Bulletin of Information System Research (BIOS)
Volume 1, No 3, August 2023 Page: 92-98
ISSN 2963-2455 (media online)
https://journal.grahamitra.id/index.php/bios

Penyisipan Pesan Pada File Dokumen PDF Dengan Metode Pixel Value Differencing

Putra Abdi Surya

Fakultas Ilmu Komputer dan Teknologi Informasi, Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia
Email: Oppoputra34@gmail.com

Abstrak−Data merupakan kumpulan dari fakta yang dapat berupa angka, simbol ataupun tulisan yang diperoleh melalui suatu objek. Data dapat bersifat rahasia dan dapat juga bersifat umum yang artinya dapat dipublikasikan. Data yang bersifat rahasia merupakan data yang tidak dibuat untuk dipublikasikan, namun untuk kalangan pribadi, sebab data-data tersebut merupakan data-data yang berkaitan dengan perusahaan atau data pribadinya. Namun, kadang kala data yang bersifat rahasia sering disalahkan gunakan oleh orang lain atau pihak ketiga dengan cara mengambil atau mencuri data tersebut untuk kepentingan pribadinya dan menjatuhkan orang lain atau perusahaan. Mengingat teknologi yang berkembang pesat saat ini memungkinkan setiap orang dapat menciptakan sebuah teknologi baru untuk memanipulasi ataupun mencuri data yang sifatnya rahasia. Untuk mengatasi masalah tersebut, perlua adanya dibuat pengamana data, sehingga data rahasia yang dimiliki oleh seseorang atau perusahaan tidak dapat diambil atau disalah gunakan. Teknik pengamanan data yang dilakukan untuk mengatasi masalah tersebut adalah teknik steganografi. PVD merupakan salah satu metode penyisipan yang cara kerjanya adalah mencari dua selisih nilai terdekat. Selisih tersebut kemudian digunakan untuk menentukan jumlah data yang dapat disisipkan berdasarkan jangkauan data yang dipilih. Dengan memanfaatkan metode ini, maka dapat dirancang sebuah aplikasi yang dapat digunakan untuk mengamankan data khususnya dalam penyisipan pesan ke dalam file dokumen pdf.

Kata Kunci: Penyisipan File; *Pixel Value Differencing*; Stenografi; Keamanan

Abstract−Data is a collection of facts that can be in the form of numbers, symbols or writing obtained through an object. Data can be confidential and can also be public which means that it can be published. Confidential data is data that is not made to be published, but for private circles, because the data is data related to the company or its personal data. However, sometimes confidential data is often blamed for use by other people or third parties by taking or stealing the data for their personal interests and bringing down other people or companies. Given the rapidly developing technology today, it is possible for everyone to create a new technology to manipulate or steal confidential data. To overcome this problem, it is necessary to create data security, so that confidential data owned by a person or company cannot be taken or misused. The data security technique used to overcome this problem is the steganography technique. PVD is an insertion method that works by finding the two closest value differences. The difference is then used to determine the amount of data that can be inserted based on the selected data range. By utilizing this method, an application can be designed that can be used to secure data, especially in the insertion of messages into pdf document files.

Keywords: File Insertion; Pixel Value Differencing; Shorthand; Security

1. PENDAHULUAN

Data merupakan kumpulan dari fakta yang dapat berupa angka, simbol ataupun tulisan yang diperoleh melalui suatu objek. Data dapat bersifat rahasia dan dapat juga bersifat umum yang artinya dapat dipublikasikan. Data yang bersifat rahasia merupakan data yang tidak dibuat untuk dipublikasikan, namun untuk kalangan pribadi, sebab data-data tersebut merupakan data-data yang berkaitan dengan perusahaan atau data pribadinya. Namun, kadang kala data yang bersifat rahasia sering disalahkan gunakan oleh orang lain atau pihak ketiga dengan cara mengambil atau mencuri data tersebut untuk kepentingan pribadinya dan menjatuhkan orang lain atau perusahaan. Mengingat teknologi yang berkembang pesat saat ini memungkinkan setiap orang dapat menciptakan sebuah teknologi baru untuk memanipulasi ataupun mencuri data yang sifatnya rahasia. Untuk mengatasi masalah tersebut, perlua adanya dibuat pengamana data, sehingga data rahasia yang dimiliki oleh seseorang atau perusahaan tidak dapat diambil atau disalah gunakan. Teknik pengamanan data yang dilakukan untuk mengatasi masalah tersebut adalah teknik steganografi.

