***DATA HIDING* MENGGUNAKAN TEKNIK *PLAYFAIR* DAN SUBTITUSI DNA DI DATA DNA**

**SKRIPSI**

**Diajukan untuk memenuhi sebagai persyaratan mendapatkan gelar Strata Satu Program Studi Informatika**

****

**Disusun Oleh:**

**DWIKO SATRIYO. U. Y. S**

**M0515008**

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS SEBELAS MARET**

**2020**

# HALAMAN PERSETUJUAN

# HALAMAN PENGESAHAN

# MOTTO

“Bersyukut artinya mengingat segala nikmat yang didapat”

*anonymous*

# PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

kedua orang tuaku tercinta,

kakakku tercinta,

teman-teman Informatika Angkatan 2015

***DATA HIDING* MENGGUNAKAN TEKNIK *PLAYFAIR* DAN SUBTITUSI DNA DI DATA DNA**

**DWIKO SATRIYO. U. Y. S**

**Program Studi Informatika**

**Fakultas Matematika dan Ilmu Pengetahuan Alam**

**Universitas Sebelas Maret**

# ABSTRAK

Belum Ada

**Kata kunci :** *Belum ada*

***DATA HIDING* MENGGUNAKAN TEKNIK *PLAYFAIR* DAN SUBTITUSI DNA DI DATA DNA**

**DWIKO SATRIYO. U. Y. S.**

**Department of Informatics**

**Faculty of Mathematics and Natural Sciences**

**Sebelas Maret University**

# ABSTRACT

*Blank.*

**Keyword:** *Blank*

# KATA PENGANTAR

Surakarta, Agustus 2020

Penulis

# DAFTAR ISI

[HALAMAN PERSETUJUAN ii](#_Toc47367790)

[HALAMAN PENGESAHAN iii](#_Toc47367791)

[MOTTO iv](#_Toc47367792)

[PERSEMBAHAN v](#_Toc47367793)

[ABSTRAK vi](#_Toc47367794)

[ABSTRACT vii](#_Toc47367795)

[KATA PENGANTAR viii](#_Toc47367796)

[DAFTAR ISI ix](#_Toc47367797)

[DAFTAR TABEL xi](#_Toc47367798)

[DAFTAR LAMPIRAN xi](#_Toc47367799)

[DAFTAR GAMBAR xii](#_Toc47367800)

[BAB I PENDAHULUAN 13](#_Toc47367801)

[1.1. Latar Belakang 13](#_Toc47367802)

[1.2. Rumusan Masalah 13](#_Toc47367803)

[1.3. Batasan Masalah 14](#_Toc47367804)

[1.4. Tujuan Penelitian 14](#_Toc47367805)

[1.5. Manfaat Penelitian 14](#_Toc47367806)

[1.6. Sistematika Penulisan 14](#_Toc47367807)

[BAB II TINJAUAN PUSTAKA 16](#_Toc47367808)

[1.1. Dasar Teori 16](#_Toc47367809)

[1.1.1. Kriptografi 16](#_Toc47367810)

[1.1.2. Playfair Cipher 17](#_Toc47367811)

[1.1.3. Vigenere Cipher 18](#_Toc47367812)

[1.1.4. Steganografi 1](#_Toc47367813)

[1.1.5. DNA 1](#_Toc47367814)

[2.2. Penelitian Terkait 2](#_Toc47367815)

[BAB III METODOLOGI PENELITIAN 6](#_Toc47367816)

[3.1. Persiapan Data Uji Coba 6](#_Toc47367817)

[3.2. Perancangan Program 6](#_Toc47367818)

[3.3. Implementasi 7](#_Toc47367819)

[3.4. Uji Coba 7](#_Toc47367820)

[3.5. Analisa 7](#_Toc47367821)

[BAB IV HASIL DAN PEMBAHASAN 8](#_Toc47367822)

[4.1. Dataset 8](#_Toc47367823)

[4.2. Perancangan Program 8](#_Toc47367824)

[4.3. Uji Coba 8](#_Toc47367825)

[4.4. Analisa 15](#_Toc47367826)

[BAB V PENUTUP 16](#_Toc47367827)

[1.1. Kesimpulan 16](#_Toc47367828)

[1.2. Saran 16](#_Toc47367829)

[DAFTAR PUSTAKA 17](#_Toc47367830)

[LAMPIRAN 18](#_Toc47367831)

# DAFTAR TABEL

# DAFTAR LAMPIRAN

**No table of figures entries found.**

# DAFTAR GAMBAR

[Gambar 6 Metodologi Penelitian 6](#_Toc47354494)

BAB I   
PENDAHULUAN

## Latar Belakang

Kriptografi DNA adalah sebuah cabang kriptografi yang mencoba memanfaatkan sifat-sifat dari sekuen DNA untuk memperkuat ataupun membuat teknik kriptografi baru. Kriptografi DNA mensimulasi proses transkripsi dan translasi dari DNA sekuen. Hasil dari metode tersebut adalah kuat terhadap serangan tertentu, khususnya serangan *brute force*. Selain itu metode ini sangat efisien dalam komputasi, penyimpanan dan transmisi(Ning, 2009). Sedangkan Steganografi DNA adalah cabang ilmu steganografi yang menggunakan media DNA sebagai media penyembunyian data(Shimanovsky et al., 2003). Steganografi DNA memanfaatkan ukuran sekuen DNA yang rata-rata memiliki ukuran yang panjang sebagai media untuk menyembunyikan data.

*DNA Sequence* dapat dimanfaatkan untuk menjadi media enkripsi karena sifat kompleksnya yang terdiri dari 4 jenis protein pembentuk yaitu A, C, G T dan juga sekuen DNA dapat diubah ke dalam bentuk protein tersusun dari 4 sekuen DNA yang memiliki abiguitas tertentu. Selain itu, media DNA juga dapat digunakan sebagai media untuk penyembunyian data. Tidak seperti, audio atau video, yang dapat diubah sedemikian rupa sehingga secara kualitatif terlihat ketika sinyal terlalu terdegradasi, DNA sulit untuk dilakukan *filtering* untuk melihat apakah ada watermark di dalamnya (Shimanovsky et al., 2003). Oleh karena itu, penyerang terpaksa melakukan pencocokan langsung terhadap DNA yang ada.

Banyak studi mengenai kriptografi dan steganografi DNA. Paper sebelumnya yang membahas enkripsi DNA yaitu Algoritma RSA(Wang and Zhang, 2009), Playfair cipher(Atito et al., 2016)(Sabry et al., 2010), Enkripsi berbasis DNA menggunakan pointer(Amin et al., 2007), dan Enkripsi DNA menggunakan PCR(Prabhu and Adimoolam, 2011). Sedangkan untuk steganografi DNA yaitu metode *Insertion, Substitution*, dan *Complementary Pair Rule* (*Complementary Pair Rule*)(Shiu et al., 2010), *Least Significant Base Subtitution* (Khalifa, 2013), dan *Table Lookup Substitution Base* (Taur et al., 2012)

Beberapa penelitian sebelumnya ada yang melakukan pengujian terhadap algoritma-algoritma kripografi DNA. Salah satunya yaitu penelitian oleh Marwan(Marwan et al., 2016). Marwan et al, membandingkan antara vigenere cipher, playfair cipher, RSA cipher, dan AES cipher. Untuk penelitian terhadap perbandingan algoritma steganografi DNA juga ada beberapa. Salah satunya yaitu penelitian oleh Shiu(Shiu et al., 2010). Pada penelitian tersebut, dibandingkan antara algoritma *Insertion*, *Complementary Pair*, dan *Subtitution*.

Tujuan dari penelitian ini yaitu membuat algoritma kombinasi kriptografi dan steganografi dengan menggunakan algoritma kriptografi *Playfair Cipher DNA* dan algoritma steganografi *DNA Subtitution* (Subtitusi DNA). Algoritma tersebut dibandingkan dengan kombinasi algoritma yang serupa dan dicari kombinasi manakah dari Algoritma Kriptografi DNA dan Steganografi DNA yang menjadi algoritma terbaik menurut aspek waktu, maksimal panjang pesan, dan *cryptanalysis*-nya.

## Rumusan Masalah

Berdasarkan uraian pada Latar Belakang, maka dapat dirumuskan masalahnya, yaitu :

* Bagaimana menyembunyikan pesan ke dalam DNA reference menggunakan algoritma playfair dan subtitusi DNA ?
* Bagaimana perbandingan dengan algoritma serupa yang lain ?

## Batasan Masalah

Berdasarkan Latar Belakang dan Rumusan masalah, penelitian ini mempunyai batasan-batasan agar lingkup penelitian tidak meluas ke permasalahan lainnya. Batasan pada penelitian ini adalah menggunakan teknik playfair dan subtitusi DNA untuk menyembunyikan pesan rahasia

## Tujuan Penelitian

Tujuan dari penelitian yang ingin dicapai yaitu bagaimana menyembunyikan pesan rahasia ke dalam *DNA reference* menggunakan algoritma playfair dan subtitusi DNA agar hasilnya optimal.

## Manfaat Penelitian

Manfaat dari penelitian ini adalah untuk menemukan teknik hiding data dan enkripsi yang aman dan memiliki kapasitas hiding data yang besar karena menggunakan media DNA.

