BAB II
TINJAUAN PUSTAKA

2.1. Kajian Pustaka

Pada penulisan skripsi ini, peneliti telah mengkaji/melakukan studi literatur terhadap penelitian-penelitian sebelumnya untuk mendapatkan informasi sebagai bahan perbandingan hasil penelitian ini, baik kelebihan ataupun kekurangannya. Penelitian-penelitian sebelumnya juga dimanfaatkan informasinya sebagai landasan teori agar penelitian ini lebih sempurna dan teori yang dipaparkan valid. Berikut adalah beberapa jurnal yang digunakan sebagai perbandingan dalam penelitian ini:

1. Penelitian oleh Wiyanto dengan judul "Aplikasi Steganografi Pada Citra Digital Dengan Metode *Least Significant Bit* (LSB)," tahun 2019 [13].

   Berdasarkan penelitian tersebut, media yang digunakan pada penyisipan pesan adalah media gambar atau biasa disebut dengan *Image Steganography*. Metode yang digunakan pada penelitian ini adalah *Least Significant Bit* (LSB). Batasan permasalahan yang digunakan adalah bahasa pemrograman yang digunakan yaitu Visual Basic 6.0, pesan yang disisipkan berbentuk teks dan media gambar yang digunakan memiliki format .BMP, .JPG, dan .GIF. Hasil dari penelitian tersebut adalah pesan yang disisipkan ke dalam gambar dengan format .BMP, .JPG, dan .GIF berhasil dengan baik dan pesan dapat diekstrak dari gambar tersebut.

   Kekurangan dari penelitian tersebut adalah belum menggunakan teknik kriptografi sehingga mengurangi tingkat keamanan kerahasiaan data, apabila ada pihak yang berhasil mengekstrak data dari *hidden file* maka pihak tersebut akan langsung mendapatkan data rahasianya.
2. Penelitian oleh Jane Irma Sari dan Hengki Tamando Sihotang dengan judul "Implementasi Penyembunyian Pesan Pada Citra Digital Dengan

Menggabungkan Algoritma Hill Cipher Dan Metode *Least Significant Bit* (LSB)," tahun 2017 [14].

Pada penelitian tersebut, media yang digunakan adalah media gambar sehingga teknik steganografinya bisa disebut dengan *Image Steganography*. Media gambar yang digunakan memiliki format BMP (*Bitmap*). Teknik pengamanan data tambahan yang digunakan untuk melakukan enkripsi pesan tersembunyi adalah menggunakan Kriptografi Hill Cipher. Pesan yang disisipkan berbentuk teks, yaitu kata "HALO". Berdasarkan penelitian tersebut didapatkan hasil yaitu:

1) Proses penyembunyian pesan pada citra digital aman dan tidak diketahui secara kasat mata, karena besar bitmap hasil steganografi tidak mengalami perubahan setelah proses penyisipan biner teks ke dalam biner bitmap menggunakan metode *Least Significant Bit* (LSB) yaitu penggantian bit terakhir sehingga kapasitas bitmap sebelum dan sesudah disteganografi tidak mengalami perubahan yang berarti.
2) Pengujian pesan teks menggunakan algoritma Hill Cipher berhasil dilakukan sesuai tepat dengan alur atau langkah-langkah sehingga menghasilkan *ciphertext* yang berupa pengacakan huruf abjad.
3) Pesan yang akan diambil dari bitmap dapat dilanjutkan ke proses dekripsi yang bertujuan untuk mengembalikan pesan *ciphertext* ke bentuk semula (*plaintext*) melalui proses dekripsi algoritma Hill Cipher yang sesuai.
4) Pengujian dilakukan dengan menjalankan aplikasi *encode* dan *decode*, dapat diambil kesimpulan bahwa pengungkapan data kembali berhasil dan pesan teks dapat diambil.
5) Pengujian dilakukan dengan menggunakan Visual Basic 2010, pesan yang disembunyikan pada citra digital dengan format bitmap tersebut menggunakan metode *Least Significant Bit* (LSB) dapat terdeteksi.

3. Penelitian oleh Erwin, Gregorius Satia Budhi, dan Rolly Intan dengan judul "Aplikasi Steganografi Pada Video Dengan Teknik *Least Significant Bit* Dan Gabungan Enkripsi Rivest Cipher 4," tahun 2018 [15].

   Berdasarkan jurnal tersebut, terdapat beberapa batasan masalah yang digunakan yaitu:

   1) Bahasa pemrograman yang digunakan adalah C#.
   2) Algoritma kriptografi yang digunakan adalah Rivest Chiper 4 yang merupakan salah satu jenis algoritma kunci simetris yang berbentuk *stream cipher*.
   3) Media yang digunakan dalam penyisipan pada penelitian ini adalah video dengan format AVI, dan *frame* video yang digunakan memiliki format BMP (*Bitmap*).
   4) Pesan yang disisipkan berbentuk sebuah *file*, namun tidak disebutkan jenis format *file* yang digunakan.
   5) Metode yang digunakan pada steganografi adalah metode *Least Significant Bit* (LSB).

   Hasil dari penelitian ini adalah proses penyembunyian *file* dan ekstraksi *file* berhasil dengan baik. Hasil uji PSNR untuk mengetahui efisiensi *byteperframe* dari *frame* asli sebesar 40 dB dan *frame* yang berisi *file* rahasia sebesar 41,6555 dB. Batas teratas dari percobaan efisiensi adalah 32% yaitu 41,268 dB, sehingga dapat disimpulkan bahwa *frame* dengan *file* rahasia memiliki efisiensi yang baik. Hasil penelitian berikutnya adalah ukuran pada *file carrier* atau media penyisipan pesan menjadi lebih besar dibandingkan dengan ukuran aslinya, hal tersebut disebabkan karena steganografi dengan metode *Least Significant Bit* (LSB) menyebabkan video yang menyimpan *file* rahasia tidak bisa dikompresi.