Steganografi merupakan salah satu teknik pengamanan data dengan menyisipkan pesan tersebut ke dalam sebuah media penyisipan yang dijadikan sebagai wadah untuk menyisipkan pesan tersebut[1]. Steganografi memiliki beberapa algoritma yang dapat digunakan untuk menyisipkan pesan yang salah satunya adalah *pixel value differencing* (PVD).

PVD merupakan salah satu metode penyisipan yang cara kerjanya adalah mencari dua selisih nilai terdekat[1][2]. Selisih tersebut kemudian digunakan untuk menentukan jumlah data yang dapat disisipkan berdasarkan jangkauan data yang dipilih. Metode ini menggunakan dua skema yang tujuannya untuk mengetahui *range* atau jarak kedua nilai. Skema tersebut adalah skema *Wu* dan *Tsai*. Berdasarkan penelitian sebelumnya, bahwa metode *pixel value differencing* sangat efektif dalam menyisipkan pesan sehingga tidak tidak diketahui oleh orang lain[1].

Penelitian ini menguraikan bagaimana prosedur yang dilakukan untuk menyisipkan pesan pada file dokumen dengan metode PVD. Dengan memanfaatkan metode ini, maka dapat dirancang sebuah aplikasi yang dapat digunakan untuk mengamankan data khususnya dalam penyisipan pesan ke dalam file dokumen pdf.

Bulletin of Information System Research (BIOS)
Volume 1, No 3, August 2023 Page: 92-98
ISSN 2963-2455 (media online)
https://journal.grahamitra.id/index.php/bios

2. METODOLOGI PENELITIAN

2.1 Kriptografi

Kriptografi Kriptografi (cryptography) berasal dari bahasa yunani cryptós (*secret*) dan gráphein (*writing*). Jadi, kriptografi berarti secret writing (tulisan rahasia). Kriptografi merupakan ilmu dan seni untuk menjaga kerahasian pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti lagi maknanya. Kriptografi adalah sebuah cara yang efektif dalam mengamankan informasi-informasi penting baik yang tersimpan dalam media penyimpanan maupun yang ditransmisikan melalui jaringan komunikasi[3][4]. Orang yang melakukan penyandian ini disebut kriptografer, sedangkan orang yang mendalami ilmu dan seni dalam membuka atau memecahkan suatu algoritma kriptografi tanpa harus mengetahui kuncinya disebut kriptanalis.

Kriptografi pada dasarnya terdiri dari dua proses, yaitu proses enkripsi dan proses dekripsi. Proses enkripsi adalah proses penyandian pesan terbuka menjadi pesan rahasia (ciphertext). Ciphertext inilah yang nantinya akan dikirimkan melalui saluran komunikasi terbuka. Pada saat ciphertext diterima oleh penerima pesan, maka pesan rahasia tersebut diubah lagi menjadi pesan terbuka melalui proses dekripsi sehingga pesan tadi dapat dibaca kembali oleh penerima pesan.

