Implementasi Key-Based Steganography Sebagai Bentuk Watermarking pada Citra Digital

Muhammad Rafi Haidar (18221134)
Program Studi Sistem dan Teknologi Informasi
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
muhammadrafihaidar03@gmail.com

Abstract*—Seiring berkembangnya zaman, keinginan manusia untuk menikmati media sebagai bentuk rekreasi juga turut meningkat. Kemudahan akses dan distribusi karya digital, di balik keuntungannya, memunculkan permasalahan terkait hak cipta pencipta karya. Solusi yang diusulkan adalah perlindungan hak cipta karya digital melalui watermarking digital dengan skema key-based steganography. Solusi ini menggabungkan teknik steganografi domain spasial dengan algoritma kriptografi asimetris RSA. Teknik steganografi domain spasial, khususnya metode LSB, digunakan untuk menyematkan watermark ke dalam gambar karya tanpa mempengaruhi kualitas gambar. Algoritma RSA digunakan untuk memastikan hanya pencipta karya asli yang dapat menyematkan watermark dan pengguna lain dapat memverifikasinya dengan kunci public pencipta. Makalah ini membahas rancangan dan implementasi solusi. Implementasi yang berhasil dicapai adalah fungsi untuk menyematkan watermark. Fungsi untuk mengekstrak watermark masih dalam tahap pengembangan. Solusi ini memiliki beberapa keuntungan, yaitu memverifikasi keaslian karya, keefektifan, dan kompleksitas komputasi yang rendah. Pengembangan lebih lanjut diperlukan untuk menyelesaikan implementasi fungsi ekstraksi watermark, meningkatkan keamanan, dan melakukan pengujian menyeluruh. Solusi watermarking digital dengan skema key-based steganography memiliki potensi untuk melindungi hak cipta karya digital. Implementasi yang lebih lengkap dan pengujian yang lebih menyeluruh diperlukan untuk membuktikan efektivitas solusi ini.

Keywords—steganography; key-based steganography; watermarking; digital image; spatial domain cryptography; asymmetric cryptography;

I. PENDAHULUAN

Seiring perkembangan zaman, keinginan manusia untuk menikmati media sebagai bentuk rekreasi juga turut meningkat. Kemajuan teknologi informasi dan komunikasi telah mengubah cara manusia menikmati media secara drastis. Kehadiran internet dan berbagai perangkat konsumen yang terjangkau seakan-akan membuka gerbang menuju dunia media yang tidak terbatas. Orang-orang dapat melihat, mengunduh, mengubah, dan membagikan media dengan mudah dan instan.

Namun, di balik segala kemudahan ini, identitas pencipta karya digital seringkali hilang, baik secara tidak sengaja maupun tidak disengaja. Hal ini menimbulkan berbagai masalah bagi para pencipta media digital, terutama mereka yang datang dari latar belakang seni. Para seniman, yang telah mencurahkan segenap waktu, tenaga, dan kreativitas dalam proses penciptaan karya asli, berhak menerima pengakuan atas usaha dan hasil mereka.

Tentunya masalah mengenai hak cipta suatu karya bukanlah hal yang baru dan berbagai upaya yang telah dikembangkan untuk melindungi hak cipta seorang seniman. Entitas besar seperti perusahaan mungkin akan mendaftarkan karya atau produknya secara resmi untuk mendapatkan perlindungan hukum atas kekayaan intelektualnya. Sayangnya, proses tersebut seringkali tidak praktis untuk dilakukan oleh seniman independen di internet. Sifat internet yang relatif *laissez-faire* dan kesulitan menerapkan hukum negara pada platform global menjadikan upaya perlindungan tradisional seperti pendaftaran hak cipta menjadi sangat kompleks.

Upaya perlindungan hak cipta tradisional dianggap kurang relevan dengan perkembangan internet dam para pencipta karya terdorong untuk mencari cara lain untuk melindungi karya mereka. Karena mencegah penyebaran karya di internet secara sepenuhnya merupakan hal yang mustahil, dan kebanyakan pencipta memang memaksudkan karyanya untuk dilihat oleh umum, teknik yang umumnya digunakan adalah penambahan *watermark* sederhana. *Watermark* ini berfungsi sebagai penanda bahwa karya tersebut adalah karya asli dari seniman terkait.