   Kekurangan dari penelitian ini adalah hasil video steganografi tidak dapat dilakukan kompresi akibat adanya *file* data yang disembunyikan pada LSB dari warna *pixel*.

4. Penelitian oleh Septya Maharani, Ismiatul Maula, dan Zainal Arifin dengan judul "Steganografi Menggunakan Metode *End Of File* (EOF)", tahun 2018 [16].

   Penelitian tersebut memparkan bahwa terdapat lima batasan masalah yang digunakan, yaitu sebagai berikut:

   1) Bahasa pemrograman yang digunakan adalah Visual Basic.
   2) Format video yang digunakan adalah *.mp4.
   3) Pesan teks maksimal 200 karakter.
   4) Durasi video maksimal 60 detik.
   5) Metode yang digunakan yaitu EOF.
   6) *Size* video ketika disisipkan akan bertambah.

   Hasil yang diperoleh dari penelitian tersebut adalah penyisipan pesan rahasia menggunakan metode EOF ke dalam video berhasil dan lamanya proses steganografi ditentukan oleh banyaknya frame dari sebuah video. Kekurangan dari penelitian ini yaitu terjadi perubahan ukuran *file* setalah dilakukan ekstraksi/pengambilan data semula dari video.

5. Penelitian oleh Abdi Ansor yang berjudul "Penerapan Steganografi Video Dengan Metode *Discrete Cosine Transform*", tahun 2020 [17].

   Berdasarkan dari pemaparan penelitian tersebut, batasan masalah yang digunakan yaitu menggunakan video dengan format 3GP *grayscale* (hitam putih) dan tanpa audio/suara sebagai media penyisipan pesan. *Frame* video yang diekstrak memiliki format JPEG dan metode steganografi yang digunakan adalah *Discrete Cosine Transform* (DCT). Pesan yang disisipkan berupa pesan teks. Hasil dari penelitian tersebut adalah penyisipan pesan menggunakan metode DCT berhasil dilakukan, namun kualitas video yang dihasilkan bergantung pada besarnya ukuran pesan yang disisipkan. Kekurangan dari penelitian ini adalah masih menggunakan format video 3GP, berwarna hitam putih, dan tanpa audio. Selain itu, pada penelitian tersebut masih belum dipaparkan tentang pengambilan data tersembunyi/ekstraksi pesan tersembunyi dari video.

Dari kelima jurnal atau penelitian yang sudah dijelaskan di atas, perbedaan kelima jurnal tersebut dengan penelitian ini adalah sebagai berikut:

Tabel 2.1. Perbedaan penelitian sebelumnya dengan penelitian saat ini

Penelitian Sebelumnya			**Penelitian Saat Ini**
No.	**Judul, Nama Peneliti, dan Tahun**	**Perbedaan**	**Perbedaan**
1.	Aplikasi Steganografi Pada Citra Digital Dengan Metode *Least Significant Bit* (LSB). Wiyanto. 2019.	• Menggunakan bahasa pemrograman Visual Basic. • Media penyisipan pesan berupa gambar. • Pesan yang disisipkan berupa teks. • Format gambar yang digunakan .BMP, .JPG, dan .GIF. • Tidak ada pengamanan data tambahan.	• Menggunakan bahasa pemrograman Python. • Media penyisipan pesan berupa video. • Pesan yang disisipkan berupa *file*. • Format video yang digunakan adalah *.mp4.* • Menggunakan teknik kriptografi untuk pengamanan data tambahan.
2.	Implementasi Penyembunyian Pesan Pada Citra Digital Dengan Menggabungkan Algoritma *Hill Cipher* Dan Metode *Least Significant Bit* (LSB). Jane Irma Sari dan Hengki Tamando Sihotang. 2017.	• Menggunakan bahasa pemrograman Visual Basic. • Media penyisipan pesan berupa gambar. • Pesan yang disisipkan berbentuk teks. • Format gambar yang digunakan adalah *.bmp*. • Algoritma kriptografi yang digunakan adalah *Hill Cipher*.	• Menggunakan bahasa pemrograman Python. • Media penyisipan pesan berupa video. • Pesan yang disisipkan berupa *file*. • Format video yang digunakan adalah *.mp4.* • Algoritma kriptografi yang digunakan

Penelitian Sebelumnya			Penelitian Saat Ini
No.	**Judul, Nama Peneliti, dan Tahun**	**Perbedaan**	**Perbedaan**
			adalah Fernet.
3.	Aplikasi Steganografi Pada Video Dengan Teknik *Least Significant Bit* Dan Gabungan Enkripsi *Rivest Cipher* 4. Erwin, Gregorius Satia Budhi, dan Rolly Intan. 2018.	• Menggunakan bahasa pemrograman C#. • Media penyisipan pesan berupa *file* namun tidak diketahui format yang digunakan. • Format video yang digunakan adalah *.avi.* • *Frame* video memiliki format *.bmp.* • Algoritma kriptografi yang digunakan adalah Rivest Cipher 4.	• Menggunakan bahasa pemrograman Python. • Media penyisipan pesan yang berupa *file* dengan format *.docx dan .pdf.* • Format video yang digunakan adalah *.mp4.* • *Frame* video memiliki format *.png*. • Algoritma kriptografi yang digunakan adalah Fernet.
4.	Steganografi Menggunakan Metode *End Of File* (EOF). Septya Maharani, Ismiatul Maula, dan Zainal Arifin. 2018	• Menggunakan bahasa pemrograman Visual Basic. • Metode steganografi yang digunakan adalah *End Of File* (EOF). • Durasi video maksimal 60 detik. • Tidak diketahui format *frame* video yang digunakan. • Pesan yang disisipkan berupa teks dengan jumlah maksimal 200 karakter	• Menggunakan bahasa pemrograman Python. • Metode steganografi yang digunakan adalah *Least Significant Bit* (LSB). • Durasi video maksimal 30 detik. • Format *frame* adalah *.png.* • Pesan yang disisipkan berupa *file* dengan format *.docx* dan *.pdf.*