## Sistematika Penulisan

Penulisan Tugas Akhir ini disusun dengan sistematika sebagai berikut:

BAB I PENDAHULUAN, berisi latar belakang masalah, rumusan masalah, batasan masalah, manfaat penelitian serta sistematika penulisan tugas akhir

BAB II DASAR TEORI, merupakan tinjauan pustaka yang menguraikan tentang landasan teori yang menjadi acuan penulisan tugas akhir

BAB III METODOLOGI, berisi langkah-langkah yang dilakukan dalam melakukan penelitian tugas akhir

BAB IV HASIL DAN PEMBAHASAN, memaparkan hasil dari penelitian beserta pembahasannya

BAB V PENUTUP, merumuskan kesimpulan berdasarkan hasil dan pembahasan beserta saran yang dapat menjadi masukan untuk penelitian berikutnya

# BAB II TINJAUAN PUSTAKA

## Dasar Teori

### Kriptografi

Kriptografi adalah studi tentang teknik matematika yang terkait dengan aspek keamanan informasi seperti kerahasiaan, integritas data, otentikasi entitas, dan otentikasi asal data. Kriptografi bukan satu-satunya cara untuk memberikan keamanan informasi, tetapi hanya membahas tentang tekniknya(Menezes, A.J., Van Oorschot, P.C. and Vanstone, 1997). Tujuan yang fundamental dari Kriptografi adalah untuk memfasilitasi 2 orang agar dapat bertukar informasi pada saluran komunikasi yang tidak aman tanpa diketahui informasi yang ditransmisikan oleh orang lain(Stinson, 2006). Informasi yang dikirimkan disebut dengan “Plaintext” yang bisa berupa text, data numerik, atau segala sesuatu yang lain. .Proses konversi *plaintext* pada kriptografi disebut enkripsi. Sedangkan hasil enkripsi dari *plaintext* disebut dengan *chippertext*(Stallings, 2013). Untuk membaca *chippertext* perlu menggunakan teknik yang dinamakan dekripsi. Enkripsi dan dekripsi memerlukan kunci rahasia (*secret key*) agar tidak dapat dilakukan oleh orang yang tidak diinginkan(Mao, 2003). Secara konvensional, untuk mendapatkan saluran informasi yang aman, maka diperlukan 2 hal, yaitu algoritma enkripsi yang kuat dan 2 pihak yang berkomunikasi (pengirim dan penerima) harus sudah mendapatkan kunci rahasia dan menjaga kunci tersebut agar tidak diketahui orang lain(Stallings, 2013).

Notasi untuk Enkripsi yaitu :

C = E(M, K)

Notasi untuk Dekripsi yaitu :

M = D(C, K)

Keterangan :

C = Ciphertext

M = Message/Pesan

E = Algoritma Enkripsi

D = Algoritma Dekripsi

K = Kunci Algoritma

Sumber (Mao, 2003)

Aspek yang menjadi tujuan terakhir pada kriptografi ada 4(Menezes, A.J., Van Oorschot, P.C. and Vanstone, 1997), yaitu :

1. *Confidentiality*, yaitu kemampuan untuk menjaga informasi rahasia dari pihak yang tidak memiliki otoritas pada informasi tersebut.
2. *Data Integrity*, yaitu kemampuan untuk menjaga informasi rahasia dari manipulasi/modifikasi oleh pihak yang tidak memiliki otoritas.
3. *Authentication,* yaitu kemampuan untuk memastikan bahwa pengirim dan penerima informasi rahasia merupakan pihak yang memiliki otoritas.
4. *Non-repudian*, yaitu kemampuan untuk memastikan mahwa informasi rahasia sudah terkirim atau sudah diterima oleh pihak yang berkomunikasi.

Di samping kriptografi, terdapat teknik yang melakukan serangan untuk mendapatkan informasi rahasia. Tujuan dari serangan tersebut yaitu untuk mendapatkan kunci rahasia daripada untuk mendapatkan *plaintext* dari satu *ciphertext*. Ada dua pendekatan untuk menyerang skema enkripsi(Stallings, 2013), yaitu :

1. *Cryptanalysis,* tipe serangan ini menganalisis dan mengeksploitasi kelemahan yang dihasilkan dari karakteristik sebuah algoritma.
2. *Brute Force*, penyerang mencoba segala kemungkinan pada kunci terhadap sebuah *ciphertext* sampai mendapatkan *plaintext*-nya.

### Playfair Cipher

Playfair cipher termasuk ke dalam kategori *polygram substitution cipher*, yaitu *subtitition cipher* yang mensubtitusikan *plaintext* dalam grup karakter/lebih dari 1. Playfair cipher memecah *plaintext* ke dalam bentuk *digram* atau grup yang berisi masing-masing 2 karakter lalu dienkripsikan terhadap tabel kunci yang di-*generate* di awal. Tabel kunci berbentuk matrix 5x5 yang berisi alfabet-alfabet yang menyusun tabel kunci (Menezes, A.J., Van Oorschot, P.C. and Vanstone, 1997).

Langkah-langkah menyusun tabel kunci(Stallings, 2005) :

* Menghilangkan karakter selain karakter alfabet dari kunci (hanya a-z)
* Menghilangkan karakter ganda pada kunci sehingga hanya tersisa 1 saja
* Membuat tabel matrik 5x5 yang berisi 25 slot
* Mengisikan kunci yang sudah di-*preproccessing* tadi kedalam tabel
* Mengisikan Tabel yang belum terisi dengan alfabet yang belum ada pada tabel
* Memilih 1 slot yang berisikan 2 alfabet yang bernilai sama. Misal I dan J bernilai sama, maka dipilih salah satu huruf untuk mengisi slot yang mewakili kedua huruf tersebut

Notasi untuk langkah-langkah enkripsi dan dekripsi playfair cipher(Menezes, A.J., Van Oorschot, P.C. and Vanstone, 1997):

p1,p2 = (x1,y1),(x2,y2)

c1,c2=

c1,c2 = (i1,j1),(i2,j2)

p1,p2=

p1,p2 = digram plaintext

c1,c2  = digram ciphertext

x1,y1 = posisi baris dan kolom p1 pada tabel kunci

x2,y2 = posisi baris dan kolom p2 pada tabel kunci

i1,j1 = posisi baris dan kolom c1 pada tabel kunci

i2,j2 = posisi baris dan kolom c2 pada tabel kunci

### Vigenere Cipher

Vigenere Cipher termasuk dari bagian *Polyalphabetic Cipher* karena kunci untuk setiap karakternya bisa lebih dari 1. Enkripsi dilakukan dengan menambahkan karakter plaintext dengan karakter ciphertext. Sebaliknya, untuk dekripsi ciphertext dengan mengurangi karakter ciphertext dengan kunci(Stallings, 2013) yang dapat dinotasikan sebagai berikut :

Ci = (pi + ki mod m) mod 26

Pi = (Ci – ki mod m) mod 26

Dengan :

Ci = Ciphertext

pi = Plaintext

ki = Kunci

m = Jumlah karakter dalam alfabet

Bila menggunakan alfabet a sampai z, kemungkinan kunci untuk ditebak yaitu 26m. Artinya, semakin panjang pesan maka semakin sulit kunci untuk ditebak. Tetapi, juga memiliki konsekuensi yaitu bila panjang pesan pendek maka kunci akan semakin mudah untuk ditebak(Stinson, 2006).

### Steganografi

Steganografi adalah metode menyembunyikan informasi sehingga tidak terdeteksi oleh manusia(Wibisurya, 2017). Steganografi menyisipkan informasi digital yang dirahasiakan ke dalam media sampul (gambar, audio dan video, dll.) tanpa diketahui oleh orang lain. Sebagian besar penerapan steganografi mengikuti prinsip umum yaitu Alice, yang ingin membagikan pesan rahasia *m* dengan Bob, secara acak memilih (menggunakan sumber random yang privat *r*) sebuah pesan biasa (*harmless*), yang disebut *cover-object*, yang dapat ditransmisikan kepada Bob tanpa menimbulkan kecurigaan, dan memasukkan (*embeds*) pesan rahasia kedalam *c*, dengan menggunakan kunci *k*, yang disebut *stego-key*. Alice lalu merubah kover *c* menjadi *stego-object* *s* sehingga pihak ketiga hanya tahu bahwa yang ditransmisikan hanya kover *c* yang belum berubah menjadi *stego-object* *s*(Katzenbeisser and Petitcolas, 2000).

Singkatnya, notasi proses steganografi seperti berikut(Cox et al., 2008) :

Emb : C x K x M 🡪 C

Ext : C 🡪 M

Sehingga, Ext(Emb(c, Ks, m)) = m untuk semua c C, Ks K, dan m M. Ks diambil dari set semua kunci stego K, M merupakan set dari semua pesan yang dapat disisimkan (*embed*), dan C merupakan set dari semua kover.

Aplikasi utama steganografi meliputi komunikasi rahasia, perlindungan hak cipta, verifikasi integritas, dan otentikasi(Ghosal and Mandal, 2018). Saat mendesain skema steganografi, perlu diperhatikan aspek dari sifat/property pada saluran komunikasi, sumber *cover* yang digunakan sebagai media Steganografi, dan juga fungsi *embedding* dan ekstraksi(Cox et al., 2008).

### DNA

Dalam biologi, asam Deoksiribonukleat (DNA) adalah molekul induk yang strukturnya menyandikan semua informasi yang diperlukan untuk membuat dan mengarahkan mesin kimia kehidupan(Fenderson, 2008). Pada tahun 1953, struktur DNA diprediksi dengan benar oleh Waton dan Francis Crick bahwa molekul DNA terdiri dari dua rantai polinukleotida panjang yang masing-masing rantai ini dikenal sebagai rantai DNA, atau untai DNA yang dibuat dari subunit sederhana, yang disebut nukleotida. Setiap nukleotida terdiri dari molekul gula-fosfat dengan gugus samping yang mengandung nitrogen, atau basa. Basa terdiri dari empat jenis (adenin, guanin, sitosin, dan timin), sesuai dengan empat nukleotida yang berbeda, berlabel A, G, C, dan T(Fenderson, 2008). Setiap tiga nukleotida yang berdekatan membentuk kodon. Mengingat bahwa setiap nukleotida dapat memiliki salah satu dari empat basa kimia dan setiap kodon terdiri dari tiga nukleotida, maka ada jumlah 43 = 64 kemungkinan kombinasi berbeda. Kombinasi ini menentukan asam amino yang akan digunakan oleh organisme hidup, yang pengaturannya menentukan struktur dan fungsi protein yang dihasilkan(Madison et al., 2007).

## Penelitian Terkait

Berikut ini adalah beberapa penelitian yang berkaitan dengan penelitian yang diajukan:

1. **Steganografi methods based upon DNA sequences.** (Shiu, H., et al., 2010)

Penelitian ini memperkenalkan dan membandingkan 3 metode steganografi yang memanfaatkan sifat-sifat dari *DNA* *Sequence*. Metode tersebut yaitu *Insertion DNA*, *Complementary Pair Rule*, dan *Subtitution DNA*.