2.2 Metode Pixel Value Differencing

Pixel Value Differencing (PVD) merupakan salah satu algoritma dalam steganografi. Metode ini dapat menyisipkan pesan lebih banyak pada pixel yang memiliki nilai kekontrasan tinggi[1][2][5]. Untuk menambah tingkat keamanan dari informasi yang akan disisipkan, digunakan kriptografi. *Pixel Value Differencing* bekerja pada sepasang nilai pixel yang bertenggaan. Proses penyisipan informasi dilakukan dengan memodifikasi selisih nilai pixel sesuai dengan nilai bit pesan dan tabel kuantisasi selisih nilai keabuan. kuantisasi selisih nilai keabuan digunakan untuk menentukan jumlah bit yang akan disisipkan pada selisih nilai tertentu. Cara kerja metode Pixel Value Differencing (PVD)[1][6][7], dijabarkan sebagai berikut :

1. Penyisipan pesan berdasarkan metode *Pixel Value Differencing* (PVD)
   a. Mengubah Pesan Menjadi biner 8 bit
   b. Menghitung 2 piksel bertetangga dengan rumus ($g_{i,}\ g_{i+1}$)
   c. Tentukan batas bawah (Ik) dan jumlah bit n, dengan cara :
      $di = gi+1 - gi$ (1)
   d. Ambil pesan sebanyak n bit, kemudian ubah menjadi desimal (b)
   e. Menghitung selisi nilai yang baru
   f. Mencari nilai m dengan rumus m = d' – d
   g. Menghitung piksel baru
2. Ekstraksi pesan berdasarkan metode *Pixel Value Differencing* (PVD)
   a. Menghitung 2 piksel bertetangga
   b. Menentukan batas bawah (Ik) dan jumlah bit n, dengan cara :
      $Ik<=di>=I_{k+1}$ (2)
   c. Menghitung nilai b dengan rumus $b = |d| - I_k$
   d. Ubah b (desimal) menjadi biner n bit dan terakhir adalah ambil pesan = n bit

3. HASIL DAN PEMBAHASAN

Teknik penyisipan gambar atau sering disebut steganografi merupakan salah satu teknik pengamanan data yang sering dimanfaatkan oleh orang banyak dalam mengamankan data. Data-data yang diamankan berupa data yang sifatnya pribadi atau rahasia. Data yang hendak dikirimkan atau disampaikan kepada penerima memerlukan pengamanan agar pesan dapat sampai ke penerima tanpa diketahui oleh orang banyak. Sebab kadang kala pesan rahasia dapat diambil dan dicuri oleh pihak-pihak yang bertanggung jawab untuk kepentingan pribadi dan dengan tujuan lain seperti menjatuhkan nama baik seseorang atau perusahaan. Untuk itu perlu adanya dibuat pengamanan agar pesan yang hendak disampaikan tersebut tidak diketahui oleh orang banyak.

Pemanfaatan teknik penyisipan merupakan solusi untuk menyelesaikan masalah tersebut, agar pesan dapat sampai ke penerima tanpa diketahui oleh orang lain. Nah pesan yang akan sampai kepada penerima disisipkan ke dalam sebuah wadah berupa file dokumen pdf sehingga ketika pesan tersebut sampai kepada penerima, maka pesan yang sampai berupa pdf. Hal ini dapat menghindari kecurigaan orang lain, sebab pesan tersebut adalah file dokumen pdf[8].

Penyisipan pesan yang akan digunakan dalam penelitian ini adalah dengan memanfaatkan algoritma *pixel value differencing*. Pesan akan disisipkan ke dalam sebuah file dokumen pdf menggunakan algoritma *pixel value differencing*. Berikut adalah proses penyisipan pesan ke dalam file dokumen pdf :

1. Pesan yang akan disisipkan terlebih dikonversikan ke dalam bentuk bilangan biner
2. Menyiapkan file dokumen pdf yang akan dijadikan sebagai wadah tempat untuk menyisipkan pesan dan kemudian mencari nilai desimal dari file dokumen pdf tersebut
3. Melakukan proses penyisipan dan menghasilkan file dokumen yang baru yang telah disisipkan pesan

4. Kemudian untuk mendapatkan pesan tersebut, dilakukan proses ekstraksi pesan
5. Setelah dilakukan proses ekstraksi pesan, maka akan didapatkan pesan awal yang telah disisipkan tersebut.