Penelitian Sebelumnya			Penelitian Saat Ini
No.	**Judul, Nama Peneliti, dan Tahun**	**Perbedaan**	**Perbedaan**
		•	
5.	Penerapan Steganografi Video Dengan Metode *Discrete Cosine Transform*. Abdi Ansor. Tahun 2020.	• Bahasa pemrograman yang digunakan tidak diketahui. • Menggunakan metode steganografi *Decrete Cosine Transform* (DCT). • Format video yang digunakan adalah *.3gp.* • Video hitam putih dan tanpa audio. • *Frame* video memiliki format JPEG. • Pesan yang disisipkan berupa teks. Tidak memiliki fitur ekstraksi pesan, yaitu proses untuk mendapatkan pesan rahasia dari video yang sudah disisipi pesan rahasia.	• Menggunakan bahasa pemrograman Python. • Menggunakan metode steganografi *Least Significant Bit* (LSB). • Format video yang digunakan adalah *.mp4.* • Video berwarna dan memiliki audio. • *Frame* video memiliki format PNG. • Pesan yang disisipkan berupa *file.* • Memiliki fitur ekstraksi pesan, yaitu proses untuk mendapatkan pesan rahasia dari video yang sudah disisipi pesan rahasia.

2.2. Dasar Teori

2.2.1. Steganografi

Steganografi merupakan suatu ilmu dan seni menyembunyikan pesan rahasia sedemikian rupa, sehingga keberadaan pesan tidak terdeteksi oleh indera manusia [18]. Menurut Saputro dan Abdullah (2019), steganografi merupakan jenis komunikasi yang tersembunyi, yang secara harfiah berarti

"tulisan tertutup". Pesannya terbuka, selalu terlihat, tetapi tidak terdeteksi bahwa adanya pesan rahasia [19]. Menurut Tambunan dkk (2018), steganografi adalah sebuah seni atau sebuah ilmu yang diimplementasikan untuk menyisipkan pesan rahasia dengan berbagai cara, sehingga hanya orang yang dituju saja yang dapat mengetahui maksud dan tujuan dari pesan tersebut [20]. Berdasarkan beberapa pendapat tersebut, dapat disimpulkan bahwa steganografi merupakan seni atau ilmu untuk menyembunyikan pesan rahasia dengan cara apapun ke dalam suatu media dan hanya diketahui oleh pihak pengirim dan penerima pesan.

Beberapa istilah yang sering digunakan dalam steganografi yaitu sebagai berikut [21]:

1. *Hiddentext* atau *embedded message* ; pesan yang disembunyikan.
2. *Covertext* atau *cover-object* ; pesan yang digunakan untuk menyembunyikan *embedded message*.
3. *Stegotext* atau *stego-object* ; pesan yang sudah berisi *embedded message*.

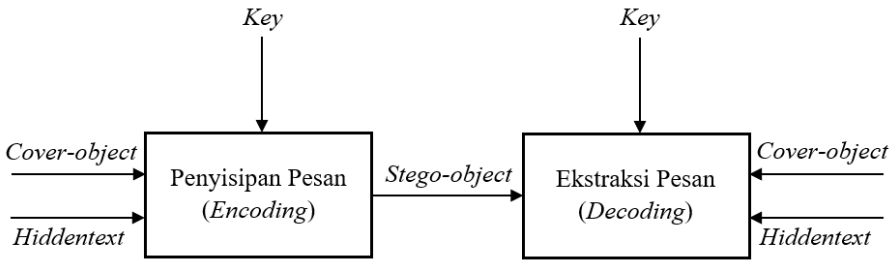
Steganografi dapat diterapkan ke dalam media penyisipan pesan atau *covertext*. Media penyisipan adalah suatu media untuk menyimpan pesan rahasia yang digunakan dalam teknik steganografi antara lain sebagai berikut [21]:

1. Teks, dalam algoritma steganografi yang menggunakan teks sebagai media penyisipannya biasanya digunakan teknik NLP (*Natural Language Processing*) sehingga teks yang telah disisipi pesan rahasia tidak akan dicurigai orang yang melihatnya.
2. Audio, format ini pun sering dipilih karena biasanya berkas dengan format ini berukuran relatif besar sehingga bisa menampung pesan rahasia dalam jumlah yang besar pula.
3. Citra, format ini juga paling sering digunakan karena format ini merupakan salah satu format *file* yang sering dipertukarkan dalam dunia internet. Alasan lainnya adalah tersedianya banyak algoritma steganografi untuk media penampung yang berupa citra.

4. Video, format ini memang merupakan format dengan ukuran *file* yang relatif sangat besar, tetapi jarang digunakan karena ukurannya yang terlalu besar, tetapi jarang digunakan karena ukurannya yang terlalu besar itu mengurangi kepraktisannya.