Sayangnya, teknik tersebut memiliki keterbatasan dalam memberikan perlindungan. Orang dapat dengan mengubah karya asli dan menghapus *watermark* milik pencipta. Hal tersebut sangat mudah dilakukan, terutama pada karya yang berupa citra. Pencipta tentunya dapat menerapkan teknik *watermarking* yang lebih agresif, seperti pola yang berulang. Namun hal tersebut tentunya akan merusak gambar dan mengganggu pengalaman orang lain dalam menikmati karya tersebut.

Teknik lain yang dapat digunakan oleh pencipta karya adalah dengan menggunakan skema *watermarking* digital melalui *key-based steganography*. Teknik ini menyisipkan *watermark* secara langsung ke berkas karyanya tanpa mempengaruhi kualitas karya secara signifikan. *Watermark* dapat berupa kode tersembunyi, pola unik, atau infomasi lain yang dapat diekstrak oleh perangkat lunak khusus dan kombinasi kunci yang tepat untuk membuktikan keaslian karya.

II. TINJAUAN PUSTAKA

Berikut merupakan beberapa hasil tinjauan pustaka yang digunakan dalam penulisan makalah ini.

A. Steganography

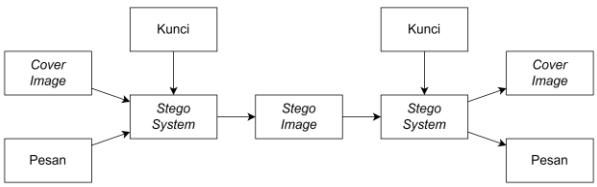
Steganografi atau stego, dalam konteks TI, adalah teknik untuk menyembunyikan informasi ke dalam media digital sedemikian rupa sehingga informasi tersebut tidak diketahui keberadaanya kecuali oleh pihak yang dituju [1]. Kata steganografi berasal dari bahasa Yunani *steganos*, yang artinya "tersembunyi", dan *graphein*, yang artinya "menulis".

Steganografi telah digunakan sepanjang sejarah yang tercatat. Herodotus, seorang sejarawan Yunani, mencatat penggunaan budak sebagai media steganografi dengan cara menuliskan pesan rahasia ke kepala budak yang telah dicukur dan membiarkan rambut budak tersebut tumbuh kembali untuk menutupi pesan rahasia tersebut.

Steganografi telah digunakan sepanjang sejarah yang tercatat. Herodotus, seorang sejarawan Yunani, mencatat penggunaan budak sebagai media steganografi dengan cara menuliskan pesan rahasia ke kepala budak yang telah dicukur dan membiarkan rambut budak tersebut tumbuh kembali untuk menutupi pesan rahasia tersebut.

Berbeda dengan kriptografi yang memungkinkan orang lain mendeteksi keberadaan informasi karena bentuknya sebagai informasi yang terenkripsi, steganografi bertujuan untuk menyembunyikan informasi ke dalam media lain sedemikian rupa sehingga media tersebut seakan-akan tidak mengandung informasi yang disembunyikan.

Umumnya steganografi dilakukan dengan cara memasukkan *cover image*, kunci, dan pesan ke dalam sebuah *stego system* untuk mendapatkan sebuah *stego image*. Pesan nantinya akan diekstrak dengan memasukkan kembali *stego image* ke dalam *stego system* dengan kunci yang sesuai untuk mendapatkan pesan dan *cover image*. Proses tersebut dapat dilihat pada Gambar 1.



Gambar 1. Alur Umum Proses Steganografi

Teknik-teknik steganografi dapat diklasifikasikan berdasarkan jenis media penutup yang disematkan informasi dan modifikasi terhadap media penutup yang dilakukan pada proses penyematan informasi [2].