Cara kerja metode *Insertion* yaitu dengan memecah pesan rahasia dan media penyembunyian DNA (*DNA reference*) ke dalam beberapa segmen, yang lalu masing-masing segmen digabung satu per satu. Kunci dari metode ini yaitu media DNA yang digunakan untuk penyembunyian pesan.

Metode kedua yaitu *Complementary Pair Rule*. Cara kerjanya yaitu pesan rahasia disisipkan sebelum pasangan substring primer. Substring primer yaitu substring yang dibuat sebagai kunci untuk menentukan letak pesan rahasia, lalu dicari pasangan *Complementary*-nya. Pengaman dari metode ini yaitu pasangan substring primer yang disisipkan pada media DNA/*DNA Reference*.

Metode terakhir yaitu *Subtitution*. Metode ini bekerja dengan cara mengganti/mensubtitusi *DNA Reference* pada posisi yang sudah ditentukan dengan pesan yang akan disembunyikan. Kunci dari metode ini yaitu posisi dna yang disubtitusi.

1. **DNA-based data encryption and hiding using playfair and insertion techniques** (Atito, A., A. Khalifa, and S. Rida, 2012)

Penelitian ini bertujuan memperkenalkan teknik untuk melakukan komunikasi data secara aman dengan komposisi enkripsi dan steganografi dengan memanfaatkan beberapa sifat DNA

Metode yang diperkenalkan yaitu gabungan antara metode enkripsi DNA *Playfair Cipher* dan metode steganografi DNA *Insertion*.

1. **Information Hiding Based on DNA Steganography** (Wang, Zicheng., Zhao, Xiaohang., Wang, Hong., Cui, Guangzhao., 2013)

Tujuan dari penelitian ini yaitu untuk mendesain prosedur untuk menenkripsi sebuah pesan dengan menggunakan metode *message decomposing* untuk menyelesaikan masalah keamanan pada Steganografi DNA.

Menurut penelitian tersebut, masalah utama pada Steganografi DNA yaitu menyembunyikan *Primer Sequence* atau sekuen penanda letak pesan rahasia. Diusulkan metode *Decomposing*, yaitu memecah pesan rahasia yang sudah dalam bentuk ASCII ke dalam dua buah himpunan bilangan. Cara memecahnya yaitu mengambil satu-satu berurutan dari kiri ke kanan yang dalam penelitian tersebut digunakan istilah *even number* dan *uneven number*. Selanjutnya mengirimkan salah satu dari dua himpunan bilangan tersebut dengan melakukan steganografi DNA terlebih dahulu. Bila pesan tidak sampai atau diketahui oleh orang lain, maka lakukan ulang langkah pertama lalu dikirimkan hingga sampai dan tidak diketahui orang lain. Setelah itu, himpinan bilangan yang tidak dikirimkan bisa disebarkan secara publik. Sebelum pesan rahasia dirubah kedalam bentuk ASCII, dilakukan enkripsi terlebih dahulu yang pada penelitian ini digunakan metode *Vigenere Cipher*.

1. **DNA-based cryptographic methods for data hiding in DNA media** (Marwan, Samiha., Shawish, Ahmed., Nagaty, Khaled., 2016)

Penelitian ini menyajikan analisa komparasi dari algoritma *vigenere cipher*, *DNA-based playfair cipher*, *AES cipher*, dan *RSA cipher*. Masing-masing tehnik dikombinasikan dengan salah satu tehnik Steganografi DNA yaitu *substitution process*.

Yang diuji pada penelitian ini adalah kapasitas sembunyi data, Ukuran data setelah dilakukan enkripsi dan steganografi, ukuran kunci, *Cryptanalysis*, dan performa waktu.

1. **A novel data hiding method based on deoxyribonucleic acid coding** (Liu, Hongjun., Lin, Da., Kadir, Abdurrahman., 2013)

Penelitian ini mengusulkan metode penyembunyian data baru yang berbasis pada *DNA Coding*. Operasi yang dilakukan pada metode ini yaitu *plaintext* di-*encode* ke dalam bentuk sekuen DNA. Selanjutnya yaitu men-*generate* 2 buah *DNA Sequence* acak yang berfungsi sebagai kunci enkripsi dan kunci penyembunyian data. Langkah selanjutnya pesan dienkripsikan dengan metode XOR terhadap sekuen acak pertama dan digambung dengan sekuen kedua atau disebut sebagai sekuen primer. Lalu, hasil enkripsi disembunyikan ke dalam media penyembunyian yang pada penelitian ini digunakan media gambar *screenshoot* sebuah dokumen *Word*. *Ciphertext* disembunyikan ke dalam *pixel* dari media penyembunyian gambar dengan mensubtitusikan dengan setiap *fore color* dari setiap karakter. Yang terakhir yaitu gambar dapat dikirimkan langusng ke penerima atau bisa dikonversi ke dalam bentuk pdf.

1. **A technique for DNA cryptography based on dynamic mechanisms** (Biswas, Md. Rafiul., et al., 2019)

Penelitian ini menggunakan teknik *encoding* DNA untuk memperkuat algoritma enkripsi asimetrik yang sudah ada. Cara kerjanya yaitu *plaintext* dikonversi ke bentuk *binary* dari nilai ASCIInya. Lalu, dikonversi atau *encode* ke dalam basis DNA. Selanjutnya, membangkitkan nilai ASCII dari hasil konversi ke DNA tadi menggunakan *dynamic sequence table* yang dibuat oleh penulis. Berikutnya, pilah-pilah ke dalam potongan yang berukuran sama. Bila perlu, tambahkan angka 0 pada bagian kiri angka. Langkah berikutnya yaitu melakukan enkripsi dengan algoritma enkripsi asimetrik yang pada penelitian ini digunakan algoritma RSA, Elgamal, dan Paillier. Hasil enkripsi dikonversi ke bentuk basis DNA. Terakhir yang itu mengembalikan ke bantuk karakter dengan menggunakan *dynabic sequence table*. Untuk proses dekripsinya kebalikan dari proses enkripsinya.

# BAB III METODOLOGI PENELITIAN

Penelitian ini dilakukan dengan beberapa tahapan yang ditunjukkan Gambar 6 di bawah ini

Gambar 6 Metodologi Penelitian

## Persiapan Data Uji Coba

Pada tahap ini mengambil sampel data *DNA Sequence* untuk sebagai media penyembunyian data atau kover dari National Center for Biotechnology Informasi <https://www.ncbi.nlm.nih.gov/>. Data sebagai media kover juga berupa data digital yang diterjemahkan ke dalam bentuk DNA. Untuk data pesan rahasia dan kunci rahasia berupa data alfabet.

## Implementasi

Pada tahap ini, program dibangun dengan menggunakan bahasa pemrograman python. Bahasa Python dipilih karena memiliki konsep desain yang bagus dan sederhana, yang berfokus pada kemudahan dalam penggunaan. Sehingga, mudah untuk dibaca, digunakan ulang, dan dirawat. Selain itu, Bahasa Python bersifat Open Source, memiliki banyak dukungan pustaka, dan juga bisa diintegrasikan dengan Bahasa pemrograman lainnya.

Terdapat 3 langkah utama dalam proses penyembunyian pesan menggunakan *playfair cipher* dan subtitusi DNA yaitu konversi *plaintext* ke bentuk protein, enkripsi *playfair*, dan steganografi subtitusi DNA.



Figure

Ketiga langkah tersebut bila dijabarkan ke dalam *flowchart* akan seperti berikut :



Figure

Pesan rahasia sebelum masuk ke sistem persandian diubah terlebih dahulu ke dalam bentuk DNA, selanjutnya dilakukan enkripsi dan steganografi sehingga menghasilkan stego-DNA. Stego-DNA ini yang akan dikirimkan ke penerima.



Figure

Untuk mengkonversi *plaintext* ke bentuk protein, data yang masuk (berbentuk biner) terlebih dahulu dikonversikan ke bentuk sekuen DNA. Mengkonversi bentuk biner ke dalam sekuen DNA dilakukan secara berpasangan, yaitu 2 bit dikonversi ke satu karakter DNA. Aturan konversinya yaitu ‘00’ ke ‘A’, ‘01’ ke ‘C’, ‘10’ ke ‘G’, dan ‘11’ ke ‘U’. Alur lengkapnya pada *flowchart* di bawah.

Setelah di dapatkan *plaintext* dalam bentuk sekuen DNA, selanjutnya sekuen DNA di konversikan ke dalam bentuk protein untuk keperluan enkripsi di tahap selanjutnya. Konversi dilakukan dengan menterjemahkan 4 DNA ke satu protein dengan mengikuti table konversinya.

|  |  |  |  |
| --- | --- | --- | --- |
| Ala/A | GCU, GCC,  GCA, GCG | Leu/L | UUA, UUG,  CUU, CUC,  CUA, CUG |
| Arg/R | CGU,CGC,  CGA,CGG,  AGA,AGG | Lys/K | AAA, AAG |
| Asn/N | AAU, AAC | Met/M | AUG |
| Asp/D | GAU, GAC | Phe/F | UUU, UUC |
| Cys/C | UGU, UGC | Pro/P | CCU,CCC,  CCA, CCG |
| Gln/Q | CAA, CAG | Ser/S | UCU, UCC,  UCA, UCG,  AGU, AGC |
| Glu/E | GAA, GAG | Thr/T | ACU, ACC,  ACA, ACG |
| Gly/G | GGU,GGC,  GGA,GGG | Trp/W | UGG |
| His/H | CAU, CAC | Tyr/Y | UAU, UAC |
| Ile/I | AUU, AUC,  AUA | Val/V | GUU, GUC,  GUA, GUG |
| START | AUG | STOP | UAA, UGA,  UAG |

Table

Tabel diatas dibuat ke bentuk alfabet agar bisa dilakukan operasi *playfair* menjadi.

