# Implementasi Tanda Tangan Digital dan Steganografi pada Karya Seni Lukis

M. Ibnu Syah Hafizh - 13519177
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13519177@std.stei.itb.ac.id

***Abstract*—Dalam era digital yang semakin maju, tantangan dalam memastikan keaslian karya seni terutama lukisan semakin meningkat. Belum lagi, maraknya plagiarisme dan penyalahgunaan hak cipta pada karya seni lukis. Kedua hal ini tentunya sangat merugikan para seniman yang karyanya diubah atau diakui oleh orang yang tidak bertanggung jawab tanpa sepengetahuannya. Namun, hal-hal tersebut dapat dicegah melalui penerapan kriptografi, salah satunya yaitu dengan menerapkan dua topik kriptografi yakni tanda tangan digital dan steganografi. Informasi mengenai lukisan dan pelukisnya dapat disembunyikan dalam lukisan itu sendiri, serta dapat diverifikasi kembali untuk memastikan integritas dan keaslian lukisan. Pada makalah ini, dibahas penerapan tanda tangan digital dengan menggunakan ECDSA dan steganografi dengan metode LSB pada karya lukis ysng bertujuan untuk menjaga keaslian lukisan dan bukti kepemilikannya dengan menyimpan informasi terkait lukisan dan pemiliknya yang sudah ditambahkan tanda tangan digital ke dalam lukisan, sehingga distribusi penyebaran lukisan secara digital dapat dilakukan dengan aman.**

**Keywords—tanda tangan digital; steganografi; lukisan.**

## I. PENDAHULUAN

Dalam perkembangan digital yang semakin pesat, masalah keaslian dan bukti kepemilikan karya seni semakin menjadi perhatian yang serius. Khususnya, dalam konteks lukisan, tantangan yang dihadapi adalah bagaimana memastikan bahwa lukisan tersebut asli dan tidak mengalami pemalsuan, serta bagaimana menjaga bukti kepemilikannya di tengah distribusi penyebaran lukisan secara digital yang semakin luas.

Tanda tangan digital dan steganografi merupakan dua konsep kriptografi yang dapat digunakan untuk mengatasi tantangan ini. Tanda tangan digital adalah metode matematis untuk memverifikasi keaslian dan integritas suatu dokumen elektronik. Sementara itu, steganografi adalah seni menyembunyikan informasi dalam suatu medium tanpa menimbulkan kecurigaan pada pihak lain.

Dalam makalah ini, akan dibahas penerapan tanda tangan digital dengan menggunakan algoritma Elliptic Curve Digital Signature Algorithm (ECDSA) dan steganografi dengan metode Least Significant Bit (LSB) pada karya seni lukis yang didigitalisasi. Tujuan utamanya adalah menjaga keaslian lukisan serta menyimpan informasi terkait lukisan dan pemiliknya secara digital dengan menggunakan tanda tangan digital yang telah disematkan ke dalam lukisan.

Dengan menerapkan tanda tangan digital menggunakan ECDSA, dapat dipastikan bahwa lukisan tersebut tidak mengalami perubahan atau pemalsuan. Tanda tangan digital ini menggunakan matematika kriptografi yang kuat untuk menghasilkan tanda tangan unik yang hanya dapat dihasilkan oleh pemilik sah atau pihak yang memiliki kunci privat yang sesuai.

Selain itu, kami akan menggabungkan teknik steganografi dengan metode LSB untuk menyimpan informasi terkait lukisan dan pemiliknya ke dalam lukisan itu sendiri. Metode LSB memanfaatkan bit terakhir dalam representasi piksel untuk menyembunyikan data tambahan. Dengan demikian, informasi terkait lukisan dan pemiliknya dapat disembunyikan secara rahasia dalam gambar lukisan tanpa menyebabkan perubahan yang terlihat secara visual.

Dengan menggunakan kombinasi tanda tangan digital dan steganografi, dapat dicapai tujuan utama yaitu menjaga keaslian lukisan dan bukti kepemilikannya. Dengan adanya tanda tangan digital yang terintegrasi dalam lukisan, serta informasi tersembunyi yang dienkripsi menggunakan metode steganografi, distribusi penyebaran lukisan secara digital dapat dilakukan dengan aman dan dapat diverifikasi keasliannya.

Dalam makalah ini, akan dijelaskan secara rinci tentang konsep tanda tangan digital, algoritma ECDSA, teknik steganografi dengan metode LSB, serta rancangan solusi dan implementasinya dalam menerapkan kedua teknik ini pada konteks karya seni lukis. Terdapat juga studi kasus untuk menguji solusi yang dibangun.

Dengan menerapkan tanda tangan digital dan steganografi pada karya lukis, diharapkan bahwa keaslian lukisan dan bukti kepemilikannya dapat terjaga dengan baik. Ini akan memberikan kepercayaan dan rasa aman bagi seniman, pemilik lukisan, atau para kolektor yang tertarik dengan distribusi penyebaran lukisan secara digital.

## II. LANDASAN TEORI

### A. *Tanda Tangan Digital*

Tanda tangan digital (digital signature) adalah bentuk alternatif modern untuk menandatangani dokumen. Tanda tangan digital memanfaatkan skema matematika untuk memeriksa keaslian dan integritas dokumen digital. Tanda tangan digunakan untuk memberikan layanan keamanan seperti otentikasi (authentication), keaslian pesan (data integrity), dan anti-penyangkalan (nonrepudiation).