Berikut merupakan klasifikasi teknik steganografi berdasarkan media penutupnya:

1. Teknik Steganografi Gambar

   Informasi rahasia disematkan ke dalam citra atau gambar digital. Teknik ini adalah teknik yang paling umum digunakan dan akan menjadi fokus pada makalah ini. Gambar digital dapat berupa berkas dengan format JPG, JPEG, atau PNG.

2. Teknik Steganografi Audio

   Informasi rahasia disematkan ke dalam berkas audio yang dapat berupa berkas MP3 atau WAV.

3. Teknik Steganografi Video

   Informasi rahasia disematkan ke dalam berkas video, seperti AVI, MPEG, atau MP4.

4. Teknik Steganografi Teks

   Informasi rahasia disematkan ke dalam berkas teks biasa.

Berdasarkan referensi [3], berikut merupakan klasifikasi teknik steganografi berdasarkan modifikasi media penutupnya:

1. Teknik Domain Spasial

Teknik ini bekerja dengan cara memodifikasi nilai bit dari media penutup [3]. Bit yang mengandung pesan dari sebuah byte penyusun media penutup diubah secara langsung nilainya sehingga teknik ini tergolong sederhana dan memiliki kompleksitas komputasi yang rendah.

Metode yang paling umum di teknik ini adalah *LSB Modification*. LSB (*Least Significant Bits)* dipilih karena informasi yang dimuatnya tidak berkorelasi kuat dengan konten dari media penutup dan memungkinkan untuk disematkan infomrasi tanpa mempengaruhi persepsi pengguna terhadap media [4]. Metode ini dapat dibagi lagi menjadi dua, yaitu *LSB Replacement* dan *LSB Matching.*

Metode lain pada teknik ini adalah metode *Optimal Pixel Adjustment* (OPA) dan metode *Pixel Value Differencing* (PVD) [3].

2. Teknik Transformasi Domain

Teknik ini bekerja dengan cara memodifikasi nilai koefisien transfomasi dari media penutup [3]. Media penutup ditransformasikan dengan memanfaatkan fungsi matematika untuk dari satu domain ke domain lain sehingga teknik ini relatif lebih kompleks daripada teknik domain spasial [5]. Kelebihan teknik ini adalah proteksi lebih baik tehadap kompresi gambar [3].

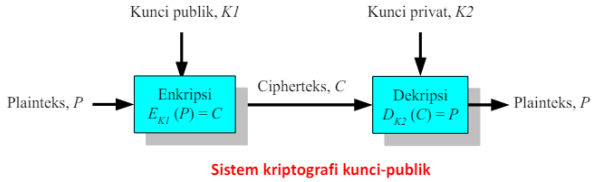
Metode yang umum digunakan pada teknik ini adalah metode yang memanfaatkan fungsi *Discrete Cosine Transform* (DCT) yang mentransformasi gambar dari domain spasial ke domain frekuensi. Setelah gambar ditransformasikan, *watermark* akan disematkan dengan cara mengubah nilai koefisien DCT pada gambar tersebut. Setelah proses oenyematan watermark selesai, gambar akan ditransformasikan kembali ke domain spasial melalui transformasi invers DCT untuk menghasilkan *stego image*.

Metode-metode lain pada teknik ini adalah metode *Fourier Transform*, *Contourlet Transform*, dan *Wavelet Transform* [3].

Teknik steganografi gambar dengan transformasi domain akan digunakan untuk menyematkan *watermark* pada berkas karya tanpa mempengaruhi kualitas karya secara signifikan.

B. *Asymmetric Cryptography*

Kriptografi asimetris, atau yang biasa dikenal dengan kriptografi kunci publik, adalah metode kriptografi yang menggunakan sepasang kunci yang berbeda untuk melakukan proses enkripsi dan dekripsi informasi [6]. Pasangan kunci ini tediri dari kunci publik yang dapat disebarkan oleh pemilik dan kunci probadi yang hanya boleh diketahui oleh pemilik. Ilustrasi dari skema kriptografi asimetris dapat dilihat pada Gambar 2.