3.1 Penerapan Metode Pixel Value Differencing

Tahapan pertama yang dilakukan dalam penerapan metode PVD untuk menyisipkan pesan ke dalam file dokumen pdf yaitu pesan yang akan diamankan terlebih dahulu diknversikan ke dalam Desimal ASCII yang selanjutnya nilai yang telah di dapat dari hasil konversi tersebut di ubah ke dalam bentuk bilangan biner. Kemudian mempersiapkan file dokumen pdf yang akan digunakan sebagai wadah untuk menampung pesan tersebut dan memilih bagian yang akan dijadikan sebagai tempat untuk menampung pesan tersebut. untuk lebih jelasnya berikut adalah tahapan penerapan metode *pixel value differencing* dalam pengamanan pesan :

1. Pesan yang akan disisipkan terlebih dikonversikan ke dalam bentuk bilangan biner
2. Menyiapkan file dokumen pdf yang akan dijadikan sebagai wadah tempat untuk menyisipkan pesan dan mencari nilai desimal dari file dokumen pdf tersebut
3. Melakukan proses penyisipan dengan menggunakan algoritma pixel value differencing dan menghasilkan file dokumen pdf yang baru yang telah disisipkan pesan
4. Kemudian untuk mendapatkan pesan tersebut, dilakukan proses ekstraksi pesan dengan menggunakan algoritma pixel value differencing
5. Setelah dilakukan proses ekstraksi pesan, maka akan didapatkan pesan awal yang telah disisipkan tersebut.

Adapun contoh penerapan algoritma *pixel value differencing* dalam menyisipkan pesan ke dalam file dokumen seperti di bawah ini :

Dimisalkan pesan yang akan disisipkan adalah KUNCI

1. Konversikan pesan di atas ke dalam desimal ASCII dan kemudian diubah nilainya ke dalam bentuk bilangan biner.
2. Sehingga biner pesan menjadi 01001011 01010101 01001110 01000011 01001001
3. Menyiapkan file dokumen pdf yang akan dijadikan sebagai wadah untuk menampung pesan di atas.
4. Kemudian mencari nilai desimal dari file dokumen di atas.
5. Proses penentuan jumlah bit setiap iterasi
   Tahapan ini merupakan tahapan untuk mendapatkan jumlah bit setiap iterasi yang akan disisipkan ke dalam setiap nilai desimal file dokumen pdf. Proses ini dilakukan sampai keseluruhan biner disisipkan
   a. Iterasi 1
      $d = 80 - 37 = 43$
      32<=d<63; *Ik* = 32 jumlah bit yang diambil n = 5 (berdasarkan rentang nilai *wu* dan *tsai*)
   b. Iterasi 2
      d = 70 – 68 = 2
      0<=d>=7; *Ik* = 8 jumlah bit yang diambil n = 3 (berdasarkan rentang nilai *wu* dan *tsai*)
   c. Iterasi 3
      $d = 49 - 45 = 3$
      0<=d<=7; *Ik* = 8 jumlah bit yang diambil n = 3 (berdasarkan rentang nilai *wu* dan *tsai*)
   d. Iterasi 4
      $d = 55 - 46 = 9$
      8<=d>=150; *Ik* = 8 jumlah bit yang diambil n = 3 (berdasarkan rentang nilai *wu* dan *tsai*)
   e. Iterasi 5
      $d = 10 - 13 = -3$
      D<0; *Ik* = 8 jumlah bit yang diambil n = 3 (berdasarkan rentang nilai *wu* dan *tsai*)
   f. Iterasi 6
      $d = 40 - 30 = 10$
      8<=d>=15; *Ik* = 8 jumlah bit yang diambil n = 3 (berdasarkan rentang nilai *wu* dan *tsai*)
   g. Iterasi 7
      $d = 49 - 10 = 39$
      32<=d<=63; *Ik* = 32 jumlah bit yang diambil n = 5 (berdasarkan rentang nilai *wu* dan *tsai*)
   h. Iterasi 8
      $d = 48 - 32 = 16$
      16<=d<=31; *Ik* = 16 jumlah bit yang diambil n = 4 (berdasarkan rentang nilai *wu* dan *tsai*)
   i. Iterasi 9
      $d = 111 - 31 = 80$
      64<=d<=127; *Ik* = 64 jumlah bit yang diambil n = 6 (berdasarkan rentang nilai *wu* dan *tsai*)
   j. Iterasi 10
      $d = 106 - 98 = 8$
      8<=d<=15; *Ik* = 8 jumlah bit yang diambil n = 3 (berdasarkan rentang nilai *wu* dan *tsai*)
   k. Iterasi 11
      $d = 116 - 97 = 19$