Baik tanda tangan digital maupun tanda tangan biasa mempunyai karakteristik sebagai berikut:

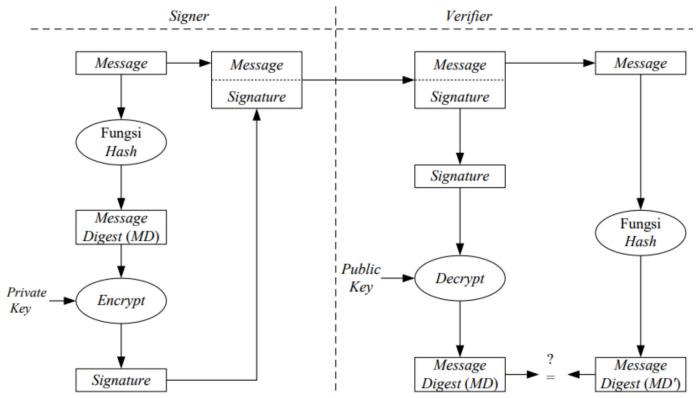
1. Tanda tangan adalah bukti yang otentik
2. Tanda tangan tidak dapat dilupakan
3. Tanda tangan tidak dapat dipindah untuk digunakan ulang
4. Dokumen yang telah ditandatangani tidak dapat diubah
5. Tanda tangan tidak dapat disangkal

Tanda tangan digital bukanlah tulisan tanda-tangan yang di-digitisasi dengan cara dipindai atau difoto. Tanda tangan digital adalah nilai kriptografis yang bergantung pada isi pesan dan kunci. Jika tanda tangan biasa akan selalu sama bentuknya jika dilakukan oleh orang yang sama, pada tanda tangan digital akan selalu berbeda walaupun ditandatangani oleh orang yang sama hal ini karena jika nilainya sama, maka akan sangat mudah untuk ditiru.

Ada dua cara dalam menandatangani pesan, pertama dengan mengenkripsi pesan. Enkripsi pesan dilakukan dengan algoritma kriptografi kunci simetris, yaitu algoritma dengan nilai kunci sama untuk proses enkripsi dan dekripsi pesan. Cara tanda tangan yang kedua yaitu dengan menggunakan kombinasi fungsi hash dan algoritma kriptografi kunci publik.

Pada tanda tangan digital dengan enkripsi pesan dengan algoritma kunci simetris, pengirim dan pengirim pesan harus memiliki kunci yang sama. Cara ini dapat memberikan solusi autentikasi, akan tetapi belum ada mekanisme anti - penyangkalan. Selain itu, isi pesan akan terenkripsi dan informasi didalamnya tidak dapat dibaca. Selanjutnya adalah dengan kombinasi fungsi hash dan algoritma kunci publik. Fungsi hash merupakan fungsi enkripsi satu arah, artinya pesan yang dienkripsi tidak akan bisa di dekripsi. Sedangkan algoritma kunci publik adalah algoritma kriptografi yang menggunakan dua buah kunci berbeda pada proses enkripsi dan dekripsi. Kunci publik digunakan untuk mengenkripsi pesan dan kunci privat digunakan untuk dekripsi pesan. Pada cara ini, isi pesan masih dapat dibaca atau tidak terenkripsi. Pada makalah ini, penulis menggunakan cara yang kedua karena informasi pada dokumen harus dapat terbaca meskipun dilakukan tanda tangan digital.

Berikut ini alur penandatanganan dengan kombinasi fungsi hash dan kriptografi kunci-publik.




Gambar II.1. Metode Tanda Tangan digital mengunakan Kriptografi kunci-publik dan Fungsi Hash

Pada makalah ini, metode tanda tangan digital yang digunakan adalah kombinasi fungsi hash dan kriptografi kunci publik karena pada permasalahan yang disebutkan tidak diperlukan kerahasiaan pesan namun yang diperlukan adalah otentikasi, keaslian pesan, dan anti-penyangkalan

### B. *ECC dan ECDSA*

ECC (Elliptic Curve Cryptography) adalah suatu pendekatan implementasi algoritma kriptografi kunci publik. ECC memanfaatkan elliptic curve pada suatu medan finite. Proses enkripsi dan dekripsi ECC dilakukan pada titik-titik yang terletak di kurva eliptik pada suatu ruang Gallois p, di mana p adalah suatu bilangan prima. Kurva eliptik pada algoritma akan memiliki persamaan berikut: $y\ 2 = x\ 3 + ax + b\ mod\ p$ dengan parameter a, b, dan p tersebut merupakan parameter dari suatu elliptic curve. ECC merupakan perluasan untuk algoritma algoritma kriptografi yang lain, misalnya:

1. ECDSA (Elliptic Curve Digital Signature Algorithm).
2. ECDH (Elliptic Curve Diffie-Hellman).
3. ECEG (Elliptic Curve ElGamal).