Gambar 2. Alur Umum Kriptografi Asimetris

(sumber: https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi-dan-Koding/2023-2024/10-Kriptografi-Kunci-Publik-2024.pdf)

Pembangkitan pasangan kunci umumnya melibatkan penerapan algoritma kunci asimetris yang memanfaatkan persoalan NP dan sifat matematis [7]. Pasangan kunci bekerja untuk menyelesaikan fungsi satu arah yang hanya dapat diselesaikan apabila dipasangkan secara sesuai.

Metode ini memiliki teknik dengan fungsinya masing-masing. Contoh umum teknik kriptografi asimetris adalah algoritma Rivest-Shamir-Adleman (RSA) dan *Eliptic Curve Cryptography* (ECC).

1. *RSA*

   RSA adalah algoritma kriptografi asimetris yang dikenalkan pada tahun 1977 oleh Ron Rivest. Adi Shamir, dan Leonard Adleman. RSA memiliki tingkat keamanan yang tinggi sehingga umum digunakan di berbagai macam tempat. Algotima RSA memanfaatkan persoalan pemkatoran bilangan bulat. Keamanan pada algoritma bergantung pada kesulitan dalam memfaktorkan hasil perkalian dua bilangan prima yang sangat besar [8].

   RSA sendiri relatif lambat dan mahal dalam segi komputasi sehingga tidak praktis untuk digunakan dalam enkripsi informasi secara langsung. Umumnya RSA digunakan untuk mengirim kunci simetris yang nantinya akan digunakan oleh pengguna untuk berkomunikasi dengan aman.

   RSA memiliki komponen utama sebagai Berikut:

   1. p dan q (bilangan prima)

      Dua buah bilangan prima yang sangat besar dan bersifat rahasia.

   2. n (modulus)

      Hasil perkalian p dan q yang bersifat tidak rahasia dan berperan sebagai modulus pada tahap enkripsi dan dekripsi.

   3. φ(n) (*Euler's totient)*

      Hasil perkalian (p-1) dan (q-1) yang bersifat rahasia.

   4. e dan d (eksponen)

      Dua pasang kunci yang berperan sebagai eksponen dalam tahap enkripsi dan dekripsi. Eksponen e berperan sebagai kuci publik dan eksponen d berperan sebagai kunci pribadi.

   5. m (pesan)

      Pesan dalam bentuk *plaintext* yang bersifat rahasia.

   6. c (pesan terenkripsi)

      Pesan dalam bentuk *ciphertext* yang bersifat tidak rahasia.

   RSA terdiri dari tiga tahap utama, yaitu

   1. Pembangkitan kunci

      Kunci akan dibangkitkan dengan cara memilih p dan q yang unik dan acak. Perhitungan kemudian dilakukan untuk mengitung n dan φ(n).

      e kemudian didapatkan dengan syarat $1 < e < \varphi(n)$ dan dapat memecahkan $\gcd(e, \varphi(n)) = 1$. Eksponen d didaparkan dengan memecahkan $d \equiv e^{-1} \pmod{\varphi(n)}$.

   2. Enkripsi

      Enkripsi dilakukan dengan cara membagi-bagi m menjadi blok-blok pesan yang lebih kecil lalu menghitung persamaan $c \equiv m^e \pmod{n}$.

   3. Dekripsi

      Dekripsi dilakukan dengan cara menghitung persamaan $m \equiv c^d \pmod{n}$ dan menggabungkannya menjadi pesan semula.

   Pada skema *watermarking* yang akan diimplementasikan di makalah, peran eksponen e dan d akan dibalik.

2. *ECC*

   ECC adalah algoritma kriptografi asimetris yang dikenalkan pada tahum 1985 oleh Neal Koblitz dan Victor S. Miller. ECC memiliki panjang kunci yang lebih pendek daripada algoritma asimetris lainnya, seperti RSA [9].

   Algoritma ECC memanfaatkan persoalan logaritma diskrit kurva eliptik. Struktur kurva eliptik sendiri adalah kurva yang didefinisikan oleh persamaan $y^2 = x^3 + ax + ab$. Keamanan pada algoritma ECC sendiri bergantung pada kemampuan untuk menghitung perkalian titik dan kesulitan dalam menentukan angka

yang dikalikan dari informasi berupa titik asli dan titik hasil perkalian.