|  |  |  |  |
| --- | --- | --- | --- |
| A | GCU, GCC,  GCA, GCG | L | CUU, CUC,  CUA, CUG |
| R | CGU,CGC,  CGA, CGG | K | AAA, AAG |
| N | AAU, AAC | M | AUG |
| D | GAU, GAC | F | UUU, UUC |
| C | UGU, UGC | P | CCU,CCC,  CCA, CCG |
| Q | CAA, CAG | S | UCU, UCC,  UCA, UCG |
| E | GAA, GAG | T | ACU, ACC,  ACA, ACG |
| G | GGU,GGC,  GGA,GGG | W | UGG |
| H | CAU, CAC | Y | UAU |
| I | AUU, AUC,  AUA | V | GUU, GUC,  GUA, GUG |
| B | UAA, UGA,  UAG | O | UUA, UUG |
| U | AGA,AGG | X | AGU, AGC |
| Z | UAC |  |  |

Table

Langkah konversi dari bentuk DNA ke dalam bentuk protein bila dibuat *flowchart* menjadi seperti berikut :



Figure

Setelah didapatkan plaintext dalam bentuk protein, tahap kedua yaitu melakukan enkripsi *playfair*. Proses enkripsi *playfair* dilakukan membangkitkan tabel kunci dan melakukan subtitusi *plaintext* menggunakan table kunci yang dibuat. Alurnya dijabarkan pada *flowchart* dibawah



Figure

Tahapan ketiga atau yang terakhir yaitu melakukan steganografi terhadap hasil enkripsi sebelumnya. Hasil enkripsi diubah terlebih dahulu ke dalam bentuk biner. Teknik yang digunakan dalam proses steganografi adalah Subtitusi DNA. Data disubtitusikan ke dalam media kover penyembunyian yang disiapkan dengan teknik tersebut.



Figure

Untuk proses mendapatkan kembali pesan rahasia, tahapan yang dilakukan hanya membalikkan tahapan sebelumnya. Proses tersebut bila digambarkan ke dalam *flowchart* menjadi berikut :



Proses ekstraksi steganografi dari stego-DNA dilakukan dengan membandingkan terhadap media kover yang digunakan saat *embedding* data. Lalu dilakukan operasi ekstraksi subtitusi DNA hingga mendapatkan data yang disembunyikan. Bentuk *flowchart* dari proses ekstraksi subtitusi DNA adalah sebagai berikut :



Figure

Tahapan selanjutnya untuk mendapatkan pesan rahasia yaitu dengan proses dekripsi yang pada algoritma ini yaitu dekripsi *playfair cipher*. Proses dekripsi *playfair cipher* hamper sama dengan proses enkripsinya. Hanya posisi subtitusi pada tabel dibalik bila dibandingkan saat proses enkripsi. Prosesnya digambarkan dalam *flowchart* berikut :



Figure

Setelah proses dekripsi, pesan rahasia/*plaintext* sudah didapatkan. Tetapi, pesan rahasia masih berbentuk protein. Untuk mendapatkan pesan rahasia sesungguhnya, dilakukan proses translasi dari proten ke DNA lalu dari DNA ke bentuk biner.

Proses translasi protein ke DNA tidak jauh berbeda dengan proses dari DNA ke protein. Yang membedakan kali ini adalah penggunaan nilai ambiguitas yang didapatkan saat proses translasi dari DNA ke protein. Nilai ambiguitas menentukan nilai DNA yang dikonversikan dari sebuah protein. Untuk *flowchart* dari proses tersebut sebagai berikut :



Figure

Sama seperti proses translasi dari protein ke DNA, proses translasi DNA ke biner hanya membalikkan proses saat *encoding*. DNA diterjemahkan sesuai dengan aturannya. Yaitu “A” ke “00”, “C” ke “01”, “G” ke “10”, dan “U” ke “11”. *Flowchart* dapat dilihat berikut :



Figure

## Uji Coba

Pada tahapan ini dilakukannya uji coba terhadap algoritma yang diusulkan yang akan diujikan dengan menggunakan program yang sudah dibuat sebelumnya. Untuk metode pengujian yaitu dilakukan penghitungan running time, analisa terhadap keamanan, perhitungan selisih ukuran kover sebelum dan setelah dilakukan pengisipan, dan kesamaan atau similarity antara kover sebelum dan sesudah dilakukan penyisipan.

## Analisa

Tahapan yang terakhir yaitu analisa. Hasil uji coba akan di analisa dan disimpulkan kelebihan dan kekurangan dari algoritma yang diusulkan terhadap algoritma pembanding lainnya.

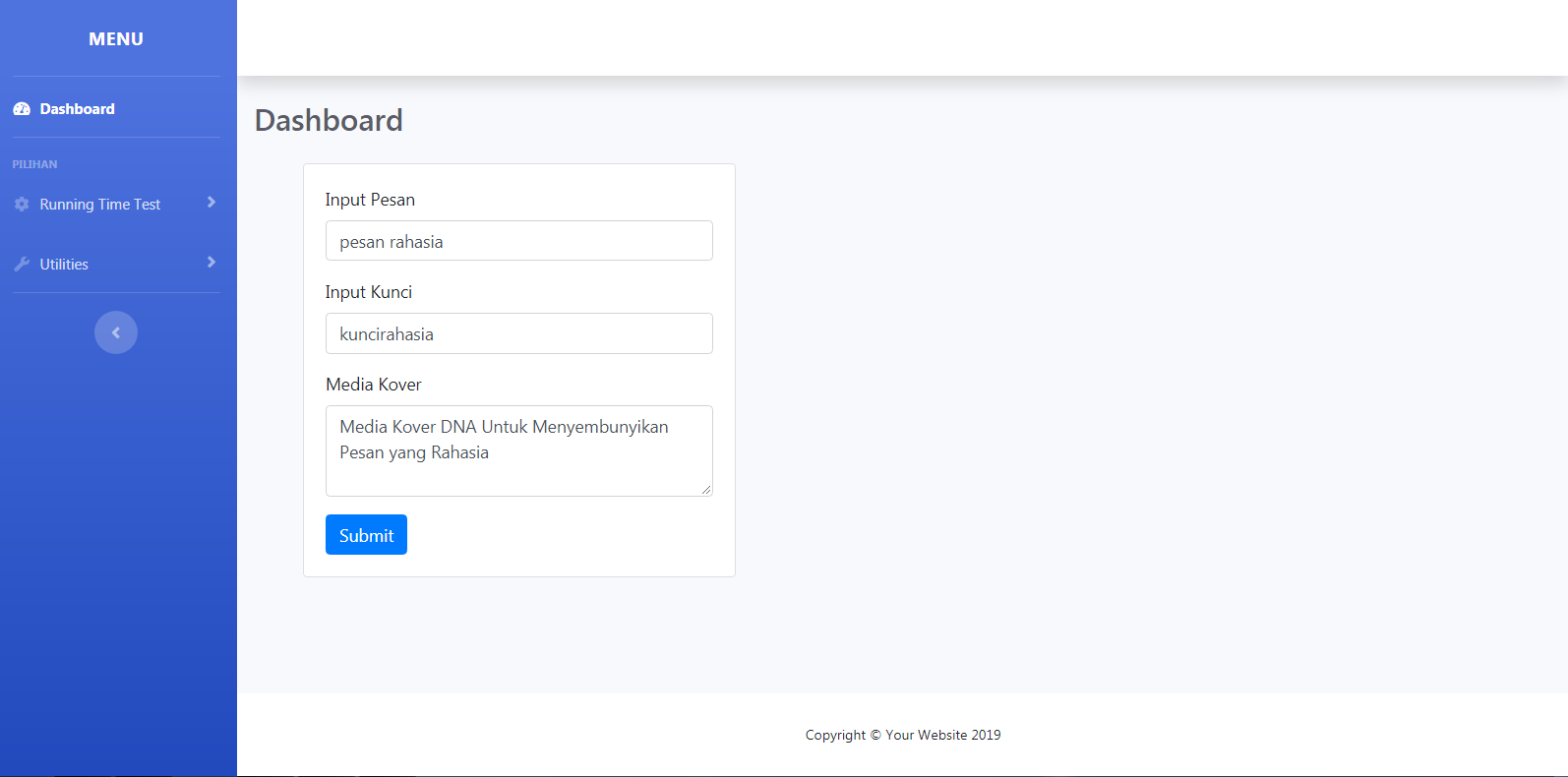
# BAB IV HASIL DAN PEMBAHASAN

## Persiapan Data Uji Coba

Data yang digunakan yaitu pesan rahasia, kunci rahasia, dan data DNA *reference* sebagai kover penyembuyian. Pesan rahasia bisa berupa data biner atau data tulisan tetapi untuk input ke dalam sistem, pesan rahasia berupa biner. Untuk kunci rahasia berupa huruf alfabet dikarenakan *playfair cipher* merupakan cipher alfabetik. Dan media kover yang dipakai berupa data DNA, bisa data DNA asli atau data digital yang dikonversi ke bentuk DNA.

