 = 232
C29 = 230 = 0000000011100110 → G1 = 00000000 = 0 G2 = 11100110 = 230
C30 = 136 = 0000000010001000 → B1 = 00000000 = 0 B2 = 10001000 = 136
C31 = 100 = 0000000001100100 → R1 = 00000000 = 0 R2 = 01100100 = 100
C32 = 175 = 00000000010101111 → G1 = 00000000 = 0 G2 = 10101111 = 175
C33 =186 = 0000000010111010 → B1 = 00000000 = 0 B2 = 10111010 = 186
C34 = 345 = 0000000101011001 → R1 = 00000001 = 1 R2 = 01011001 = 89
C35 = 42 = 0000000000101010 → G1 = 00000000 = 0 G2 = 00101010 = 42
C36 = 294 = 0000000100100110 → B1 = 00000001 = 1 B2 = 00100110 = 38

Susun nilai perhitungan diatas menjadi citra baru (citra acak) dengan nilai RGB pada setiap piksel seperti berikut:

Tabel 3. *Nilai RGB pada perhitungan proses enkripsi Pohlih Hellman*

R = 1 G = 0 B = 0	R = 1 G = 0 B = 0	R = 0 G = 1 B = 0	R = 0 G = 0 B = 0
R = 61 G = 87 B = 255	R = 78 G = 153 B = 114	R = 57 G = 22 B = 234	R = 48 G = 33 B = 104
R = 0 G = 1 B = 0	R = 0 G = 0 B = 0	R = 0 G = 0 B = 0	R = 0 G = 0 B = 0
R = 78 G = 42 B = 35	R = 223 G = 91 B = 218	R = 181 G = 153 B = 227	R = 216 G = 103 B = 149
R = 0 G = 1 B = 0	R = 0 G = 0 B = 0	R = 0 G = 0 B = 0	R = 1 G = 0 B = 1
R = 32 G = 68 B = 177	R = 232 G = 230 B = 136	R = 100 G = 175 B = 186	R = 89 G = 42 B = 38

Proses Dekripsi Pohlig Hellman

Sebelum dilakukan proses dekripsi, pertama tama nilai piksel pada citra hasil proses enkripsi digabung, dengan ketentuan seperti berikut:

1. Ubah nilai piksel pada citra menjadi bilangan biner, ambil nilai R biner (8 bit) pada piksel baris pertama kolom pertama, lalu digabungkan dengan nilai R biner (8 bit) pada baris kedua kolom pertama, sehingga dihasilkan nilai R biner 16 bit.
2. Ambil nilai G biner (8 bit) pada piksel baris pertama kolom pertama, lalu digabungkan dengan nilai G biner (8 bit) pada baris kedua kolom pertama, sehingga dihasilkan nilai G biner 16 bit.
3. Ambil nilai B biner (8 bit) pada piksel baris pertama kolom pertama, lalu digabungkan dengan nilai B biner (8 bit) pada baris kedua kolom pertama, sehingga dihasilkan nilai B biner 16 bit. Begitu seterusnya sehingga semua nilai biner pada piksel berhasil digabungkan, seperti berikut ini:

R1= 00000001= 1 R2 = 00111101 = 61 → 0000000100111101 = 317
G1 = 00000000 = 0 G2 = 01010111 = 87 → 00000000001010111 = 87
B1 = 00000000 = 0 B2 = 11111111= 255 → 0000000011111111 = 255
R1 = 00000001 = 1 R2 = 01001110 = 78 → 0000000101001110 = 334
G1 = 00000000 = 0 G2 = 10011001 = 153 → 0000000010011001 = 153
B1 = 00000000 = 0 B2 = 01110010 = 114 → 0000000001110010 = 114
R1 = 00000000 = 0 R2 = 00111001 = 57 → 0000000000111001 = 57
G1 = 00000001 = 1 G2 = 00010110 = 22 → 0000000100010110 = 278
B1 = 00000000 = 0 B2 = 11101010 = 234 → 0000000011101010 = 234
R1 = 00000000 = 0 R2 = 00110000 = 48 → 0000000000110000 = 48
G1 = 00000000 = 0 G2 = 00100001 = 33 → 0000000000100001 = 33
B1 = 00000000 = 0 B2 = 01101000 = 104 → 0000000001101000 = 104
R1 = 00000000 = 0 R2 = 01001110 = 78 → 0000000001001110 = 78
G1 = 00000001 = 1 G2 = 00101010 = 42 → 00000000100101010 = 298
B1 = 00000000 = 0 B2 = 00100011= 35 →0000000000100011 = 35
R1 = 00000000 = 0 R2 = 11011111 = 223 → 0000000011011111 = 223
G1 = 00000000 = 0 G2 = 01011011 = 91 → 0000000001011011 = 91
B1 = 00000000 = 0 B2 = 11011010 = 218 → 0000000011011010 = 218
R1 = 00000000 = 0 R2 = 10110101 = 181 → 0000000010110101 = 181
G1 = 00000000 = 0 G2 = 10011001 = 153 → 00000000010011001 = 153
B1 = 00000000 = 0 B2 = 11100011 = 227 → 0000000011100011 = 227
R1 = 00000000 = 0 R2 = 11011000 = 216 → 0000000011011000 = 216
G1 = 00000000 = 0 G2 = 01100111 = 103 → 0000000001100111 = 103
B1 = 00000000 = 0 B2 = 10010101 = 149 → 0000000010010101 = 149
R1 = 00000000 = 0 R2 = 00100000 = 32 → 0000000000100000 = 32
G1 = 00000001 = 1 G2 = 01000100 = 68 → 00000000101000100 = 324
B1 = 00000000 = 0 B2 = 10110001 = 177 →0000000010110001 = 177
R1 = 00000000 = 0 R2 = 11101000 = 232 → 0000000011101000 = 232
G1 = 00000000 = 0 G2 = 11100110 = 230 → 0000000011100110 = 230
B1 = 00000000 = 0 B2 = 10001000 = 136 → 0000000010001000 = 136
R1 = 00000000 = 0 R2 = 01100100 = 100 →0000000001100100 = 100
G1 = 00000000 = 0 G2 = 10101111 = 175 → 00000000010101111 = 175
B1 = 00000000 = 0 B2 = 10111010 = 186 → 0000000010111010 = 186
R1 = 00000001 = 1 R2 = 01011001 = 89 → 0000000101011001 = 345
G1 = 00000000 = 0 G2 = 00101010 = 42 → 0000000000101010 = 42
B1 = 00000001 = 1 B2 = 00100110 = 38 → 0000000100100110 = 294