Steganografi memiliki tujuan yang sama dengan kriptografi yaitu untuk menyembunyikan pesan. Perbedaan kedua teknik tersebut terletak pada hasilnya. Kriptografi menghasilkan *output* data yang jauh berbeda dari bentuk aslinya dan sangat berantakan sehingga tidak dapat dibaca dan diketahui apa informasi dari data tersebut. Namun, dengan adanya data yang berantakan tersebut membuat seseorang yang ahli dalam kriptografi dapat mengetahui bahwa di dalam data tersebut terdapat informasi yang disembunyikan. *Output* dari steganografi memiliki bentuk persepsi yang sama dengan bentuk aslinya, persepsi yang dimaksud yaitu terletak pada indera manusia seperti penglihatan dan pendengaran. Akan tetapi, apabila *output* tersebut dicek melalui komputer atau perangkat digital lainnya, dapat diketahui perbedaannya secara jelas [22].

Cara kerja steganografi yaitu, *hiddentext* atau pesan yang akan disisipkan ke dalam *cover-object* (media penyisipan pesan) akan melalui proses penyisipan pesan yang disebut *encoding*. Proses tersebut akan menggunakan kunci yang hanya diketahui oleh pihak pengirim dan penerima pesan rahasia. Kunci tersebut digunakan untuk meningkatkan keamanan pesann sehingga pesan tersebut tidak mudah didapatkan oleh pihak lain. Setelah proses *encoding* selesai maka akan dihasilkan *ouput* yaitu *stego-object.* Ukuran pada *stego-object* bisa saja berubah tergantung dengan metode steganografi yang digunakan. Untuk mendapatkan pesan rahasia atau *hiddentext* kembali maka akan dilakukan proses *decoding* atau ekstraksi pesan. Proses tersebut membutuhkan kunci yang hanya diketahui oleh pihak yang berhak saja. Hasil dari proses *decoding* adalah *hiddentext* dan *cover-object*/media penyisipan pesan yang utuh seperti saat sebelum penyisipan pesan [23].



Gambar 2.1. Cara kerja steganografi secara umum

Ada beberapa metode steganografi yang dapat digunakan untuk menyembunyikan pesan, antara lain sebagai berikut [9]:

1. *Least Significant Bit* (LSB), metode ini merupakan pendekatan paling sederhana untuk menyembunyikan data dalam *file* citra. Metode ini menggunakan bit terakhir pada citra digital sebagai bit yang tidak penting lalu digantikan dengan *pixel* dari pesan rahasia.
2. *Most Significant Bit* (MSB), pada metode ini, *pixel* dari pesan rahasia akan disisipkan ke bit paling awal dari citra digital.
3. *Spread Sepctrum*, merupakan metode pentransmisian dengan menggunakan *pseudonoise code*, yang independen terhadap data informasi, sebagai modulator bentuk gelombang untuk menyebarkan *energy* sinyal dalam sebuah jalur komunikasi (*bandwidth*) yang lebih besar daripada sinyal jalur komunikasi informasi. Kemudian sinyal dikumpulkan kembali oleh penerima menggunakan replica *pseudonoise code* tersinkronisasi. Proses penyisipan pesan menggunakan metode ini terdiri dari tiga proses, yaitu *spreading*, modulasi, dan penyisipan pesan ke citra. Sedangkan proses ekstraksi pesan menggunakan metode ini terdiri dari tiga proses, yaitu pengambilan pesan dari matriks frekuensi, demodulasi, dan *despreading*.
4. *Discrete Cosine Transform* (DCT), merupakan metode yang mentransfrormasikan sebuah informasi dari domain ruang atau waktu ke dalam domain frekuensi dengan tujuan untuk mempercepat transmisi, mengurangi penyimpanan di dalam memori, menyediakan representasi *compact*, dan lain-lain.

5. *Discrete Wavelet Transform* (DWT), merupakan metode yang dapat membagi informasi dari suatu citra menjadi pendekatan dan detail sinyal. LL *band* meliputi koefisien *low pass* dan pendekatan terhadap suatu citra serta detail *sub signal* lainnya yang menunjukkan rincian vertikal, horizontal, atau diagonal atau perubahan di dalam suatu citra digital.
6. *Bit-Plane Complexity Segmentation* (BPCS), merupakan metode yang tidak berdasarkan teknik pemrograman, tetapi teknik yang menggunakan sifat penglihatan manusia. Proses penyisipan pesan dilakukan pada segmen yang memiliki kompleksitas yang tinggi. Segmen yang memiliki kompleksitas tinggi ini disebut *noise-like regions*. Pada segmen-segmen ini penyisipan dilakukan tidak hanya pada *least significant bit*, tapi pada seluruh *bitplane* yang termasuk *noise-like regions*. Oleh sebab itu, pada teknik BPCS, kapasitas data yang disisipkan dapat mencapai 50% dari ukuran *cover-object*-nya.

2.2.2. Kriptografi

Kriptografi adalah sebuah metode yang dapat mengubah data teks asli menjadi data teks baru yang tidak dapat dibaca seperti aslinya [24]. Menurut Harahap (2017), kriptografi adalah ilmu menulis pesan rahasia dengan tujuan menyembunyikan makna pesan tersebut [25]. Widiartha dan Bhaudhayana (2018), dalam penelitiannya menyatakan bahwa kriptografi merupakan ilmu dan seni penyandian yang bertujuan untuk menjaga keamanan dan kerahasiaan suatu data [26]. Jadi, dapat disimpulkan bahwa kriptografi adalah ilmu dan seni untuk mengubah pesan menjadi pesan baru yang tidak dapat dibaca untuk dijaga keamanan dan kerahasiaanya.