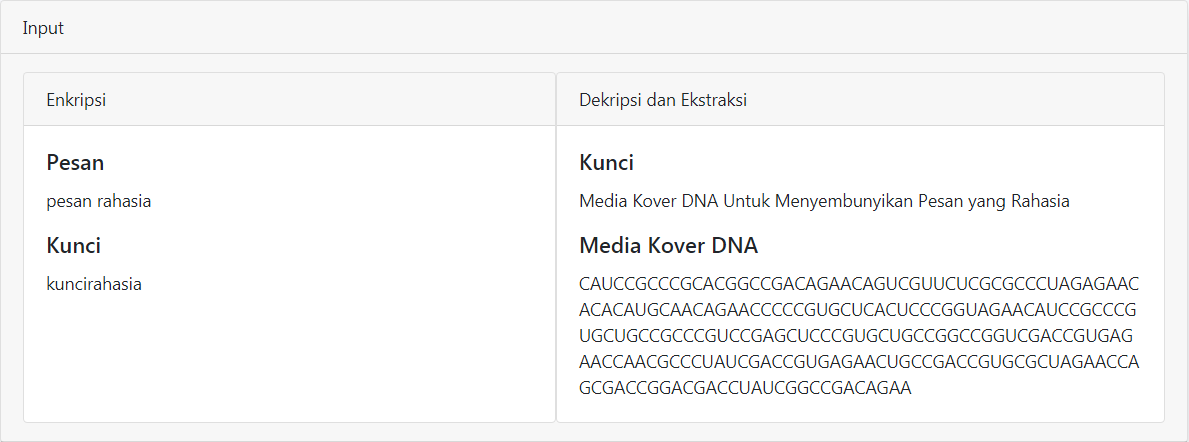
## Implementasi

Tahapan ini bertujuan untuk mengimplementasikan bagan/*flowchart* yang sudah disusun sebelumnya. Untuk tampilan utama input adalah sebagai berikut



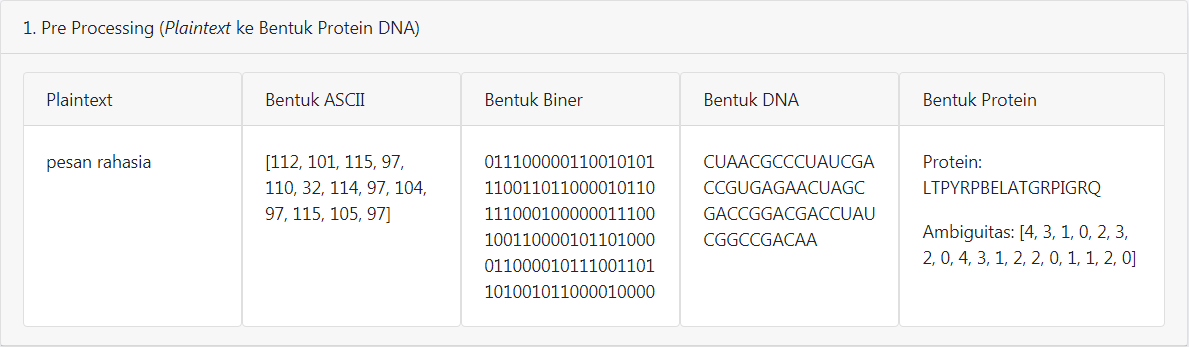
Setelah men-*submit* data, maka data akan diproses sesuai dengan alur yang sudah dirancang yang akan dibahas sebagai berikut :

1. **Input**



Input pesan berupa “pesan rahasia”, kunci berupa “kuncirahasia”, dan media kover “Media Kover DNA Untuk Menyembunyikan Pesan yang Rahasia”. Media kover yang masih berupa data digital diubah ke dalam bentuk DNA.

1. **Encode Protein**

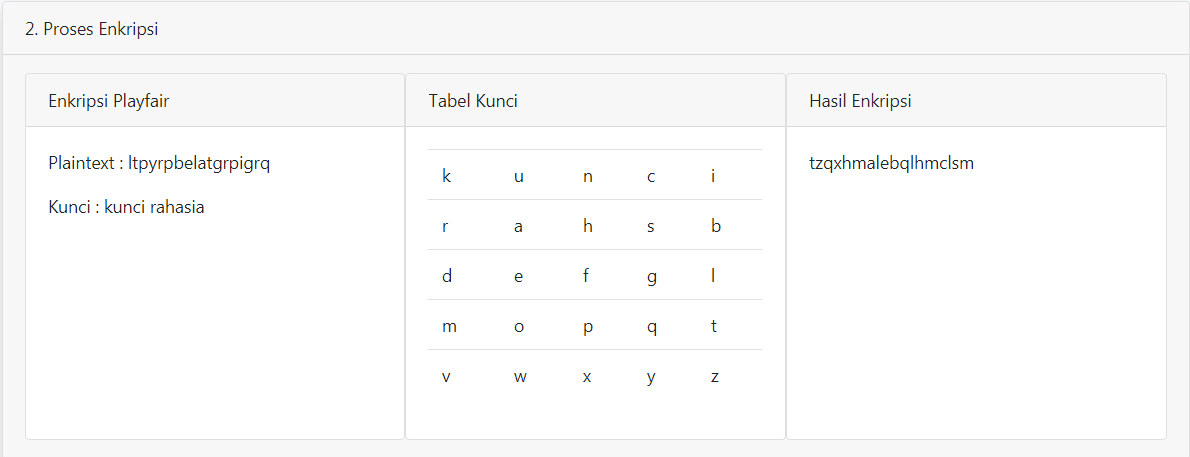


Terhadap pesan rahasia, dilakukan tahap *pre-processing* yaitu merubah ke dalam bentuk protein. Ada 2 tahapan yang perlu dilakukan, yaitu yang pertama merubah dari data digital ke data DNA dan yang kedua merubah data DNA ke bentuk protein.

Tahapan pertama yaitu merubah data digital ke data DNA menggunakan aturan konversi. Sebagai contoh diambil 2 karakter pertama pada *plaintext* yaitu “p” dan “e”. Karakter tersebut dalam tabel ASCII bernilai 112 dan 101. Nilai tersebut diubah ke bentuk biner menjadi 01110

Tahapan kedua yaitu menkonversi data DNA ke protein perlu menggunakan sebuah tabel protein. Nantinya kelompok yang terdiri dari 3 buah DNA dicocokkan ke dalam tabel protein sehingga menghasilkan 1 buah protein yang dibentuk oleh 3 buah DNA. Dalam tabel protein, satu protein bisa dibentuk oleh lebih dari 1 kelompok DNA. Oleh sebab itu, diberikanlah angka ambiguitas untuk sebagai pengenal kelompok DNA mana yang menjadi pembentuk protein tersebut. Sebagai contoh, diambil 6 karakter pertama pada data DNA yaitu “CUAACG”. Konversi DNA ke protein dilakukan 3 DNA ke 1 protein. “CUA” bila melihat tabel konversi DNA protein memiliki nilai protein “L” dengan nilai ambiguitas 4. Nilai ambiguitas didapatkan karena DNA “CUA” berada pada posisi ke 5 dalam tabel di protein “L”. Sehingga nilai ambiguitasnya adalah 4 karena dimulai dari nilai 0. Untuk “ACG” bila dikonversikan menjadi protein “T” dengan nilai ambiguitas 3.

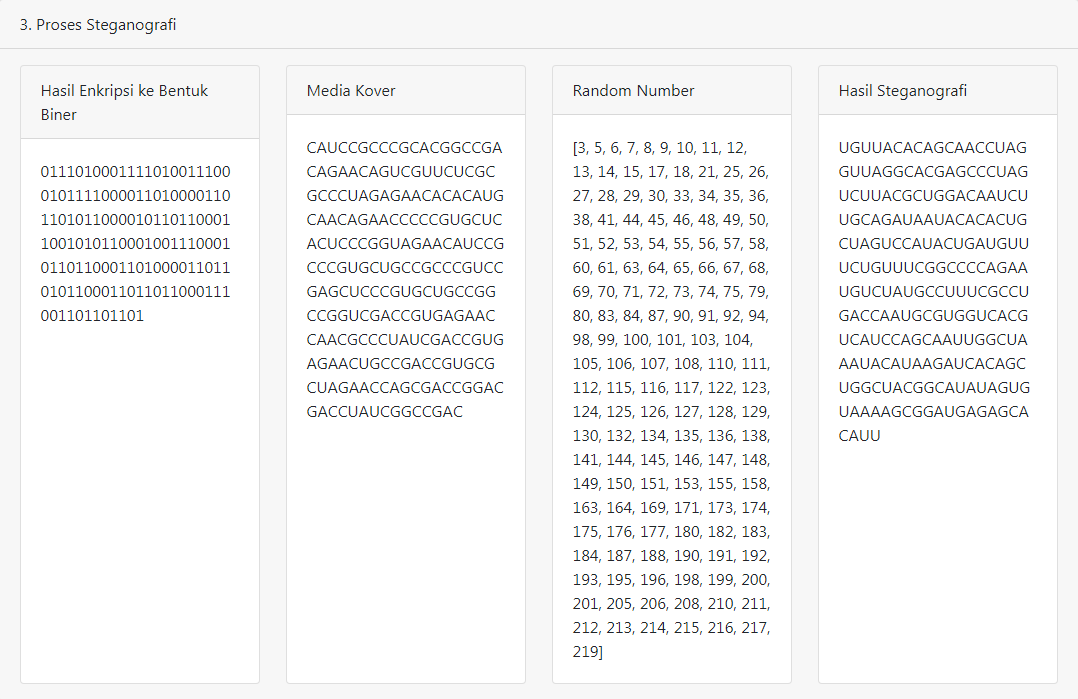
1. **Enkripsi Playfair**



Setelah pesan rahasia/*plaintext* dilakukan *pre-processing*, tahapan selanjutnya yaitu proses enkripsi terhadap *plaintext* dengan menggunakan *playfair cipher*. Tahapan enkripsi *playfair* sama seperti standar enkripsi *playfair* pada umumnya yaitu dengan langkah membangkitkan tabel kunci dan melakukan operasi penggeseran *plaintext* satu-persatu abjadnya dengan menggunakan tabel kunci dan operasinya sesuai dengan algoritma *playfair cipher*.

Operasi *playfair cipher* dilakukan setiap 2 karakter pada *plaintext*. Sebagai contoh dilakukan enkripsi untuk 2 karakter pertama yaitu “lt”. Karakter “l” berada pada posisi kolom 5 dan baris 3, karakter “t” berada pada kolom 5 dan baris 4. Karena posisi kedua karakter berada pada kolom yang sama, maka posisi karakter baru yaitu posisi baris ditambah 1. Sehingga karakter “l” menjadi “t” dan karakter “t” menjadi “z”. Karakter selanjutnya yaitu “p” dan “y”. Posisi “p” ada di kolom 3 dan baris ke 4. Posisi “y” berada di kolom 4 baris 5. Karakter “p” diganti karakter dengan baris sama yaitu 4 dan posisi kolom dari karakter “y” yaitu 4 yang berarti karakter “q”. Karakter “y” diganti dengan yang berposisi baris sama yaitu 4 dan posisi kolom dari “p” yaitu 3. Sehingga, karakter “y” diganti dengan karakter “x”. Menghasilkan dari “py” ke “qx”. Dan seterusnya terhadap semua *plaintext* yang ada.