Berdasarkan nilai RGB dari citra diatas dan dengan menggunakan persamaan M = Cd mod P, maka proses dekripsinya adalah sebagai berikut:

Tabel 4. *Nilai RGB pada proses dekripsi Pohlig Hellman*

RGB (C)	M = Cd mod P
317	317269 mod 347 = 20
87	87269 mod 347 = 12
255	255269 mod 347 = 10
334	334269 mod 347 = 68
153	152269 mod 347 = 77
114	114269 mod 347 = 90
57	57269 mod 347 = 97
278	278269 mod 347 = 38
234	234269 mod 347 = 62
48	48269 mod 347 = 168
33	33269 mod 347 = 140
103	103269 mod 347 = 7
78	78269 mod 347 = 141
298	298269 mod 347 = 218
35	35269 mod 347 = 75
223	223269 mod 347 = 211
91	91269 mod 347 = 252
218	218269 mod 347 = 32
181	181269 mod 347 = 107
153	153269 mod 347 = 77
227	227269 mod 347 = 233

RGB (C)	M = Cd mod P
216	216269 mod 347 = 101
103	103269 mod 347 = 7
149	149269 mod 347 = 127
32	32269 mod 347 = 72
324	324269 mod 347 = 120
177	177269 mod 347 = 194
232	232269 mod 347 = 50
230	230269 mod 347 = 18
136	136269 mod 347 = 100
100	100269 mod 347 = 46
175	175269 mod 347 = 118
186	186269 mod 347 = 86
345	345269 mod 347 = 42
42	42269 mod 347 = 35
294	294269 mod 347 = 54

Susun nilai hasil proses dekripsi diatas menjadi citra baru dengan nilai RGB pada setiap piksel seperti berikut:

Tabel 5. *Nilai RGB padan hasil proses dekripsi Pohlig Hellman*

R = 20 G = 12 B = 10	R = 68 G = 77 B = 90	R = 97 G = 38 B = 62	R = 168 G = 140 B = 7
R = 141 G = 218 B = 75	R = 211 G = 252 B = 32	R = 107 G = 77 B = 233	R = 101 G = 7 B = 127
R = 72 G = 120 B = 194	R = 50 G = 18 B = 100	R = 46 G = 118 B = 86	R = 42 G = 35 B = 54

KESIMPULAN

Berdasarkan pembahasan dan hasil dari penelitian, maka diperoleh beberapa kesimpulan sebagai berikut:

1. Dengan adanya aplikasi pengamanan citra ini, maka dapat membantu pengguna mengamankan gambar yang bersifat rahasia sehingga tidak dapat diakses (diketahui) oleh oleh pihak lain.
2. Ukuran piksel citra hasil enkripsi menjadi lebih besar 2 kali lipat dari file citra asli.
3. Jenis gambar yang diuji dalam pengamanan berekstensi JPG.

DAFTAR PUSTAKA

[1] Jumeidi Mohammad,dkk, 2016 *Implementasi Algoritma Kriptografi Vernam Cipher Berbasis FPGA* ISSN:2338-493X, Vol 4

[2] Munir Rinaldi, 2011, *Algoritma Enkripsi Citra dengan Pseudo One-Time Pad yang Menggunakan Sistem Chaos*, ISSN:2087-3328

[3] Putro Susanto Sigit, 2007, *Peranan Kriptografi dalam Keamanan Data Pada Jaringan Komputer*, ISSN:0216-0544, Vol 3

[4] Sholeh M, Hamokwarong J.V, 2001 *Aplikasi Kriptografi dengan Metode Vernam Cipher dan Metode Permutasi Biner*Vol. 7

[5] Simarmata Allwin, 2013, *Rancangan Model Algoritma Pohlig-Hellman dengan Menggunakan Multiple-key Berdasarkan Algoritma RSA Multiple-key*, ISSN:1907-5022

[6] Suarga, 2012, *Algoritma dan Pemrograman*, Yogyakarta penerbit Andi