ECDSA (Elliptic Curve Digital Signature Algorithm) adalah suatu implementasi tanda tangan digital yang memanfaatkan elliptic curve cryptography. Terdapat dua bagian utama pada ECDSA, yaitu sign dan verify signature, Pada ECDSA terdapat beberapa parameter lain sebagai tambahan parameter elliptic curve yang digunakan (a, b, p), yaitu:

1. $G$, elliptic curve base point, yang menjadi generator subgroup pada elliptic curve yang dipakai.
2. n yang merupakan orde dari elliptic curve. Hubungan antara $n$, $G$, dan $O$ (elemen identitas) dapat dinyatakan dalam persamaan $n \times G = O$
3. d yang merupakan private key yang digunakan dalam
4. $Q$ yang merupakan public key yang digunakan dalam ECDSA.

Sebagai catatan hubungan antara $d$, $Q$, dan $G$ dapat dinyatakan dalam persamaan $d \times G = Q$. Tahapan untuk melakukan pembangkitan signature adalah sebagai berikut:

1. Hitung nilai hash h dari pesan yang ingin dibangkitkan signaturenya. Fungsi hash yang digunakan bebas asalkan aman secara kriptografi, misal SHA-256.
2. Hitung nilai z yaitu x most significant bit dari h dengan x adalah panjang bit dari n.
3. Ambil suatu angka k dari rentang $1 \leq k \leq n-1$
4. Hitung $kG = (x1, y1)$.
5. Hitung $r = x1 \ mod \ n$. Jika $r \neq 0$ lanjut ke langkah berikutnya. Jika tidak, kembali ke langkah nomor 3.
6. Hitung $s = k^{-1}(z + dr) \ mod$ n. Jika $s \neq 0$ lanjut ke langkah berikutnya. Jika tidak, kembali ke langkah nomor 3.
7. Didapat signature dari pesan masukan adalah pasangan nilai $(r, s)$.

Tahapan untuk melakukan verifikasi signature adalah sebagai berikut:

1. Cek apakah $1 \leq s, r \leq n-1$. Jika terpenuhi, lanjut ke langkah berikutnya. Jika tidak, signature tidak valid.
2. Hitung nilai $z$ dengan metode yang sama dengan langkah 1 hingga 2 pada proses pembangkitan signature.
3. Hitung nilai $s_{inv} = s^{-1} mod \ n$
4. Hitung nilai $u_1 = zs_{inv} \ mod \ n$ dan $u_2 = rs_{inv} \ mod \ n$
5. Hitung nilai $X = u_1 G + u_2 Q$. Jika X adalah titik identitas O, signature tidak valid. Jika bukan, lanjut ke langkah berikutnya.
6. Misal $(x_1, x_2)$ adalah koordinat titik. Signature valid jika dan hanya jika $r = x_1 \ (mod \ n)$

### C. *Steganografi*

Steganografi adalah ilmu dan seni menyembunyikan pesan rahasian dengan suatu cara sedemikan sehingga tidak seorang pun yang mengetahui keberadaan pesan tersebut. Tujuan dari steganografi adalah agar pesan tidak terdeteksi keberadaannya.

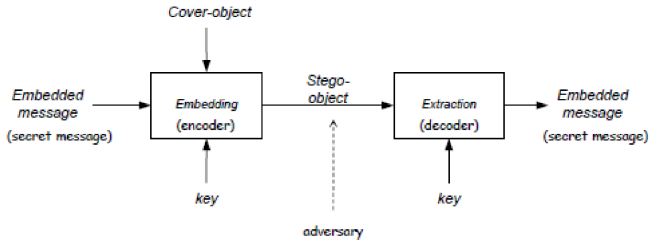
Steganografi digital merupakan penyembunyian pesan digital di dalam dokumen digital lainnya. Dokumen digital yang digunakan sebagai media untuk penyembunyian pesan dapat berupa teks, gambar, audio, dan video. Terdapat beberapa terminologi steganografi digital, yaitu:

1. Embedded message atau secret message merupakan pesan yang disembunyikan.
2. Cover-object merupakan media digital yang diguanakan untuk menyembunyikan embedded message.
3. Stego-object yaitu media yang sudah berisi pesan embedded message.
4. Stego-key yaitu kunci yang digunakan untuk menyisipkan pesan dan mengekstraksi pesan dari stego-object.

Kriteria Steganografi yang baik, yaitu:

1. Imperceptible: Keberadaan pesan rahasia tidak dapat dipersepsi secara visual atau secara audial
2. Fidelity: Kualitas cover-object tidak jauh berubah akibat penyisipan pesan rahasia
3. Recovery: Pesan yang disembunyikan harus dapat diekstraksi kembali
4. Capacity: Ukuran pesan yang disembunyikan dapat sebesar mungkin

Berikut merupakan diagram alur proses steganografi:




Gambar II.2 Alur Steganografi

Salah satu metode penyisipan pesan pada cover-obejct yang paling sering dipakai adalah metode Least Significany Byte (LSB). LSB adalah bit pada sebuah byte yang memiliki nilai yang kurang berarti untuk seluruh byte tersebut. Jika bit ini dirubah, informasi pada citra tidak akan rusak. Bitplane LSB, yaitu bitplane 0, terlihat seperti citra acak. Bitplane LSB merupakan bagian yang redundan pada citra yang berarti perubahan nilai bit ini tidak mengubah persepsi citra secara keseluruhan. Maka dari itu, metode LSB ini adalah metode yang mengganti bit LSB dari pixel dengan bit-bit pesan. Pada citra true color, terdapat 24 bit dalam sebuah pixel yang terdiri dari komponen RGB (Red-Green-Blue). Satu pixel dalam citra true color memiliki bentuk 8 bit Red, diikuti dengan 8 bit Green, dan diikuti dengan 8 bit Blue, sehingga setiap pixel berukuran 3 byte. Metode LSB pada citra true color adalah dengan mengubah bit LSB pada setiap byte RGB dalam sebuah pixel. Untuk mengekstraksi pesan dari stego-image kita hanya perlu membaca byte-byte di dalam citra, mengambil bit-bit LSBnya, dan merangkainya kembali menjadi bit-bit pesan.