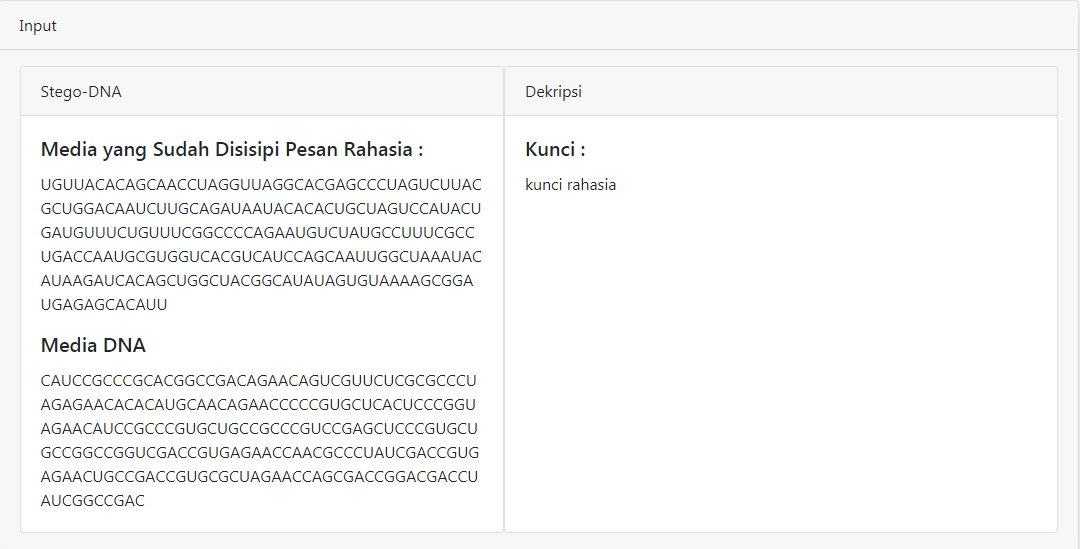
1. **Steganografi Subtitusi**



Proses ini bertujuan untuk menyisipkan data, yang pada kali ini yaitu hasil enkripsi *playfair cipher* di langkah sebelumnya, ke dalam sebuah media kover yang berupa data DNA. Teknik yang digunakan pada steganografi kali ini yaitu teknik subtitusi DNA. Teknik subtitusi DNA bekerja dengan melakukan subtitusi pada media kover dengan aturan tertentu

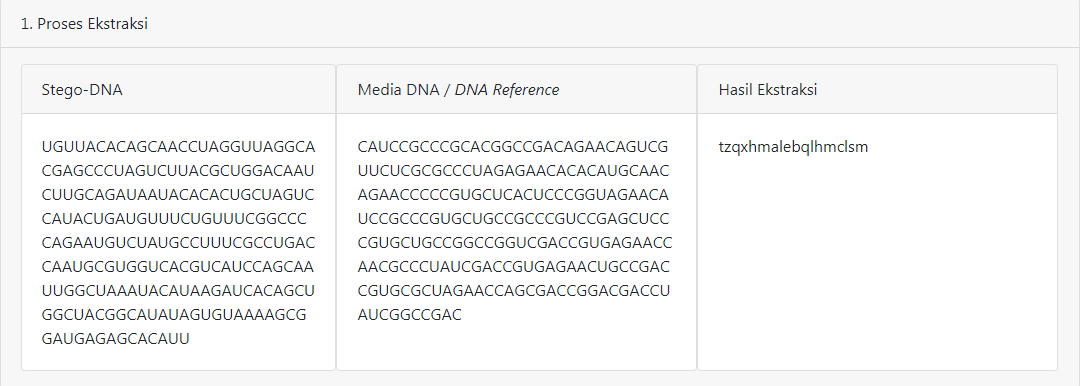
Hasil dari algoritma yang diusulkan berupa data DNA yang dikirimkan dari pengirim (*sender*) ke penerima pesan rahasia (*receiver*). Kunci dari algoritma ini adalah kunci rahasia dan nilai ambiguitas protein.

1. **Input Decode**



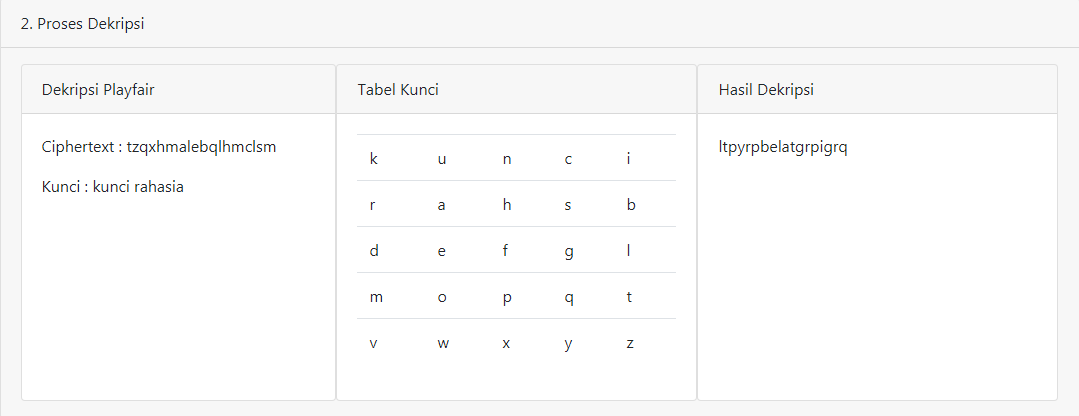
Pada proses *decoding*, data yang diperlukan yaitu media yang sudah disisipi pesan rahasia, media kover DNA, kunci rahasia, dan nilai ambiguitas.

1. **Ekstraksi**



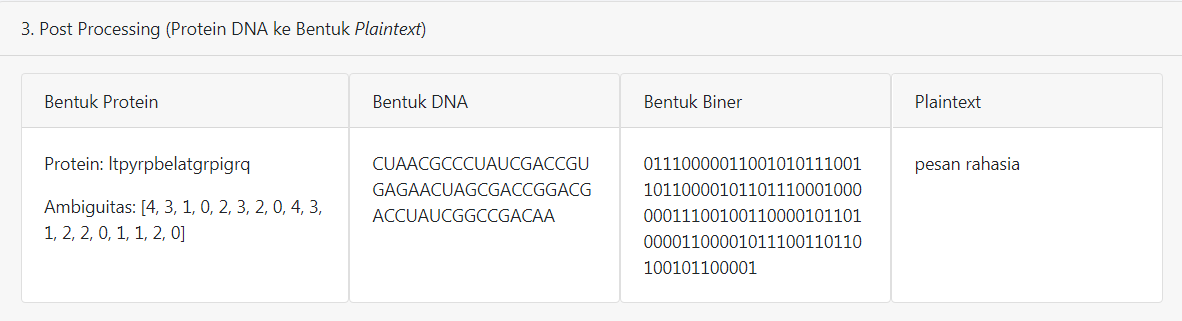
Proses ini bertujuan untuk melakukan ekstraksi pesan rahasia dari stego-DNA atau media yang sudah disisipi pesan rahasia. Proses ekstraksi membandingkan satu per satu dari stego-DNA terhadap media kover DNA sebelum disisipi. Cara kerjanya yaitu bila stego-DNA disimbolkan dengan S’, S adalah media kover dan M adalah pesan rahasia yang diekstrak. Maka bila S’i = Si nilai Mj adalah 0, bila S’­i = C(Si) nilai Mj adalah 1. Simbol i dan j sebagai posisi karakter.

1. **Dekripsi**



Proses deskripsi *playfair cipher* hamper sama dengan proses enkripsinya, hanya aturan bila baris atau kolom ataupun keduanya sama berkebalikan dengan aturan pada enkripsi. Operasi yang dilakukan yaitu dengan melakukan pergeseran karakter terhadap tabel kunci. Misal dicontohkan karakter pertama dan kedua dalam *ciphertext* yaitu “tz”akan dideskripsikan dengan tabel kunci. Posisi huruf “t” berada pada kolom 5 dan baris 4. Posisi huruf “z” berada pada kolom 5 baris 5. Dikarenakan posisi kolom sama, maka yang dilakukan yaitu setiap huruf diganti dengan posisi huruf di atasnya karena pada enkripsi aturan yang berlaku adalah mengganti dengan huruf dibawahnya. Oleh karena itu, huruf “t” diganti huruf “l” dan karakter “z” diganti huruf “t”. Begitupula dengan seterusnya.

1. **Decode Protein**



Langkah terakhir dalam mendapatkan pesan rahasia yaitu melakukan konversi dari DNA ke dalam bentuk digital/biner. Setelah itu, karena pesan berupa kalimat, maka perlu dikembalikan ke dalam bentuknya yang pada kali ini menggunakan format ASCII. Sehingga didapatkan kalimat pesan rahasia yaitu “pesan rahasia”.