Bulletin of Information System Research (BIOS)
Volume 1, No 3, August 2023 Page: 92-98
ISSN 2963-2455 (media online)
https://journal.grahamitra.id/index.php/bios

16<=d<=31; *Ik* = 16 jumlah bit yang diambil n = 4 (berdasarkan rentang nilai *wu* dan *tsai*)

Maka jumlah bit yang akan disisipkan adalah 42 bit dan membutuhkan 22 nilai desimal dari file dokumen pdf

6. Tahapan penyisipan

Pesan = 01001011 01010101 01001110 01000011 01001001

a. Penyisipan iterasi 1

Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 1 adalah 5, maka ambil pesan sebanyak 5 bit dimulai dari kiri

b = 01001 = 11 (desimal)

d>=0, maka selisih baru : d' = (lk + b) = (31 + 11) = 42

m = d' – d = 42 – 43 = -1

karena m ganjil, maka

gi', gi'_{i+1} = (37 – (-1/2), 80 + (-1/2) = (37,80)

b. Penyisipan iterasi 2

Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 2 adalah 3, maka ambil pesan sebanyak 3 bit dimulai dari kiri

b = 011 = 3 (desimal)

d>=0, maka selisih baru : d' = (lk + b) = (8 + 3) = 11

m = d' – d = 11 – (3) = 8

karna m ganjil, maka

gi', gi'_{i+1} = (68 – (8/2), 70 + (8/2) = (63,74)

c. Penyisipan iterasi 3

Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 3 adalah 3, maka ambil pesan sebanyak 3 bit dimulai dari kiri

b = 010= 2 (desimal)

d>=0, maka selisih baru : d' = (lk + b) = (8 + 2) = 10

m = d' – d = 10 – 3 = 7

karna m ganjil, maka

gi', gi'_{i+1} = (45 – (7/2), 49 + (7/2) = (43,53)

d. Penyisipan iterasi 4

Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 4 adalah 3, maka ambil pesan sebanyak 3 bit dimulai dari kiri

b = 101= 5 (desimal)

d<=0, maka selisih baru : d' = (lk + b) = (8 + 5) = 13

m = d' – d = 13 – 9 = 4

karna m ganjil, maka

gi', gi'_{i+1} = (46 – (4/2), 55 + (4/2) = (43,57)

e. Penyisipan iterasi 5

Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 5 adalah 3, maka ambil pesan sebanyak 3 bit dimulai dari kiri

b = 010 = 2 (desimal)

d>=0, maka selisih baru : d' = (lk + b) = (8 + 2) = 10

m = d' – d = 10 – (-3)) = 13

karna m genap, maka

gi', gi'_{i+1} = (13 – (13/2), 10 + (13/2) = (7,17)

f. Penyisipan iterasi 6

Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 6 adalah 3, maka ambil pesan sebanyak 3 bit dimulai dari kiri

b = 100 = 4 (desimal)