Selain melindungi kerahasiaan dari suatu pesan, kriptografi asimetris juga dapat membuktikan keaslian dari suatu pesan. Kunci publik hanya dapat digunakan dengan kunci pribadi pasangannya, dan sebaliknya. Hal ini dapat digunakan untuk memverifikasi pencipta karya dengan cara memasangkan pasangan kunci publik dan melihat apakah *watermark* yang dihasilkan benar.

III. RANCANGAN DAN IMPLEMETASI SOLUSI

Dari permasalahan yang diangkat pada bab pendahuluan, solusi yang diberikan di akhir berupa penerapan *watermarking* digital menggunakan skema *key-based steganography*. Makalah ini akan berfokus pada implementasi *watermark* digital pada karya berupa gambar digital. Solusi ini akan menggabungkan dua konsep yang telah ditinjau lebih dalam sebelumnya, yaitu steganografi dan kriptografi asimetris.

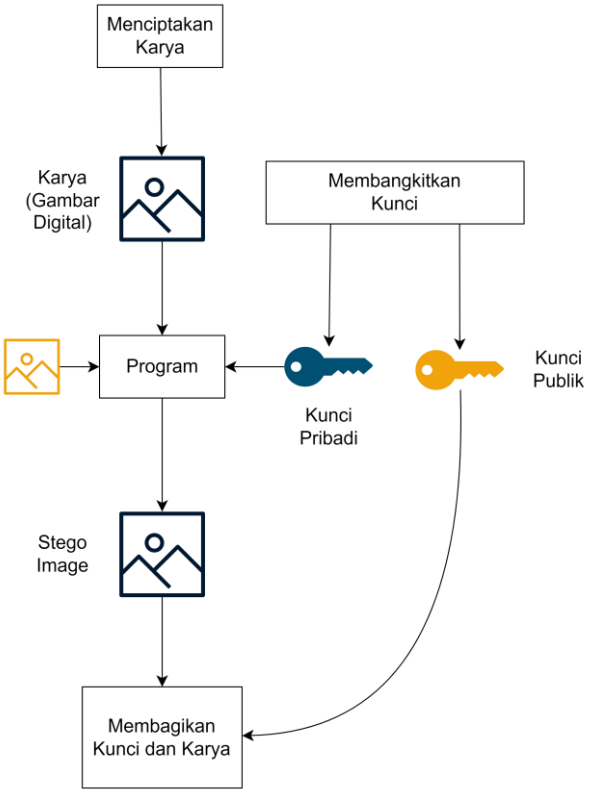
Teknik steganografi yang akan digunakan adalah steganografi teknik domain spasial menggunakan metode modifikasi *Least Significant Bits* (LSB). Solusi ini dipilih karena menawarkan kompleksitas komputasi yang paling kecil dibandingkan dengan metode atau teknik lain. Sementara itu, algoritma kriptografi asimetris yang akan digunakan adalah algoritma RSA. Solusi yang akan diimplementasikan nantinya akan digunakan untuk memverifikasi keaslian sebuah karya dengan cara menyematkan *watermark* pencipta karya ke dalam gambar tanpa mempengaruhi pengalaman orang lain dalam menikmati karya tersebut.

Proses *watermarking* akan dilakukan setelah pencipta karya telah menghasilkan karyanya dalam bentuk gambar digital yang dapat berupa foto atau ilustrasi. Pencipta karya akan menyematkan *watermark* yang dapat berupa logo atau penanda lain yang identik dengannya dengan memasukkan gambar karya, gambar penanda sebagai *watermark*, dan kunci privat miliknya sebagai sebuah *seed* ke dalam program. Setelah proses *watermarking* selesai, pencipta dapat membagikan karyanya ke interntet untuk dilihat dan dibagikan oleh orang lain.