Kriptografi memiliki beberapa jenis berdasarkan kunci yang digunakan yaitu ada 3, sebagai berikut [27]:

1. Kunci simetris.

   Kriptografi kunci simetris adalah kriptografi yang dalam operasi enkripsi dan dekripsinya menggunakan kunci yang sama atau dikenal sebagai *private key*. Contoh kriptografi kunci simetris yaitu DES (*Data Encyption*

Standard), 3DES, IDEA, Blowfish, Twofish, Shift Cipher, Hill Cipher *Vernam Cipher*, dan AES (*Advanced Encryption Standard*).

2. Kunci asimetris.

   Kriptografi kunci asimetris adalah kriptografi yang dalam operasi enkripsi dan dekripsinya menggunakan 2 buah kunci berbeda yang disebut dengan kunci privat (*private key*) dan kunci publik (*public key*). Contoh dari kriptografi kunci asimetris yaitu RSA (*Riverst Shamir Adleman*), DSA (*Digial Signature Algorithm*), ECC (*Elliptic Curve Cryptography*), DH (*Deffie Hellman*) dan *El Gamal*.

3. Kunci *hybrid* (gabungan dari kunci simetris dan asimetris).

   Kriptografi *hybrid* adalah kriptografi yang menggunakan model persetujuan dari kedua belah pihak baik pengirim maupun penerima, di mana *session key* digunakan untuk mengenkripsi percakapan maupun mengenkripsi pertukaran data yang terjadi. Dalam hal ini setiap *session key* hanya dapat digunakan satu kali saja sehingga untuk sesi selanjutnya harus dibuat *session key* yang baru.

Beberapa istilah yang sering digunakan dalam kriptografi adalah sebagai berikut [11]:

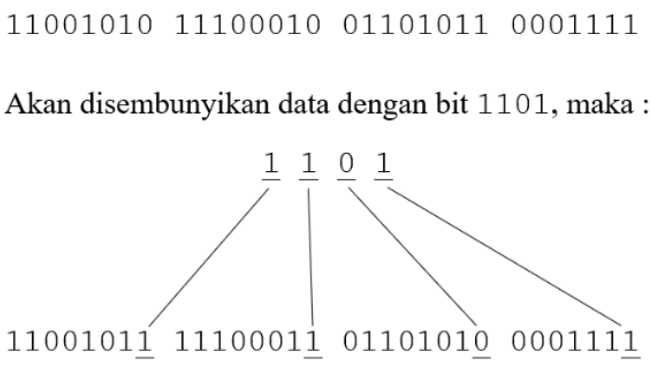
1. Pesan (*plaintext* dan *ciphertext*) : pesan (*message*) adalah data atau informasi yang dapat dibaca dan dimengerti maknanya. Pesan asli disebut plainteks (*plaintext*) atau teks-jelas (*cleartext*). Sedangkan pesan yang sudah disandikan disebut cipherteks (*chipertext*).
2. Enkripsi dan dekripsi : proses menyandikan pleinteks menjadi cipherteks disebut enkripsi (*encryption*) atau *enciphering*, sedangkan proses mengembalikan cipherteks menjadi plainteks semula dinamakan dekripsi (*decryption*) atau *deciphering*.
3. *Cipher* dan kunci : algoritma kriptografi disebut juga *cipher* yaitu aturan untuk *enciphering* dan *deciphering*, atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Kunci (*key*) adalah parameter yang

digunakan untuk transformasi *enciphering* dan *deciphering*. Kunci biasanya berupa *string* atau deretan bilangan.

2.2.3. *Least Significant Bit* (LSB)

Least significant bit adalah bagian dari barisan data biner (basis dua) yang mempunyai nilai paling tidak berarti/paling kecil. Letaknya adalah paling kanan dari barisan bit [28]. LSB adalah salah satu teknik dari steganografi yang masih sering digunakan dan masih sederhana serta sangat mudah diterapkan kedalam sebuah aplikasi. LSB ini menggunakan gambar yang berformat digital sebagai tempat penyisipan teks atau pesan. Pada susunan bit di dalam sebuah byte (1 byte = 8 bit), ada bit yang paling berarti (most significant bit atau MSB) dan bit yang paling kurang berarti (least significant bit atau LSB) [29]. Metode ini akan menghasilkan video rekontruksi yang sangat mirip dengan aslinya, karena hanya mengubah nilai bit terakhir dari data [30].

Sebagai contoh, ada sebuah *cover-object* dengan bit "11001010 11100010 01101011 0001111" akan disisipkan atau disembunyikan data dengan nilai bit "1101". Maka bit dari data tersebut akan disisipkan ke bit terakhir dari *cover-object*. Hasilnya seperti Gambar 2.2 berikut ini:



Gambar 2.2. Cara kerja metode *Least Significant Bit*

Beberapa teknik yang sering menerapkan metode LSB adalah *Image Steganography* dan *Video Steganography*. Kedua teknik tersebut menggunakan media gambar sebagai *cover-object* dalam steganografi. Media

gambar memiliki 2 jenis kompresi yaitu *lossless compresion* dan *lossy compression*. Hal tersebut sesuai dengan pendapat Sitompul dkk (2017), dalam penelitiannya menyatakan bahwa teknik kompresi citra dapat dikelompokkan menjadi dua yaitu *lossless compression* dan *lossy compression*. *Lossy compression* adalah kompresi citra di mana hasil dekompresi dari citra yang terkompresi tidak sama dengan citra aslinya karena ada informasi yang hilang, tetapi masih bisa ditolerir oleh persepsi mata. *Lossless compression* merupakan kompresi citra di mana hasil dekompresi dari citra yang terkompresi sama dengan citra aslinya, tidak ada informasi yang hilang [31].