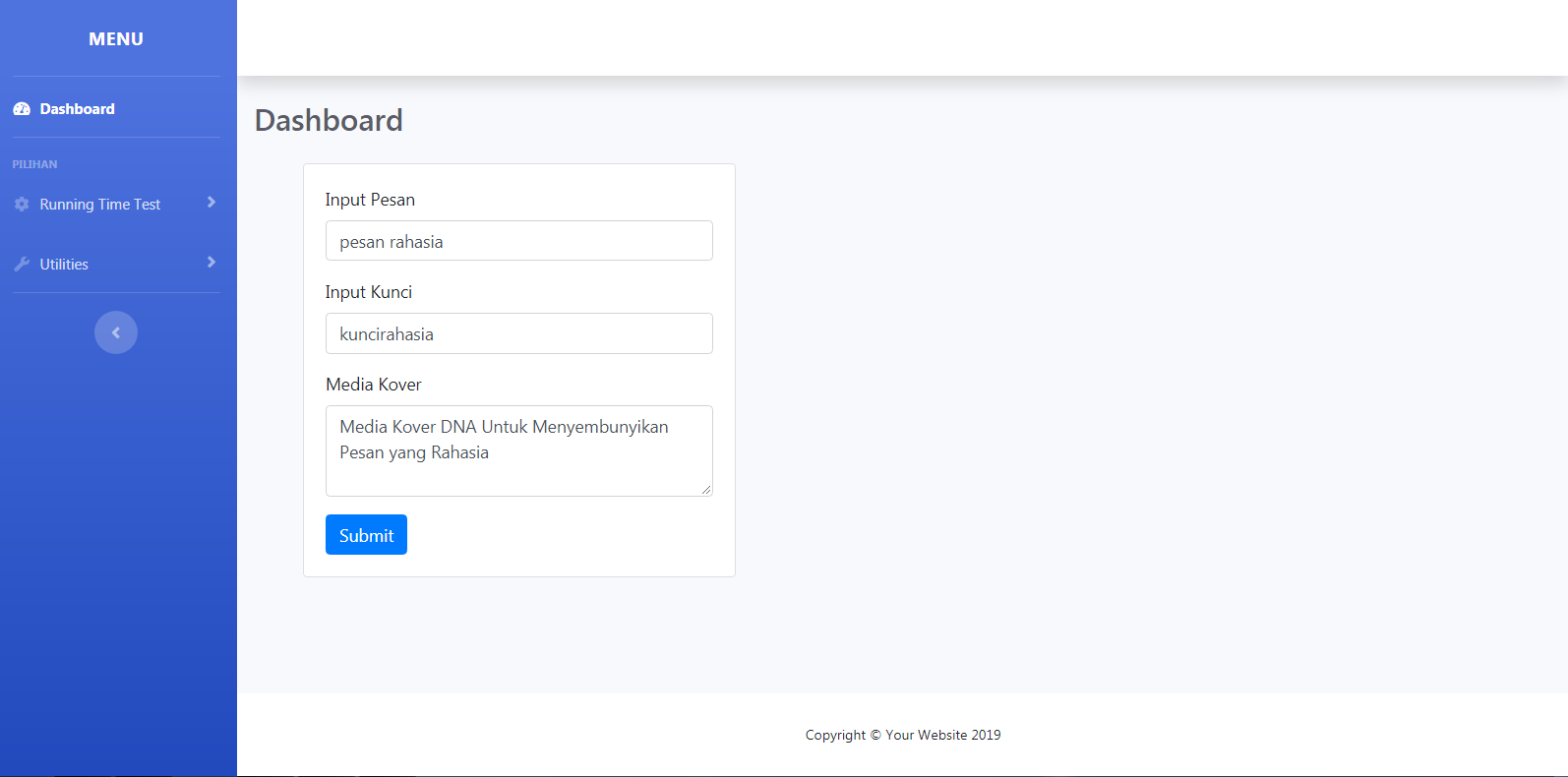
Setelah serangkaian operasi di atas dilakukan, akan didapatkan *plaintext* atau pesan rahasia yang ingin dikirimkan oleh pengirim ke penerima. Pesan rahasia diharapkan aman dan tidak diketahui oleh selain dari si penerima yang semestinya. Dan kapasitas serta kecepatan dari algoritma perlu diperhatikan. Sehingga, tahapan selanjutnya yaitu tahap uji coba terhadap algoritma yang diusulkan. Hasil uji coba akan menentukan nilai performa algoritma yang diusulkan dan juga sebagai pembanding terhadap algoritma lainnya yang serupa.

## Uji Coba

Pada tahapan ini, rangkaian uji coba yang dilakukan adalah uji kecepatan, uji kapasitas, dan uji ketahanan. Setiap uji coba akan dilakukan uji bandingan terhadap algoritma lainnya yang serupa, sehingga akan didapatkan gambaran mengenai posisi performa algoritma yang diusulkan

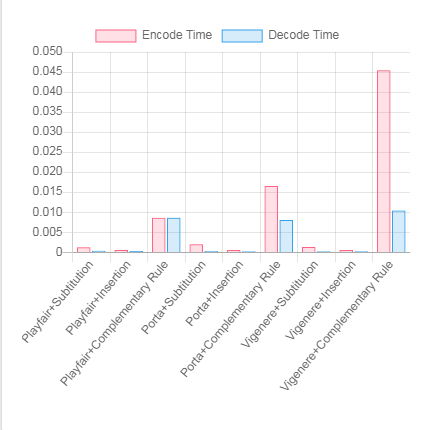
1. **Kecepatan**

Uji kocepatan bertujuan untuk menguji performa algoritma yang diusulkan dengan algoritma lain yang serupa dari sisi kinerja kecepatannya. Uji coba kecepatan dilakukan dengan input *plaintext* “pesan rahasia”, input kunci “kuncirahasia”, dan media kover “Media Kover DNA Untuk Menyembunyikan Pesan yang Rahasia”



Yang diuji dalam percobaan ini adalah waktu *encode* dan waktu *decode*. Waktu *encode* adalah waktu yang dihabiskan dalam proses dari input awal, konversi DNA, enkripsi, sampai *embedding* steganografi. Sedangkan, waktu *decode* adalah waktu yang ditempuh dari proses ekstraksi steganografi, dekripsi, sampai konversi kembali ke bentuk data semula saat diinputkan.

Grafik hasil uji coba *encode* adalah sebagai berikut :



Dari grafik terlihat perbandingan kecepatan diantara metode yang diujikan. Metode yang diusulkan yaitu *playfair cipher* dengan subtitusi DNA berada di posisi paling kiri. Sedangkan metode selainnya merupakan kombinasi dari metode lain yang serupa. Terlihat perbedaan kecepatan diantara metode hanyak sedikit kecuali pada 3 metode yaitu “Playfair+Complementary Rule”, “Porta+Complementary Rule”, dan “Vigenere+Complementary Rule”. Diagram tersebut dijabarkan ke dalam tabel berikut :

|  |  |  |
| --- | --- | --- |
| Algoritma | Waktu Encoding | Waktu Decoding |
| Playfair+Subtitution | 0.001116377999977658 seconds | 0.0003934119999939867 seconds |
| Playfair+Insertion | 0.000614065000007713 seconds | 0.00043560400001751987 seconds |
| Playfair+Complementary Rule | 0.011416933999981893 seconds | 0.009132292000003872 seconds |
| Porta+Subtitution | 0.001507508999992524 seconds | 0.00020012699999938377 seconds |
| Porta+Insertion | 0.0005256900000176756 seconds | 0.0003649040000084369 seconds |
| Porta+Complementary Rule | 0.02175911099999439 seconds | 0.008871728000002577 seconds |
| Vigenere+Subtitution | 0.0012058929999909651 seconds | 0.00017903099998761718 seconds |
| Vigenere+Insertion | 0.000846690999992461 seconds | 0.0003728860000080658 seconds |
| Vigenere+Complementary Rule | 0.00857524299999568 seconds | 0.009026812000001883 seconds |

Dari tabel diatas, dapat dilihat perbedaan detail dari kecepatan masing-masing algoritma. Algoritma tercepat ada pada metode “Porta+Insertion” dalam kecepatan *encoding* tetapi tidak begitu signifikan perbedaan kecepatannya. Kecepatan *decoding* tercepat ada di teknik “Vigenere+Subtitution”. Perbedaannya juga tidak begitu signifikan. Dari tabel diatas, terdapat hasil analisa yaitu algoritma yang menggunakan teknik steganografi “Complementary Rule” mendapatkan waktu *encode* dan *decode* yang jauh lebih lama dari algoritma lain.

1. Kapasitas

* Kriptografi

|  |  |  |
| --- | --- | --- |
| Algoritma | Kapasitas Kunci | Kapasitas Plaintext |
| Playfair-DNA | 25 karakter | ∞ |
| Porta-DNA | <Plaintext | ∞ |
| Vigenere-DNA | <Plaintext | ∞ |
| AES | 256 bit | 256 bit |

Kapasitas kunci sangat bergantung pada algoritma enkripsi yang menyertai. Pada algoritma *Playfair*, jumlah kunci bergantung pada ukuran tabel kunci yang digunakan. Bila tabel kunci yang digunakan berukuran 5x5, maka ukuran kunci maksimalnya yaitu 25 karakter.

Untuk algortima porta dan vigenere, jumlah kunci bergantung pada panjang pesan karena termasuk dalam jenis *Shift Cipher*. Cara kerja *Shift Cipher* yaitu dengan menggeser posisi karakter sejumlah kunci.

Pada algoritma AES yang merupakan turunan algoritma *Rijndael* , termasuk ke dalam tipe enkripsi blok atau *Block Cipher*. Semua operasinya sangat bergantung pada blok yang sudah ditentukan ukurannya. Sehingga, ukuran kuncinya mengikuti ukuran dari bloknya yang pada AES maksimal ukuran blok yaitu 256 bit.

* Steganografi

|  |  |
| --- | --- |
| Algoritma | Kapasitas Sembunyi |
| Subtitution | < Media Penyembunyian |
| Insertion | ∞ |
| Complementary Pair | ∞ |

Untuk steganografi, kapasitas sembunyi algoritma *Subtitution* yaitu sebanyak panjang dari media sembunyi karena konsep utama dari algoritma ini yaitu mengganti karakter pada media penyembunyian dengan karakter yang akan disembunyikan. Dan pada algoritma *Insertion* dan *Complementary Pair*, kapasitas sembunyi tidak memiliki batas karena konsep utama dari algoritma tersebut yaitu dengan menyisipkan pesan rahasia ke dalam media sembunyi. Terlepas dari itu, kemampuan untuk menyembunyikan pesan rahasia menjadi masalah lain karena bila terlalu banyak yang disembunyikan, maka efektifitas media penyembunyian akan jauh berbeda dari kondisi awalnya.