## III. Rancangan Solusi dan Implementasi

Pada bagian ini, akan dibahas terkait rancangan solusi yang akan dibangun dan implementasi dari rancangan solusi tersbeut untuk menyelesaikan permasalahan yang diberikan.

### A. *Rancangan Solusi*

a. Pembuatan tanda tangan digital dan stego-image

Dalam rancangan solusi ini, tahap pertama adalah menerima informasi lukisan dari pengguna. Informasi ini mencakup nama, judul lukisan, dan tanggal pembuatan lukisan. Selanjutnya, program akan menerima input pengguna mengenai kurva yang akan digunakan dalam algoritma ECDSA. Pilihan ini bisa antara NIST, BRAINPOOLP, atau SECP, dll. Setelah semua informasi diterima, program akan melakukan hash informasi lukisan dengan fungsi hash, fungsi hash yang digunakan adalah SHA256. Hasil dari hash informasi yang di-hash adalah *message digest*.

Tahap berikutnya adalah menghasilkan kunci privat dan kunci publik berdasarkan kurva yang dipilih pengguna. Kunci privat digunakan untuk menandatangani informasi, dan kunci publik digunakan untuk memverifikasi tanda tangan tersebut. Selanjutnya, dilakukan enkripsi terhadap *message digest* tadi dengan menggunakan algoritma ECDSA dan kunci private yang telah dibangkitkan. Proses enkripsi ini akan menghasilkan tanda tangan digital

Tanda tangan digital ini kemudian digabungkan dengan informasi lukisan, membentuk satu dokumen yang siap disisipkan dalam citra lukisan. Penyisipan ini dilakukan dengan menggunakan metode steganografi LSB, bit terakhir (bit paling tidak signifikan) dari setiap pixel pada gambar akan diganti dengan bit dari dokumen yang berisi informasi lukisan dan tanda tangan digital.

Hasil akhirnya adalah "stego-image", yaitu gambar lukisan yang telah disisipkan dokumen yang berisi informasi terkait lukisan dan tanda tangan digital. Gambar ini dapat dijadikan bukti kepemilikan dan keaslian lukisan.

b. Verifikasi tanda tangan digital dari stegoimage

Pada tahap verifikasi, penerima lukisan dapat menggunakan kunci publik untuk memverifikasi tanda tangan digital. Verifikasi dimulai dengan menerima gambar yang berisi tanda tangan digital dan kunci publik yang sesuai. Kunci publik ini adalah bagian dari pasangan kunci yang dibuat selama proses penandatanganan dan digunakan untuk memverifikasi tanda tangan tersebut.

Dokumen yang berisi informasi lukisan dan tanda tangan digital yang disisipkan dalam gambar dengan teknik steganografi LSB perlu diekstraksi. Proses ini melibatkan pengambilan bit terakhir (bit paling tidak signifikan) dari setiap pixel pada gambar dan penggabungannya kembali untuk membentuk dokumen.

Setelah dokumen berhasil diekstraksi, informasi lukisan dan tanda tangan digital perlu dipisahkan. Tergantung pada bagaimana implementasi nantinya, ini bisa menemukan penanda tertentu dalam dokumen di mana informasi terkait lukisan berakhir dan tanda tangan dimulai.

Hash SHA256 dari informasi lukisan kemudian diambil, sama seperti yang dilakukan dalam proses penandatanganan. Hash ini kemudian akan digunakan untuk memverifikasi tanda tangan digital.

Tanda tangan digital yang diekstrak dari gambar dan diverifikasi dengan mendekripsi tanda tangan digital tersebut menggunakan kunci publik dan hash dari informasi lukisan. Algoritma ECDSA digunakan dalam proses ini. Jika tanda tangan berhasil diverifikasi, ini menunjukkan bahwa informasi lukisan belum diubah sejak tanda tangan dibuat dan bahwa tanda tangan itu sendiri adalah sah.

Hasil dari proses verifikasi ini adalah kepastian bahwa lukisan adalah asli dan belum diubah sejak tanda tangan dibuat. Ini memberikan bukti kuat tentang keaslian dan kepemilikan lukisan. Dengan cara ini, mereka dapat memastikan bahwa lukisan tersebut asli dan belum diubah sejak tanda tangan dibuat. Jadi, melalui penerapan ECDSA, SHA256, dan steganografi LSB, kita dapat menciptakan mekanisme efektif untuk melindungi hak cipta dan otentisitas karya seni.