Apabila pihak lain yang ingin memverifikasi keaslian karya tersebut, mereka dapat memasukkan karya tersebut ke dalam program yang sama dengan input kedua berupa kunci publik milik pencipta karya. Program kemudian akan mengekstrak *watermark* dari gambar tersebut dan memberikannya ke pengguna. Apabila gambar *watermark* yang dikeluarkan sesuai dengan *watermark* pencipta, maka keaslian dari karya tersebut dapat dibuktikan.

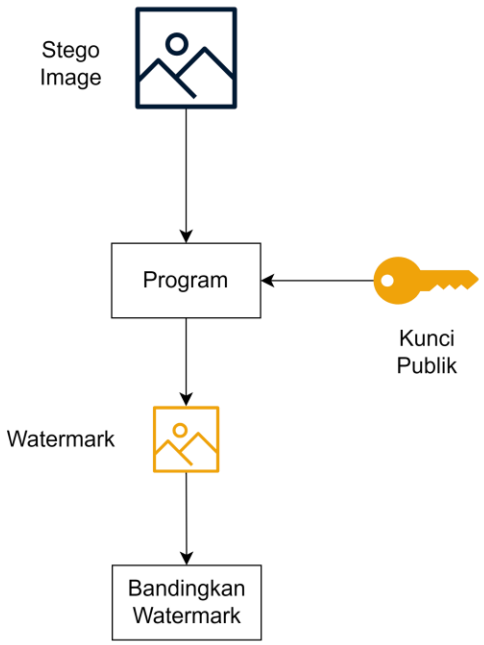
Solusi ini memiliki beberapa keuntungan. Keuntungan pertama adalah penggunaan kriptografi kunci asimetris memastikan bahwa nhanya pencipta dari karya asli itu saja yang dapat menyematkan *watermark* ke dalam gambar. Siapapun dapat memverifikasi keaslian karya tersebut menggunakan kunci publik yang diasumsikan akan dibagikan oleh pencipta karya tersebut. Solusi ini juga diharapkan lebih efektif dibandingkan skema *watermarking* tradisional yang menempelkan *watermark* secara langsung ke gambar.

Berikut merupakan diagram alir proses dari sisi pencipta karya yang dapat dilihat pada Gambar 3.



Gambar 3. Diagram Alir Penggunaan Program dari Sisi Pencipta Karya

Berikut merupakan diagram alir proses dari sisi pengguna lain yang ingin memverifikasi keaslian karya yang dapat dilihat pada Gambar 4.



Gambar 4. Diagram Alir Penggunaan Program dari Sisi Pengguna Lain

Berikut merupakan cuplikan program yang dibuat oleh penulis sebagai bentuk implementasi Solusi.

```
key = RSA.generate(2048)
private_key = key
public_key = key.public_key()
print("Kunci pribadi baru:", private_key.export_key().decode())
print("Kunci publik baru:", public_key.export_key().decode())
```

Gambar 5. Program Pembangkit Kunci dan Pembangkitan Kunci di Awal

Untuk memudahkan pengguna, kunci baru selalu dibangkitkan di awal program dan nantinya akan dicetak ke layar di fungsi main(). Penulis berhasil mengimplementasikan fungsi embed_watermark() yang menerima input *path* ke gambar yang akan disematkan *watermark* dan *path* ket *watermark* yang akan digunakan.

```
def embed_watermark(original_image, watermark):
    global private_key

    selected_image = cv2.imread(os.path.join('source_image', original_image), cv2.IMREAD_COLOR)
    selected_watermark = cv2.imread(os.path.join('watermark', watermark), cv2.IMREAD_COLOR)

    # Mengecek apakah ukuran watermark lebih kecil (watermark harus selalu lebih kecil )
    if (selected_watermark.shape[0] > selected_image.shape[0] or selected_watermark.shape[1] > selected_image.shape[1]):
        print('Error')
        return
    else:
        selected_watermark_flattened = selected_watermark.flatten()

    # Kunci -> Seed
    private_key_bytes = private_key.export_key(format='DER')
    seed = int.from_bytes(token_bytes(32) + private_key_bytes, 'big')
    rng = np.random.default_rng(seed)

    # Proses Penyematan
    indices = rng.choice(len(selected_image.flatten()), len(selected_watermark_flattened), replace=True)
    for i, idx in enumerate(indices):
        selected_image.flat[idx] = (selected_image.flat[idx] & ~1) | (selected_watermark_flattened[i] & 1)

    return selected_image
```