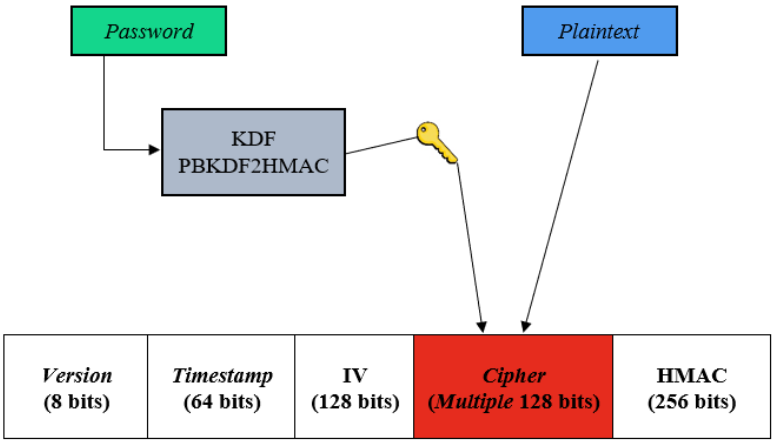
Menurut Reddy Chenna dkk. (2017), dalam penelitiannya menyatakan bahwa di antara dua jenis kompresi gambar yaitu *lossless compression* dan *lossy compression*. Format *lossless compression* menawarkan hasil yang lebih maksimal, sedangkan format *lossy compression* memungkinkan untuk tidak mempertahankan keaslian data pada gambar. *Lossless compression* menjaga keaslian data pada gambar sehingga lebih direkomendasikan dalam metode LSB [32]. Contoh format *lossy compresion* adalah JPEG, sedangkan contoh format *lossless compresion* adalah .PNG, .BMP, dan .GIF.

Pernyataan tersebut juga didukung oleh penelitian yang dilakukan oleh Sinha (2020), di dalam penelitiannya disimpulkan bahwa penerapan LSB konvensional tidak efektif pada gambar JPEG karena data dimanipulasi pada saat kompresi karena sifat kompresi tersebut adalah *lossy*, sedangkan pada gambar PNG, metode LSB sederhana dapat diterapkan tanpa kehilangan data saat kompresi. Pada format JPEG, data spasial akan diubah menjadi domain frekuensi dan menggunakan *lossy compression*, sehingga pada setiap pemrosesan dan konversi kembali ke data spasial, gambar kehilangan integritas data karena terlalu banyak *noise* dan hilangnya data. Oleh karena itu, disimpulkan bahwa steganografi ini tidak dapat dilakukan pada gambar JPEG [33].

2.2.4. Fernet

Fernet adalah metode kriptografi yang menyediakan metode otentikasi dan enkripsi sederhana menggunakan HMAC (*Hash-based Message Authentication Code*) dengan SHA256 untuk otentikasi dan AES-128 simetris dalam mode CBC (*Cipher Block Chaining*) menggunakan padding PKCS7 yang menawarkan panjang 128 bit (16 *byte*) [12]. Rotasi kunci diimplementasikan menggunakan fernet melalui MultiFernet. Enkripsi data dilakukan dengan menggunakan fernet yang sesuai dengan memori.

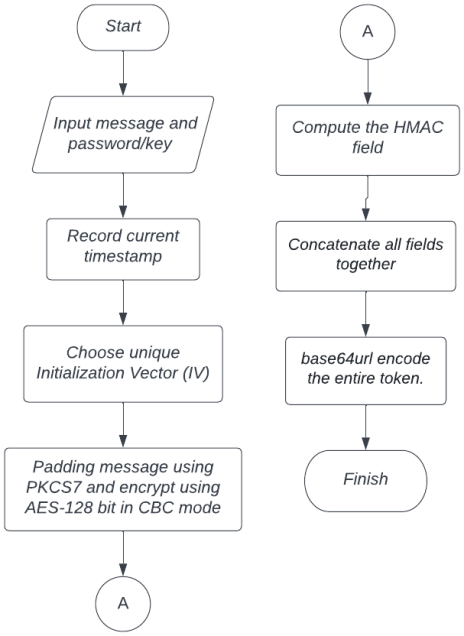
Berikut adalah skema algoritma fernet:



Gambar 2.3. Skema Fernet *System*

Enkripsi data dilakukan dengan menggunakan fernet yang sesuai dengan memori. Pesan yang dienkripsi tidak dapat dimanipulasi/dibaca tanpa kunci karena fernet adalah jenis enkripsi simetris. Fernet menerapakan URL *safe encoding* pada kunci yang digunakan [34]. Penelitian yang dilakukan Naoufal dkk. (2020), menyatakan bahwa SHA-256 adalah salah satu HMAC terbaik dalam enkripsi data, terutama karena kemampuannya untuk menghindari *collision attack* dan membuat serangan *brute force* hampir mustahil dilakukan tanpa adanya otentikasi dan kerahasiaan dengan konsep fernet *encryption* [35].

Selanjutnya, Gambar 2.4. menampilkan proses enkripsi fernet.



Gambar 2.4. Proses enkripsi fernet

Proses enkripsi fernet dimulai dengan memasukkan pesan dan *password* atau *key* yang akan digunakan. Kemudian *timestamp* akan disimpan sebagai salah satu token fernet. Lalu *Initialization Vector* (IV) 128 bit dipilih yang bersifat unik namun secara acak oleh sistem. Setelah itu, pesan rahasia akan dilakukan *padding* menggunakan PKCS7 agar ukuran pesan sesuai dengan bit *blocks* yang digunakan yaitu 128 bit. Pesan yang telah dilakukan *padding* akan dienkripsi menggunakan AES-128 dengan mode CBC. Setelah dienkripsi, pesan akan diotentikasi menggunakan HMAC SHA-256 pada token *version*, *timestamp*, *IV,* dan *ciphertext* yang telah diproses sebelumnya. Kemudian hasil dari semua proses tersebut digabungkan dan proses terakhir adalah *encode* hasil tersebut menggunakan *base64url* [36].