## *B. Implementasi*

Rancangan solusi yang telah dijabarkan pada bagian sebelumnya akan diimplementasikan menjadi sebuah program menggunakan bahasa Python. Berikut ini adalah implementasi tanda tangan digital dan steganografi.

a. Tanda Tangan Digital

Impelementasi tanda tangan digital terdapat pada file **digital_sign.py**. Implementasi tanda tangan digital dengan bahasa pemrograman python menggunakan beberapa library, yaitu ***ecdsa***, ***hashlib***, dan ***binascii***. Untuk mengimplementasikan tanda tangan digital dengan *Elliptic Curve Digital Signature Algorithm* (ECDSA). Program ini memiliki fungsi-fungsi utama yaitu sign_message, verify_signature, show_curve_options, generate_key _pair, save_key_pair, save_message_with_signature, verify_ signature_with_public_key, main_generate _digital _signature, dan main_verify_digital_signature.

Fungsi ***sign_message(message, private_key***) digunakan untuk menandatangani pesan dengan menggunakan kunci privat. Pesan yang diterima diubah menjadi format byte menggunakan encode('utf8'), kemudian di-hash menggunakan fungsi hash SHA256. Selanjutnya, tanda tangan digital dihasilkan dengan memanggil metode sign() pada objek kunci privat. Fungsi ini mengembalikan tanda tangan digital yang dihasilkan.

Fungsi ***verify_signature(message, signature, public_key)*** digunakan untuk memverifikasi tanda tangan digital pada pesan dengan menggunakan kunci publik. Pesan yang diterima diubah menjadi format byte menggunakan encode('utf8'), kemudian di-hash menggunakan fungsi hash SHA256. Setelah itu, tanda tangan digital diperiksa menggunakan metode verify() pada objek kunci publik. Fungsi ini mengembalikan nilai True jika verifikasi berhasil, dan False jika verifikasi gagal.

Fungsi **show_curve_options()** digunakan untuk menampilkan opsi kurva yang tersedia untuk pembuatan pasangan kunci privat-publik. Opsi-opsi kurva ini mencakup beberapa pilihan standar seperti NIST256p, NIST384p, NIST521p, dan sebagainya.

Fungsi ***generate_key_pair(curve_option)*** digunakan untuk menghasilkan pasangan kunci privat-publik berdasarkan pilihan kurva yang diberikan. Pilihan kurva ini ditentukan oleh parameter curve_option. Fungsi ini akan menghasilkan objek kunci privat dan kunci publik menggunakan metode generate() dari kelas SigningKey dalam library ecdsa.

Fungsi ***save_message_with_signature(name, title, date, private_key)*** digunakan untuk menyimpan pesan yang telah ditandatangani digital beserta tanda tangan digitalnya ke dalam file .txt. Pesan yang akan disimpan terdiri dari informasi seperti nama, judul, dan tanggal yang diteruskan melalui parameter fungsi. Pesan ini akan ditandatangani menggunakan fungsi ***sign_message(),*** dan tanda tangan digitalnya akan disimpan bersama dengan pesan dalam file .txt.

Fungsi ***verify_signature_with_public_key(message, public_key***) digunakan untuk memverifikasi tanda tangan digital pada pesan yang disimpan dalam file .txt. Fungsi ini akan membaca isi file pesan, mengambil tanda tangan digital dari pesan tersebut, dan memisahkan pesan dari tanda tangan digital. Tanda tangan digital akan diubah dari format heksadesimal menjadi byte menggunakan fungsi ***binascii.unhexlify().*** Pesan kemudian di-hash menggunakan fungsi hash SHA256. Verifikasi tanda tangan digital dilakukan menggunakan fungsi ***verify_signature()***, yang mengembalikan nilai *True* jika verifikasi berhasil dan *False* jika verifikasi gagal.

**main_generate_digital_signature()** dan ***main_verify_digital_signature(pubkey_file, message_file)*** merupakan bagian dari program utama. ***main_generate_digital_signature()*** digunakan untuk menghasilkan tanda tangan digital dan pasangan kunci privat-publik, serta menyimpannya ke dalam file-file terkait. ***main_verify_digital_signature(pubkey_file, message_file)*** digunakan untuk memverifikasi tanda tangan digital pada pesan yang disimpan dalam file .txt, dengan menggunakan kunci publik yang dibaca dari file .pem.

Dengan menggunakan fungsi-fungsi tersebut, program ini dapat melakukan proses pembuatan tanda tangan digital, penyimpanan tanda tangan digital dan kunci privat-publik, serta verifikasi tanda tangan digital pada pesan yang telah ditandatangani digital sebelumnya.

b. Steganografi

Impelementasi tanda tangan digital terdapat pada file **stegano.py**. Implementasi steganografi menggunakan *Least Significant Bit* (LSB), yang bertujuan untuk menyisipkan pesan dalam gambar dan mengekstrak pesan yang disisipkan kembali dari gambar. Terdapat dua fungsi utama dalam implementasi ini.

Fungsi pertama adalah ***embed_message(image_file, message)*** digunakan untuk menyisipkan pesan ke dalam gambar menggunakan teknik steganografi LSB. Fungsi ini menerima parameter image_file yang merupakan path file gambar yang akan digunakan, dan message yang merupakan pesan yang akan disisipkan. Pertama, gambar dibuka menggunakan library PIL, kemudian daftar piksel dari gambar tersebut diambil. Pesan yang akan disisipkan diubah menjadi bentuk biner menggunakan fungsi ***message_to_binary***. Panjang pesan diuji agar tidak melebihi jumlah piksel dalam gambar. Pesan kemudian disisipkan ke dalam piksel gambar menggunakan teknik LSB. Setiap bit pesan akan disisipkan ke bit paling tidak signifikan (LSB) dalam nilai komponen merah (R), hijau (G), dan biru (B) dari setiap piksel. Piksel yang telah dimodifikasi disimpan dalam daftar modified_pixels. Selanjutnya, gambar baru yang berisi piksel yang telah dimodifikasi dibuat menggunakan ***Image.new*** dan ***putdata***. Fungsi mengembalikan gambar yang telah dimodifikasi.