Gambar 6. Fungsi embed_watermark()

Fungsi embed_watermark() akan mengecek terlebih dahulu apakah gambar asli lebih besar daripada *watermark* yang akan disisipkan. Apabila gambar asli lebih besar, program akan mengubah *watermark* menjadi array satu dimensi. Program kemudian akan membangkitkan *seed* dari kunci publik dan melakukan penyematan dengan *seed* yang didapatkan menggunakan operasi *bitwise* terhadap gambar asli. Berikut merupakan gambar asli dan *watermark* yang digunakan beserta gambar stego yang dihasilkan.

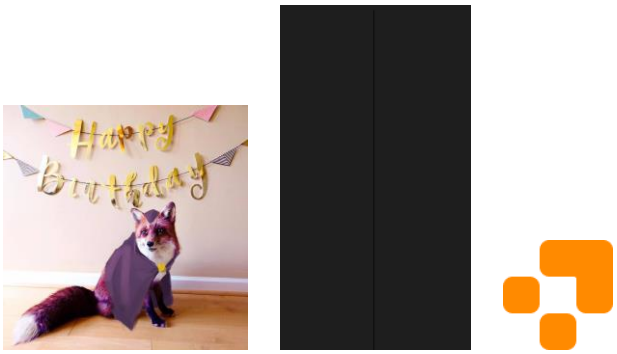
Gambar 7. (Dari Kiri) Gambar Asli, *Watermark*, dan Gambar Stego

Sayangnya, akibat ketidakahlian penulis, penulis gagal mengimplementasikan fungsi kedua yang seharusnya melakukan ekstraksi *watermark* dari gambar stego dengan argument *path* menuju gambar stego.

```
def extract_watermark(stego_image):
    global public_key

    stego = cv2.imread(os.path.join('stego_image', stego_image), cv2.IMREAD_COLOR)

    # Kunci -> Seed
    public_key_bytes = public_key.export_key(format='DER')
    seed = int.from_bytes(public_key_bytes, byteorder='big')
    rng = np.random.default_rng(seed)

    # Proses Ekstraksi
    indices = rng.choice(len(stego.flatten()), len(stego.flatten()) // 3, replace=False)
    selected_watermark_flattened = np.zeros(len(indices), dtype=np.uint8)
    for i, idx in enumerate(indices):
        selected_watermark_flattened[i] = stego.flat[idx] & 1

    # Proses Rekonstruksi Watermark
    selected_watermark_shape = (selected_watermark_flattened.size // 3, 1, 3)
    selected_watermark = np.reshape(selected_watermark_flattened, selected_watermark_shape).squeeze()

    return selected_watermark
```

Gambar 8. Fungsi embed_watermark() yang Gagal Diimplementasikan

Fungsi extract_watermark() seharusnya mengekstrak *watermark* dengan menggunakan kunci publik sebagai pembangkit *seed* tersebut nantinya akan digunakan untuk menyusun kembali *watermark* dengan cara melakukan traversal hasil enumerasi *random number generator* menggunakan *seed* tadi. Selanjutnya *watermark* akan disusun ulang dan dikembalikan oleh fungsi ini. Berikut merupakan stego image dan hasil ekstraksi *watermark* yang gagal.