d>=0, maka selisih baru : d' = (lk + b) = (8+ 4) = 12

m = d' – d = 12– (10)) = 2

karna m genap, maka

gi', gi'_{i+1} = (30 – (2/2), 40 + (2/2) = (29,41)

g. Penyisipan iterasi 7

Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 7 adalah 5, maka ambil pesan sebanyak 5 bit dimulai dari kiri

b = 11100 = 28 (desimal)

d>=0, maka selisih baru : d' = (lk + b) = (32 + 28) = 60

m = d' – d = 60 – 39) = 21

karna m ganjil, maka

gi', gi'_{i+1} = (10 – (21/2), 49 + (21/2) = (0,60)

h. Penyisipan iterasi 8

Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 8 adalah 4, maka ambil pesan sebanyak 4 bit dimulai dari kiri
b = 1000 = 8 (desimal)
d>=0, maka selisih baru : d' = (lk + b) = (16 + 8) = 24
m = d' – d = 24 – 16) = 8
karna m genap, maka
gi',gi'_{i+1} = (32 – (8/2), 48 + (8/2) = (28,52)

i. Penyisipan iterasi 9
Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 9 adalah 6, maka ambil pesan sebanyak 6 bit dimulai dari kiri
b = 011010 = 26 (desimal)
d>=0, maka selisih baru : d' = (1k + b) = (64 + 26) = 90
m = d' – d = 90– 80) = 10
karna m genap, maka
gi',gi'_{i+1} = (31 – (10/2), 11 + (10/2) = (26,16)

j. Penyisipan iterasi 10
Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 10 adalah 3, maka ambil pesan sebanyak 3 bit dimulai dari kiri.
b = 010 = 2 (desimal)
d>=0, maka selisih baru : d' = (1k + b) = (8+ 2) = 10
m = d' – d = 10 – 8 = 2
karna m genap, maka
gi',gi'_{i+1} = (98 – (2/2), 106 + (2/2) = (97,108)

k. Penyisipan iterasi 11
Jumlah bit yang disisipkan berdasarkan hasil dari perhitungan iterasi 11 adalah 3, maka ambil pesan sebanyak 3 bit dimulai dari kiri.
b = 01 = 1 (desimal)
d>=0, maka selisih baru : d' = (1k + b) = (16 + 1) = 17
m = d' – d = 17 –19 = -2
karna m genap, maka
gi',gi'_{i+1} = (97 – (-2/2), 116 + (-2/2) = (98,114)

7. Tahapan Ektraksi Pesan
Tahapan ekstraksi merupaka tahapan pengembalian pesan yang telah disisipkan. Berikut adalah proses ektraksi pesan

a. Iterasi 1
d = 80 – 37 = 43
32<=d<=63 jumlah bit yang disisipkan adalah 5 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |43| - 32 = 11
b = 5 = 01001
pesan 5 bit = 01001

b. Iterasi 2
d = 74 – 63 = 11
0<=d<=8 jumlah bit yang disisipkan adalah 3 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |11| - 8 = 3
b = 3 = 011
pesan 3 bit = 011

c. Iterasi 3
d = 53– 43 = 10
8<=d<=15 jumlah bit yang disisipkan adalah 3 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |10| - 8 = 2
b = 3 = 010
pesan 3 bit = 010

d. Iterasi 4
d = 57 – 43 = 13
8<=d<=15 jumlah bit yang disisipkan adalah 3 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |13| - 8 = 5
b = 5 = 101
pesan 2 bit = 101

e. Iterasi 5
d = 17 – 7 = 10
8<=d<=15 jumlah bit yang disisipkan adalah 3 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |10| - 8 = 2

b = 2 = 010
pesan 3 bit = 010

f. Iterasi 6
d = 41 – 29 = 12
8<=d<=15 jumlah bit yang disisipkan adalah 3 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |12| - 8 = 4
b = 4 = 100
pesan 3 bit = 100

g. Iterasi 7
d = 60 – 0 = 60
32<=d<=63 jumlah bit yang disisipkan adalah 5 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |60| - 32 = 28
b = 28 = 11100
pesan 5 bit = 11100