Fungsi kedua adalah ***extract_message(image_file)***, yang digunakan untuk mengekstrak pesan yang telah disisipkan dalam gambar menggunakan steganografi LSB. Fungsi ini menerima parameter image_file yang merupakan *path* file gambar yang berisi pesan yang akan diekstrak. Gambar dibuka menggunakan library PIL, dan daftar piksel dari gambar tersebut diambil. Bit terakhir (LSB) dari setiap komponen R, G, dan B dari setiap piksel digunakan untuk membentuk pesan dalam bentuk biner. Proses ekstraksi pesan berhenti ketika ditemukan string ***END_OF_MESSAGE*** yang menandakan akhir pesan. Pesan biner dipotong hingga indeks yang menandakan akhir pesan, dan kemudian dikonversi kembali menjadi pesan asli menggunakan fungsi ***binary_to_message***. Fungsi mengembalikan pesan yang telah diekstrak.

Dengan menggunakan kedua fungsi ini, kita dapat menyisipkan pesan ke dalam gambar menggunakan steganografi LSB, serta mengekstrak pesan yang telah disisipkan.

## IV. Pengujian dan Pembahasan

Pada bagian ini, akan dilakukan pengujian terhadap program yang dibangun serta pembahasannya. Terdapat beberapa kasus yang dapat dilakukan untuk menguji program ini. Pada kesempatan kali ini, pengujian dilakukan dengan studi kasus. Diceritakan bahwa seorang seniman bernama abil diundang oleh seorang promotor untuk melakukan pameran di Jepang. Abil diminta oleh promotor untuk mengirimkan lukisannya secara digital. Agar terjamin keamanan dan keaslian lukisannya, Abil berencana untuk menyisipkan informasi terkait lukisannya yang sudah ditandatangani secara digital pada citra lukisannya.

Abil akan mengirimkan lukisannya yang berjudul *"Bucket of Crackers"* yang dibuat pada tanggal 8 November 2022. Sebelum citra lukisannya dikirim, ia menyusupkan dokumen yang berisi informasi pelukis dan lukisannya.

Gambar IV.1. Bucket of Crackers *(original)*

Abil menggunakan program ini untuk melakukan hal tersebut. Abil memilih kurva NIST512p untuk pembangkitan kunci. Output dari program yang dibangun adalah sebagai berikut.

```
(base) ibnuhafizh@Ibnus-MacBook-Pro crypto-in-paintings % python3 src/main.py
What do you want to do?
1. Generate Stego Digitalized Painting
2. Verify Stego Digitalized Painting
invalid input
Enter the main option: 1
Enter the artist's name: Abil
Enter the painting's title: Bucket of Crackers
Enter the painting's creation date: 08-11-2022
Available curve options for generating private keys:
1. NIST256p
2. NIST384p
3. NIST521p
4. BRAINPOOLP256r1
5. BRAINPOOLP384r1
6. BRAINPOOLP512r1
7. SECP128r1
8. SECP160r1
9. SECP256k1
10. Ed25519
Enter the curve option for generating private keys: 3
Digital signature and key pair files created successfully.
enter the painting image (.jpg): img/bucket_of_crackers.jpg
Stego-painting created successfully
```

Dokumen informasi terkait lukisan beserta tanda tangan digitalnya disimpan dalam file **message_with_signature.txt**.

message_with_signature.txt

message_with_signature.txt

```
Name: Abil
Title: Bucket of Crackers
Date: 08-11-2022
Digital Signature:
00c817c88edd69cfae8c0d5a014937a727c116c8321e
c818f5d38e611d2f84212937931aac5d3aacfa08fbbd
c2fef4cc103b3f2836a4135e6d50798e5a2323de031c
0066ce9debb2753a164e92a6ae722207d102c7d978f3
5fe48e7dddf19b25feb6689dabeabe882e1c952c7a7f
2ecd07ea99617b8f9c8a8c26cbaf8899a626266b9df9
-----END OF MESSAGE-----
```

Hasil pembangkitan kunci private dan kunci public tersimpan dalam file private.pem dan public.pem.

private.pem M

key > private.pem

```
1  -----BEGIN EC PRIVATE KEY-----
2  MIHcAgEBBEIBcX1sbqOrNnr4/+qFdDFfmRkAfbmSX53wRKtzkngvNhoPViD8gUC+
3  5ac26eCmlsjffCnyWouduDj688xeY/eyCtygBwYFK4EEACOhgYkDgYYABAGNYHzu
4  ZKHms/j58Cr67zuIRs2rbvBY2ywEQ5Co1ahvNjsG39XARZGjKucmd15DCtKQcH+D
5  Fbgm5ekwN0uXCjzEcAEvkxX8PzaX2rdT7YSKMMMh+Qs2rxEoUF0H9fsbqVigmrgE
6  3By5ilRnV/xXauslgrmp511AZrvq4uazgJIIOuZ6dA==
7  -----END EC PRIVATE KEY-----
8
```

public.pem M

key > public.pem

```
1  -----BEGIN PUBLIC KEY-----
2  MIGbMBAGByqGSM49AgEGBSuBBAAjA4GGAAQBjWB87mSh5rP4+fAq+u87iEbNq27w
3  WNssBEOQqNWobzY7Bt/VwEWRoyrnJndeQwrSkHB/gxW4JuXpMDdLlwo8xHABL5MV
4  /D82l9q3U+2EijDDIfkLNq8RKFBdB/X7G6lYoJq4BNwcuYpUZ1f8V2rrJYK5qedd
5  QGa76uLms4CSCDrmenQ=
6  -----END PUBLIC KEY-----
```