2.2.5. *Hash-based Message Authentication Code* (HMAC)

Message Authentication Code (MAC) adalah sebuah prosedur untuk memverifikasi bahwa pesan yang diterima berasal sumber yang otentik. MAC

mengasumsikan bahwa dua entitas yang saling berkomunikasi akan berbagi *secret key* yang sama. Salah satu fungsi MAC yang sering digunakan adalah HMAC [37].

Hash-based Message Authentication Code (HMAC) adalah teknik autentikasi pesan dengan memanfaatkan fungsi hash terhadap pesan dan kemudian mengenkripsi pesan tersebut dengan menggunakan kunci *private* [38]. HMAC merupakan jenis MAC yang menggunakan *cryptographic hash function*. *Cryptographic hash function* adalah sebuah fungsi dengan masukan string biner yang panjangnya sembarang dan menghasilkan string biner dengan panjang yang tetap. HMAC dapat digunakan dengan *cryptographic hash function* iteratif, misalnya MD5, SHA-1, SHA-128, SHA-256, atau SHA-512 [37][39].

2.2.6. SHA-256

Secure Hash Algorithm (SHA) adalah fungsi hash satu arah yang dianggap aman oleh *National Institute of Standards and Technology* (NIST), dikarenakan secara pengolahan komputasi isi pesan yang dihasilkan tidak ditemukan dan tidak untuk pesan yang berbeda tidak dapat menghasilkan hasil yang sama. SHA merupakan fungsi *hash* yang bekerja satu arah, yaitu pesan yang sudah diubah menjadi *message digest* tidak dapat dikembalikan menjadi pesan semula. SHA mempublikasikan beberapa jenis SHA, yaitu SHA-0, SHA-1, SHA-224, SHA-256, SHA-384, dan SHA-512 [40].

Salah satu SHA yang menjadi bagian dari Fernet *system* adalah algoritma SHA-256. SHA-256 merupakan sebuah operasi hash yang memiliki ukuran 256-bit yang berfungsi untuk mengubah klaim dengan pemetaan string dengan panjang acak menjadi teks khusus yang disebut *message digest* [41]. Algoritma SHA-256 berfungsi untuk menjamin integrasi data sekaligus keamanan data [42].

2.2.7. AES-128

Algoritma Advanced Encryption Standard (AES) adalah suatu algoritma *block chiper* dan yang menggunakan kunci simetri pada waktu proses enkripsi

dan dekripsi. Algoritma ini dipublikasikan oleh *National Institute of Standard and Technology* (NIST) sebagai pengganti algoritma *Data Encryption Standard* DES. Algoritma AES memiliki panjang kunci yang bervariasi untuk melakukan enkripsi dan dekripsi, yaitu 128 bit, 192 bit, dan 256 bit. [43].

Algoritma AES-128 merupakan bagian dari Fernet System. Algoritma AES-128 merupakan algoritma dengan kunci simetris yang menggunakan panjang kunci 128 bit dalam proses enkripsi dan dekripsi pesan. Algoritma ini merupakan algoritma dengan ukuran blok yang tercepat dalam pemrosesannya, karena semakin kecil kunci blok yang digunakan maka semakin cepat dalam pemrosesannya [44].

2.2.8. Mode *Cipher Block Chaining* (CBC)

Block cipher adalah salah satu kategori dalam kriptografi modern menggunakan kunci simteris yang beroperasi pada blok data. Ukuran blok data merupakan kelipatan 8 bit (1 byte). *Block cipher* memiliki 4 mode operasi, yaitu *Electronic Code Book* (ECB), *Cipher Block Chaining* (CBC), *Cipher Feedback* (CFB), *Output Feedback* (OFB). Mode operasi *cipher block* yang merupakan bagian dari Fernet *System* [45].

Mode *Cipher Block Chaining* (CBC) merupakan penerapan mekanisme umpan balik pada sebuah blok bit di mana hasil enkripsi blok sebelumnya diumpan balikkan ke dalam proses enkripsi saat itu [46]. Dengan algoritma CBC, setiap blok *ciphertext* tidak hanya bergantung pada blok *plaintext*-nya tetapi juga pada seluruh blok *plaintext* sebelumnya. Algoritma ini lebih memperkuat keamanan pesan karena blok-blok *plaintext* yang sama tidak menghasilkan blok-blok *ciphertext* yang sama pula, sehingga menyulitkan kriptanalis untuk mendapatkan pesan rahasia [47].

2.2.9. *Padding Public Key Cryptography Standards* #7 (PKCS7)

Public Key Cryptography Standards (PKCS) adalah seperangkat standar untuk kriptografi kunci publik, yang dikembangkan oleh RSA *Laboratories* yang bekerja sama dengan konsorsium informal, termasuk Apple, Microsoft, DEC, Lotus, Sun dan MIT. PKCS mencakup standar implementasi algoritma-

spesifik dan algoritma-independen. PKCS *standards* memiliki 15 jenis PKCS, yaitu PKCS1 sampai PKCS15 [48][49].

PKCS yang merupakan bagian dari Fernet adalah PKCS7. Padding PKCS7 merupakan penambahan ukuran piksel dengan nilai tertentu di sekitar data *input* agar hasil dari bidang *receptive* tidak terlalu kecil sehingga informasinya tidak banyak informasi yang hilang, di mana nilai setiap *byte* yang ditambahkan adalah jumlah *byte* yang ditambahkan [50][51].