h. Iterasi 8
d = 52 – 28 = 24
16<=d<=31jumlah bit yang disisipkan adalah 4 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |24| - 16 = 8
b = 8 = 1000
pesan 4 bit = 1000

i. Iterasi 9
d = 16 – 26 = -10
8<=d<=15 jumlah bit yang disisipkan adalah 6 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |-10| - 8 = 18
b = 26 = 011010
pesan 6 bit = 011010

j. Iterasi 10
d = 107 – 97 = 11
8<=d<=15 jumlah bit yang disisipkan adalah 3 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |10| - 8 = 2
b = 2 = 010
pesan 3 bit = 010

k. Iterasi 11
d = 114 – 98 = 9
8<=d<=15 jumlah bit yang disisipkan adalah 3 (berdasarkan rentang nilai *wu* dan *tsai*)
b = |9| - 8 = 1
b = 1 = 01
pesan 3 bit = 01

Maka pesan yang sudah berhasil diekstrak adalah 01001011 01010101 01001110 01000011 01001001

4. KESIMPULAN

Berdasarkan hasil dari analisa yang telah dilakukan, maka penulis mengambil kesimpulan dari penelitian ini sebagai berikut Proses penyisipan pesan ke dalam dokumen pdf berdasarkan algoritma *pixel value differncing* dimulai dari tahapan konversi pesan yang akan disisipkan ke dalam file dokumen pdf ke dalam bentuk desimal ASCII dan kemudian diubah menjadi bilangan biner. Setelah itu memilih file dokumen PDF yang akan dijadikan sebagai tempat penyisipan pesan dan mencari nilai ASCII dari file dokumen pdf tersebut, lalu melakukan proses penyisipan.

REFERENCES

[1] A. Lestari, A. S. Sembiring, and T. Zebua, “Teknik Penyembunyian Pesan Teks Terenkripsi Algoritma Merkle-Hellman Knapsack Menggunakan Metode Pixel Value Differencing Ke Dalam Citra Digital,” *KOMIK (Konferensi Nas. Teknol. Inf. dan Komputer)*, vol. 3, no. 1, pp. 204–212, 2019, doi: 10.30865/komik.v3i1.1590.
[2] R. Andri, R. K. Hondro, and K. Tampubolon, “Implementasi Metode Pixel Value Differencing Untuk Penyembunyian Pesan Pada Citra Digital,” *KOMIK (Konferensi Nas. Teknol. Inf. dan Komputer)*, vol. 3, no. 1, pp. 355–361, 2019, doi: 10.30865/komik.v3i1.1613.
[3] R. Munir, *Kriptografi*, 1st ed. Bandung: Informatika Bandung, 2006.
[4] A. H. K. Fresly Nandar Pabokory, Indah Fitri Astuti, “Implementasi Kriptografi pengamanan Data pada Pesan Text Isi File Dokumen Dan FIle Dokumen Menggunakan Algoritma Advanced Encription Standard,” *J. Inform. Mulawarman*, vol. 10, no. 1, pp. 1–10, 2015
[5] R. Sadikin, *Kriptografi Untuk Keamanan Jaringan*. 2012.l
[6] H. E. Dwiyanto, “Format File,” *Teknol. J.*, vol. 1, no. 2, pp. 1–5, 2020.
[7] R, Rahim, " Penyisipan Pesan Dengan Algoritma Pixel Value Differencing Dengan Algoritma Caesar Cipher Pada Proses

Steganografi," *J. TIMES*, vol. V, no. 1, pp. 6–11, 2016.
[8] Z. Nabil, A. Tirta, and A. Prihanto, "Penerapan Steganografi Dengan Menggunakan Metode Least Significant Bit (Lsb) Dan Pixel Value Differencing (Pvd) Pada Citra Warna," vol. 01, pp. 165–173, 2020.