Program menghasilkan file Stego image yang sudah disusupkan dokumen bernama **result_image.png**.

Gambar IV.2. Bucket of Crackers *(stego)*

Terdapat 3 kasus yang dapat terjadi pada distribusi lukisan. Pertama, lukisan *"Bucket of Crackers"* terverifikasi asli dan tidak diubah sama sekali hingga lukisan tersebut dipamerkan. Kedua, lukisan *"Bucket of Crackers"* sudah diedit dengan mengubah tingkat brightness dan kontrasnya sehingga terjadi perubahan warna. Ketiga, lukisan *"Bucket of Crackers"* diekstrak oleh orang yang tidak bertanggung jawab (misalkan Carol) dan mengubah nama pelukisnya menjadi Carol sehingga diakui oleh Carol tersebut bahwa lukisan itu adalah lukisannya. Pembahasan untuk masing-masing studi kasus adalah sebagai berikut.

### *A. Lukisan Valid dan Tidak Diubah.*

Berikut adalah hasil verifikasi jika lukisan dan informasi terkait lukisan tidak diubah dalam distribusinya.

```
(base) ibnuhafizh@Ibnus-MacBook-Pro crypto-in-paintings % python3 src/mai
What do you want to do?
1. Generate Stego Digitalized Painting
2. Verify Stego Digitalized Painting
invalid input
Enter the main option: 2
Enter the painting image that want to extract: result_image.png
Enter the public key file (.pem) : key/public.pem
Digital signature is verified.
```

Dapat dilihat bahwa program mengeluarkan *output* "Digital signature is verified" pernyataan ini berarti bahwa visual lukisan dan informasi terkait lukisan tidak diubah.

### *B. Citra Lukisan Diubah*

Berikut adalah hasil verifikasi jika lukisan diubah dalam distribusinya. Pada pengujian ini, citra lukisan diubah tingkat brightness dan kontrasnya.

Gambar IV.3. Bucket of Crackers *(modified)*

Berikut adalah hasil verifikasi jika lukisan diubah visualnya,

```
(base) ibnuhafizh@Ibnus-MacBook-Pro crypto-in-paintings % python3 src/main.py
What do you want to do?
1. Generate Stego Digitalized Painting
2. Verify Stego Digitalized Painting
invalid input
Enter the main option: 2
Enter the painting image that want to extract: modified_image.png
Enter the public key file (.pem) : key/public.pem
Traceback (most recent call last):
  File "/Users/ibnuhafizh/Documents/ITB/Semester8/kripto/Makalah2/crypto-in-pa
intings/src/digital_signature.py", line 106, in verify_signature_with_public_k
ey
    signature = binascii.unhexlify(signature_str)
ValueError: string argument should contain only ASCII characters

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "/Users/ibnuhafizh/Documents/ITB/Semester8/kripto/Makalah2/crypto-in-pa
intings/src/main.py", line 31, in <module>
    main()
  File "/Users/ibnuhafizh/Documents/ITB/Semester8/kripto/Makalah2/crypto-in-pa
intings/src/main.py", line 27, in main
    main_verify_digital_signature(pubkey_file=pubkey_file, message_file="extra
cted_message_from_image.txt")
  File "/Users/ibnuhafizh/Documents/ITB/Semester8/kripto/Makalah2/crypto-in-pa
intings/src/digital_signature.py", line 144, in main_verify_digital_signature
    is_verified = verify_signature_with_public_key(message_file, public_key)
  File "/Users/ibnuhafizh/Documents/ITB/Semester8/kripto/Makalah2/crypto-in-pa
intings/src/digital_signature.py", line 108, in verify_signature_with_public_k
ey
    raise ValueError("There is no signature in image or argument not contain A
SCII characters")
ValueError: There is no signature in image or argument not contain ASCII chara
cters
```

Dapat dilihat bahwa program mengeluarkan *output* error yang menandakan bahwa tidak terdapat tanda tangan digital dalam lukisan tersebut. Hal ini dikarenakan nilai bit pada piksel dapat berubah akibat tampilan visualnya diubah.

### *C. Nama Pelukis Diubah*

Berikut adalah hasil verifikasi jika informasi terkait lukisan diubah dalam distribusinya. Pada pengujian ini, Nama pelukis diubah dari 'Abil' menjadi 'Carol'

```
(base) ibnuhafizh@Ibnus-MacBook-Pro crypto-in-paintings % python3 src/main.py
What do you want to do?
1. Generate Stego Digitalized Painting
2. Verify Stego Digitalized Painting
invalid input
Enter the main option: 2
Enter the painting image that want to extract: result_image.png
Enter the public key file (.pem) : key/public.pem
Digital signature is NOT verified.
```

Dapat dilihat bahwa program mengeluarkan output "Digital signature is NOT verified" pernyataan ini berarti bahwa informasi terkait lukisan sudah diubah karena message digest hasil dekripsi digital signature dan message digest hasil peneran fungsi hash pada informasi lukisan tidak sesuai.