2.2.10. Video Digital

Video merupakan media elektronik untuk merekam,menyalin, memutar, menyiarkan, dan menampilkan media visual yang bergerak [52]. Video adalah suatu citra atau *frame* yang berputar secara berurutan dengan kecepatan tertentu sehingga terlihat seperti bergerak [53]. Secara garis besar video dapat dibedakan menjadi dua macam, yaitu video analog dan video digital [54].

1. Video analog merupakan video yang dibentuk dari deretan sinyal elektrik (gelombang analog) yang direkam oleh kamera dan dipancarluaskan melalui gelombang udara. Contohnya video televisi, video tape, dan film.
2. Video digital merupakan video yang dibentuk dari sederetan sinyal digital yang berbentuk, yang menggambarkan titik sebagai rangkaian nilai minimum atau maksimum, nilai minimum berarti 0 dan nilai maksimum berarti 1.

Menurut Ariani dan Batubara (2020), menyatakan bahwa ada 10 jenis *file* video yaitu AVI, MPEG, WMV, DiVX, MP4, MOV, RealMedia, ASF, 3GP, Flash Video (FLV) [55]. Pada penelitian ini yang digunakan adalah *file* video dengan jenis MP4 karena jenis video tersebut banyak dijumpai di internet.

Video digital terusun dari serangkaian *frame*. Rangkaian *frame* tersebut ditampilkan pada layar dengan durasi yang memiliki kecepatan tertentu, tergantung dari *framerate* yang diberikan. Kalau *framerate* cukup tinggi, maka mata manusia tidak dapat mencerna gambar atau *frame*, tetapi hanya dapat mencerna rangkaian yang berkelanjutan (kontinu) [56]. *Framerate* adalah jumlah bingkai gambar atau *frame* yang ditunjukkan setiap detik dalam

membuat gambar bergerak, diwujudkan dalam satuan *frames per second* (fps). Makin tinggi angka fps-nya, semakin halus gambarnya [57].

Rangkaian *frame* dapat diambil dari sebuah video menggunakan teknik Ekstraksi *frame*. Ekstraksi *frame* video adalah salah satu proses yang dilakukan dalam *computer vision* untuk mengekstrak gambar-gambar (*frame*) yang terdapat dalam video untuk dapat melakukan proses selanjutnya [58] *Frame* dapat digunakan sebagai *stego-object* untuk penyembunyian *file* dalam video steganografi.

2.2.11. Format MP4

MPEG-4 sub-bagian 14 atau lebih dikenal sebagai MP4 adalah salah satu format berkas pengodean suara dan gambar/video digital yang dikeluarkan oleh sebuah organisasi MPEG. Ekstensi nama berkas jenis MPEG-4 ini banyak menggunakan *.mp4*, dan merupakan pengembangan dari format *QuickTime* dari komputer *Apple* [19].

2.2.12. *Fast Forward Motion Picture Expert Group* (FFMPEG)

FFMPEG merupakan sebuah *tool* yang sangat berguna dalam proses *streaming, recording, editing, converting file* berbentuk audio, video, dan gambar yang dapat digunakan di berbagai platform [59]. FFMPEG adalah program komputer yang dapat perekam, mengkonversikan dan streaming audio dan video digital dalam berbagai format. FFMPEG terdiri dari kumpulan pustaka perangkat lunak bebas/*open source* [30]. Pada penelitian ini, FFMPEG digunakan untuk ekstraksi *frame* video, ekstraksi audio video, dan pembuatan video kembali dari kumpulan *frame*.

2.2.13. Format DOCX

Docx adalah jenis format untuk *file* versi Microsoft Word 2007 ke atas dan tentunya mempunyai banyak keuntungandibandingkan dengan format *file* sebelumnya. Di antaranya adalah mempunyai ukuran simpan yang relative lebih kecil,mempunyai ketahanan terhadap serangan virus saat berbagi dokumen dan masih banyak lagi [60]. Docx mempunyai performa yang cepat dan ringan [61].

2.2.14. Format PDF

PDF (*Portable Document Format*) adalah sebuah format berkas yang dibuat oleh Adobe Systems pada tahun 1993 untuk keperluan pertukaran dokumen digital. Format PDF digunakan untuk merepresentasikan dokumen dua dimensi yang meliputi teks, huruf, citra dan grafik. Format PDF merupakan format dokumen yang sangat populer karena tidak tergantung pada satu *software* [62]

2.2.15. PSNR

Metode PSNR (*Peak Signal to Noise Ratio*) adalah ukuran perbandingan antara nilai piksel *cover image* dengan nilai piksel pada citra stego yang dihasilkan. Sebelum menentukan PSNR terlebih dahulu ditentukan nilai rata-rata kuadrat absolute error antara *cover image* dengan citra stego yaitu nilai MSE (*Mean Square Error*) [63]. Nilai PSNR berbanding terbalik dengan nilai MSE. Semakin kecil nilai MSE, maka semakin baik kualitas gambarnya, akan tetapi tidak untuk nilai PSNR. Kualitas gambar akan semakin baik jika nilai PSNR yang dihasilkan semakin besar.

Rumus untuk menentukan nilai MSE adalah sebagai berikut [64]:

$$MSE_{AVG} = \frac{MSE_R + MSE_G + MSE_B}{X.Y}$$

Rumus PSNR adalah sebagai berikut:

$$PSNR = {}^{20}log\left(\frac{MAX}{\sqrt{MSE}}\right)$$

PSNR diukur dalam satuan *decibel* (dB). Nilai PSNR dapat dikatakan baik jika nilai PSNR lebih dari 40 dB [64].