Gambar 9. (Dari Kiri) Gambar Stego, *Watermark* Hasil Ekstarksi, dan *Watermark* Seharusnya

Menu utama berupa *command line interface* dengan empat opsi, yaitu bangkitkan kunci, sematkan *watermark*, ekstrak *watermark*, dan keluar. Pengguna akan diminta untuk memilih opsi dengan memasukkan angka dan nantinya pengguna juga akan diminta untuk memasukkan nama berkas gambar asli, gambar *watermark*, dan gambar hasil penyematan *watermark*.

```
Pilih opsi:
1. Bangkitkan kunci
2. Sematkan Watermark
3. Dapatkan Watermark
4. Keluar dari program
Masukkan pilihan: 2
Masukkan nama berkas Watermark: wm.png
Masukkan nama berkas output untuk Gambar Stego: st.png
Gambar stego disimpan sebagai st.png
Pilih opsi:
1. Bangkitkan kunci
2. Sematkan Watermark
3. Dapatkan Watermark
4. Keluar dari program
Masukkan pilihan: 3
Masukkan nama berkas gambar stego: st.png
Watermark diekstrak dan disimpan sebagai extracted_watermark.png
```

Gambar 10. Gambar Antarmuka Program

IV. KESIMPULAN DAN SARAN

Menghormati dan mengakui hak cipta milik seorang pencipta karya adalah sebuah bentuk rasa hormat atas usahanya dalam menciptakan karya asli. Banyak cara yang dapat dilakukan untuk melindungi hak cipta pada media digital yang disebarluaskan di internet. Salah satu caranya adalah penggunaan *watermark* sederhana maupun *watermark* digital yang telah dicoba implementasinya pada makalah ini. Dari kajian teori yang telah dilakukan, teknik *digital watermark with key-based steganography* secara teori mempunyai beberapa keuntungan. Keuntungan-keuntungan tersebut berupa kemampuan untuk memverifikasi secara pasti opencipta karya dan *watermark* yang disematkan tidak mengganggu persepsi pengamat karya. Pencipta karya dapat dengan mudah menyematkan *watermark* dengan program yang telah diimplementasikan dan harapannya akan dikembangkan fungsi untuk mengekstraksi *watermark* pada gambar sego pada implementasi nyatanya kelak.

Penulis berharap rancangan dari skema *watermarking* ini dapat dikembangkan lebih jauh nantinya. Saat ini hasil implementasi masih berupa purwarupa sederhana dari sistem asli dan masih dapat dikembangkan lebih lanjut, terutama dalam implementasi fungsi untuk melakukan ekstraksi dari gambar stego yang gagal diimplementasikan oleh penulis.

PRANALA GITHUB

https://github.com/haidar2003/makalah-kriptografi

REFERENSI

[1] A. Jammi, Y. Raju, S. Munishankaraiah, and K. Srinivas, "Steganography: An Overview," in International Journal of Engineering Science and Technology, vol. 2, no. 10, pp. 5985-5992, 2010.

[2] Maheswari. S., D. Jude, "Different Methodology for Image Steganography-Based Data Hiding: Review Paper," in International Journal of Information and Communication Technology, vol. 7, no. 4/5, pp. 521, 2015.

[3] W. M. Abduallaha and A. M. S. Rahma, "A Review on Steganography Techniques," American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS), 2016.

[4] P. Havaldar., G. Medioni, "Multimedia Systems: Algorithms, Standards, and Industry Practices" (illustrated ed.). Belmont, CA: Cengage Learning, 2009.R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[5] T. J. Pattiasina, "Studi Dan Implementasi Steganografi Metode Algoritma Dan Transformasi Pada Citra Jpeg," in Teknika, vol. 2, no. 1, pp. 46-58, doi: 10.34148/teknika.v2i1.13, 2013.

[6] D. J. Bernstein, and T. Lange, "Post-quantum cryptography," in Nature, vol. 549, no. 7671, pp. 188-194, 2017. doi:10.1038/nature23461.

[7] W. Stallings, "Cryptography and Network Security: Principles and Practice". Upper Saddle River, NJ, USA: Prentice Hall, 1990.

[8] Rivest, R. L., Shamir, A., Adleman, L.,"A method for obtaining digital signatures and public-key cryptosystems". Communications of the ACM, 21(2), 120-126. https://doi.org/10.1145/359340.359342

[9] Akkaya, M. (2016). Extension of Kerberos with X.509 and Integration of Elliptic Curve Cryptography in Authentication. *International Journal of Communications, Network and System Sciences*, 9, 603-612.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 12 Juni 2024

Muhammad Rafi Haidar (18221134)