## KESIMPULAN

. Berdasarkan program yang dibangun dengan menerapkan tanda tangan digital menggunakan algoritma ECDSA dan steganografi dengan metode LSB pada karya seni lukis, dapat ditarik kesimpulan sebagai berikut:

1. Dengan menggunakan tanda tangan digital menggunakan algoritma ECDSA, program dapat memverifikasi keaslian dan integritas lukisan. Tanda tangan digital ini menggunakan matematika kriptografi yang kuat untuk menghasilkan tanda tangan unik yang hanya dapat dihasilkan oleh pemilik sah atau pihak yang memiliki kunci privat yang sesuai. Dengan demikian, program berhasil melindungi lukisan dari perubahan atau pemalsuan yang tidak diinginkan.
2. Program berhasil menyimpan informasi terkait lukisan dan pemiliknya secara digital dengan menggunakan tanda tangan digital yang disematkan ke dalam lukisan. Dengan demikian, program memberikan bukti kepemilikan yang sah dan dapat diverifikasi terkait lukisan tersebut. Hal ini memberikan kepercayaan dan rasa aman bagi seniman, pemilik lukisan, atau para kolektor yang tertarik dengan distribusi penyebaran lukisan secara digital.
3. Melalui penggunaan teknik steganografi dengan metode LSB, program berhasil menyembunyikan informasi terkait lukisan dan pemiliknya ke dalam lukisan itu sendiri tanpa menyebabkan perubahan yang terlihat secara visual. Dengan demikian, program berhasil menjaga kerahasiaan informasi yang disematkan dalam gambar lukisan, sehingga informasi tersebut hanya dapat diakses oleh pihak yang memiliki akses yang sah.
4. Dengan menggunakan kombinasi tanda tangan digital dan steganografi, program ini berhasil memastikan

bahwa distribusi penyebaran lukisan secara digital dapat dilakukan dengan aman dan dapat diverifikasi keasliannya. Seniman, pemilik lukisan, atau para kolektor dapat dengan yakin menyebarkan lukisan secara digital, mengetahui bahwa keaslian lukisan dan bukti kepemilikannya terjaga dengan baik.

5. Dari percobaan yang telah dilakukan, program berhasil menangani berbagai kasus yang mungkin dapat terjadi dalam distribusi lukisan secara digital, seperti lukisan yang terjaga keasliannya dan tidak diubah, lukisan yang sudah diubah visualnya, atau informasi yang terkandung dalam lukisan sudah diubah.

Berdasarkan poin-poin di atas, dapat disimpulkan bahwa program yang dibangun berhasil diimplementasikan dan berhasil memenuhi tujuannya dalam menjaga keaslian lukisan dan bukti kepemilikannya melalui penggunaan tanda tangan digital dan steganografi. Program ini memberikan solusi yang efektif untuk mengatasi masalah keaslian dan bukti kepemilikan karya seni dalam konteks lukisan yang didigitalisasi.

## TAUTAN PROGRAM

https://github.com/ibnuhafizzh/crypto-in-paintings

## PENUTUP

Segala puji bagi Allah SWT yang telah memberi kemudahan dan kelancaran dalam menusun makalah ini sehingga dapat diselesaikan dengan baik dan tepat waktu. Terima kasih kepada kedua orang tua dan teman-teman yang memberi dukungan secara moral maupun doa selama penyusunan makalah. Terima kasih kepada Bapak Dr. Ir. Rinaldi Munir, M.T selaku dosen mata kuliah kriptografi yang telah memberi ilmu yang menjadi pondasi dasar dari makalah ini. Terima kasih kepada Nursyifa Salsabila (Abil) selaku seniman yang karyanya digunakan pada makalah ini. Disadari masih terdapat kekurangan dan kesalahan kata dalam makalah ini, diharapkan makalah ini dapat berguna dan bermanfaat serta dikembangkan lebih jauh sehingga berdampak untuk masyarakat luas.

## REFERENCES

[1] Munir, Rinaldi. 2023. Slide Kuliah IF4020 Kriptografi: Elliptic Curve Cryptography (Bagian 1)

[2] Munir, Rinaldi. 2023. Slide Kuliah IF4020 Kriptografi: Elliptic Curve Cryptography (Bagian 2)

[3] Munir, Rinaldi. 2023. Slide Kuliah IF4020 Kriptografi: Fungsi Hash

[4] Munir, Rinaldi. 2023. Slide Kuliah IF4020 Kriptografi: Tanda Tangan Digital

[5] Munir, Rinaldi. 2023. Slide Kuliah IF4020 Kriptografi: SHA-3 (Kecak)

[6] Munir, Rinaldi. 2023. Slide Kuliah IF4020 Kriptografi: Steganografi (Bagian 1)

[7] Munir, Rinaldi. 2023. Slide Kuliah IF4020 Kriptografi: Steganografi (Bagian 2)

## 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, 22 Mei 2023

M. Ibnu Syah Hafizh
13519177