Bila dibuat tabel kapasitas untuk kombinasi algoritma akan seperti berikut :

|  |  |  |
| --- | --- | --- |
| Algoritma | Kapasitas Kunci | Kapasitas Pesan Rahasia |
| Playfair+Subtitution | 25 karakter | < Media Penyembunyian |
| Playfair+Insertion | 25 karakter | ∞ |
| Playfair+Complementary Pair | 25 karakter | ∞ |
| Porta+Subtitution | <Plaintext | < Media Penyembunyian |
| Porta+Insertion | <Plaintext |  |
| Porta+Complementary Pair | <Plaintext | ∞ |
| Vigenere+Subtitution | <Plaintext | < Media Penyembunyian |
| Vigenere+Insertion | <Plaintext | ∞ |
| Vigenere+Complementary Pair | <Plaintext | ∞ |

1. Ketahanan

* Enkripsi

|  |  |
| --- | --- |
| Algoritma | Kompleksitas |
| Playfair-DNA | 25! |
| Vigenere-DNA | n\*26 |
| Porta-DNA | n\*13 |
| AES | 2256 |

* Vigenere

Vigenere cipher sulit dipecahkan bila pesan dan kuncinya sangat panjang. Keamanan dari vigenere cipher bergantung pada keduanya karena cara kerja vigenere cipher yaitu dengan mengoperasikan masing-masing karakter pada plaintext dan kuncinya. Selain itu, panjang plaintext dan kunci pada algoritma vigenere tidak terbatas yang menyebabkan kompleksitasnya juga tergantung pada panjang keduanya. Akan tetapi, vigenere memiliki kelemahan. Kelemahan vigenere cipher yaitu pada serangan *plaintext-ciphertext known*. Sangat mudah untuk mencari kunci rahasianya bila telah diketahui plaintext dan ciphertextnya, karena operasi vigenere hanya menambahkan antara plaintext dan kunci. Kompleksitas *vigenere cipher* yaitu n\*26, n didapatkan dari panjang *plaintext* sedangkan 26 adalah jumlah alfabet.

Contoh :

Ciphertext = (Plaintext+Kunci) mod 25

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Plaintext | P | E | S | A | N | R | A | H | A | S | I | A |
| Kunci | K | U | N | C | I | R | A | H | A | S | I | A |
| Ciphertext | Z | Y | F | C | V | I | A | O | A | K | Q | A |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| Plaintext | 15 | 4 | 18 | 0 | 13 | 17 | 0 | 7 | 0 | 18 | 8 | 0 |
| Kunci | 10 | 20 | 13 | 2 | 8 | 17 | 0 | 7 | 0 | 18 | 8 | 0 |
| Ciphertext | 25 | 24 | 5 | 2 | 21 | 8 | 0 | 14 | 0 | 10 | 16 | 0 |

Plaintext = (|Ciphertext-Kunci|) mod-125

Plaintext = 25-(|Ciphertext-Kunci|)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ciphertext | 25 | 24 | 5 | 2 | 21 | 8 | 0 | 14 | 0 | 10 | 16 | 0 |
| Kunci | 10 | 20 | 13 | 2 | 8 | 17 | 0 | 7 | 0 | 18 | 8 | 0 |
| Plaintext | 15 | 4 | 18 | 0 | 13 | 17 | 0 | 7 | 0 | 18 | 8 | 0 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| Ciphertext | Z | Y | F | C | V | I | A | O | A | K | Q | A |
| Kunci | K | U | N | C | I | R | A | H | A | S | I | A |
| Plaintext | P | E | S | A | N | R | A | H | A | S | I | A |

* Porta

Untuk algoritma Porta hampir sama dengan algoritma *Vigenere* karena sama-sama merupakan kategori *Shift Cipher*. Tetapi *porta cipher* menggunakan setengah alfabet, sehingga nilai kompleksitasnya menjadi 13 untuk setiap karakter *plaintext*. Kompleksitasnya adalah n\*13. Dengan n adalah panjang *plaintext*.

* Playfair

Kemungkinan untuk mencoba setiap kemungkinan tabel yaitu 25! Atau sekitar 1.551121e+25. Hasil tesebut didapatkan bila menggunakan aturan umum playfair menggunakan tabel dengan ukuran 5x5.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | C | D | E |
| F | G | H | I | J |
| K | L | M | N | O |
| P | Q | R | S | T |
| U | V/W | X | Y | Z |

Akan tetapi, kompleksitas bisa ditingkatkan bila menambah panjang dari tabel kunci.

* AES

Kemungkinan untuk mencoba setiap kunci pada AES yaitu 2256 karena ukuran kunci pada AES maksimal 256 bit

* Steganografi

Steganografi mudah dicari pesan rahasianya bila penyerang mengetahui media yang digunakan untuk menyembunyikan pesan. Yang menjadi pengaman bila penyerang mengetahui media penyembunyian adalah kunci penyembunyiannya. Kunci penyembunyian menentukan posisi/letak pesan rahasia pada media sembunyi sehingga penyerang tidak bisa merangkai pesan rahasia yang kemudian dari bentuk DNA dikonversi ke dalam bentuk aslinya.

1. Perubahan Media Kover

Pada uji ini, media kover sebelum dilakukan operasi penyisipan data dibandingkan dengan media kover yang sudah disisipkan data. Dicari nilai kemiripan antar keduanya. Hasilnya adalah sebagai berikut :

* Teknik Playfair+Subtitution : 0.004545454545454545
* Teknik Playfair+Insertion : 0.0
* Teknik Playfair+Complementary Rule : 0.006498781478472786
* Teknik Porta+Subtitution : 0.004545454545454545
* Teknik Porta+Insertion : 0.0
* Teknik Porta+Complementary Rule : 0.006498781478472786
* Teknik Vigenere+Subtitution : 0.0
* Teknik Vigenere+Insertion : 0.0
* Teknik Vigenere+Complementary Rule : 0.006498781478472786

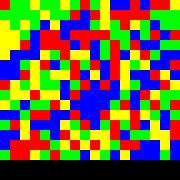
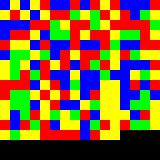
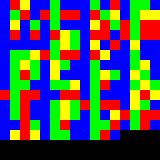
Dari nilai di atas, kemiripan *encode* dengan media kover asli sangatlah sedikit. Bahkan, ada yang bernilai 0 atau tidak mirip sama sekali. Sisanya, nilai kemiripannya dibawah 0,01 atau dibawah 1 persen.

Bila dibuat gambar ilustrasi pola sekuen DNA dari media DNA tersebut adalah seperti berikut :

Original

Playfair + Subtitution

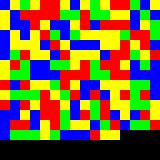
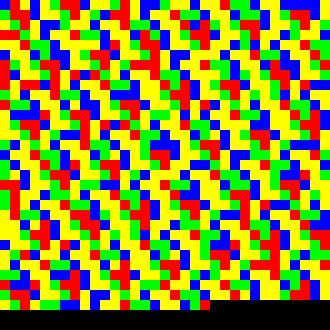
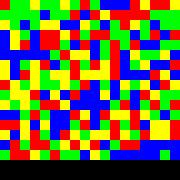
Playfair + Insertion



Playfair + Complementary Pair

Porta + Subtitution

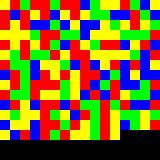
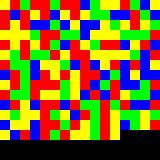
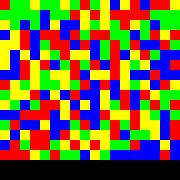
Porta + Subtitution



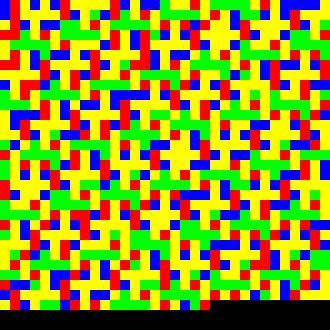
Porta + Complementary Pair

Vigenere + Subtitution

Vigenere + Insertion



Vigenere + Complementary Pair



Figure

Figure

Gambar ilustrasi di atas disusun dengan aturan yaitu piksel merah untuk DNA “A”, hijau untuk “G”, biru untuk “C”, dan kuning untuk “U”. Terlihat dari perbandingan gambar tesebut bahwa pola masing-masing gambar jauh berbeda dengan media kover asli. Hal ini dibuktikan dari tabel kemiripan sebelumnya bahwa kemiripan dengan media kover asli paling besar 0,0064 atau sekitar 0,6 persen. Sehingga , dapat ditarik kesimpulan bahwa semua teknik yang diujikan tidak efektif dalam menyembunyikan kehadiran data pada sebuah media kover DNA.

## Analisa

Dari hasil uji coba, dapat dilakukan beberapa analisa. Dari segi kecepatan, hamper semua teknik memiliki kecepatan yang rata-rata kecuali pada teknik yang menggunakan algoritma steganografi *Complementary Pair.* Sehingga dari 3 algoritma steganografi yang digunakan yaitu algoritma *Subtitution*, *Insertion*, dan *Complementary Pair* algoritma *Complementary Pair* menjadi yang memiliki kecepatan terlama jauh dari kedua algoritma lainnya. Sementara, untuk kecepatan teknik enkripsi yang diujikan yaitu *Playfair cipher*, *Vigenere cipher*, dan *Porta cipher* memiliki kecepatan enkripsi yang tidak jauh berbeda.

Selanjutnya, analisa dari hasil uji coba pada kapasitas *plaintext*, kunci, dan kapasitas *hiding* atau kapasitas sembunyi. Kapasitas 3 algoritma kriptografi yang diujikan yaitu *Playfair cipher*, *Vigenere cipher,* dan *Porta cipher* memiliki maksimal tak terhingga. Hal itu disebabkan karena algoritma tersebut bukan merupakan *Block cipher.* Sedangkan pada AES, yang merupakan tipe *Block cipher*, kapasitas tergantung dari besaran blok yang digunakan. Operasi *Block cipher* terjadi antara blok kunci yang dibangkitkan dan dengan block pesan rahasianya. Sehingga, dalam satu kali operasi panjang *plaintext* dan kuncinya maksimal hanya 1 blok yang pada AES ada tipe 128 bit, 192 bit, dan 256 bit. Pada kapasitas kunci, beberapa algoritma bergantung pada ukuran dari *plaintext*. Yaitu pada algoritma *Vigenere cipher* dan *Porta cipher*. Hal itu disebabkan karena kedua algoritma tersebut termasuk kedalam *Shift cipher*, yaitu algoritma kriptografi yang cara kerjanya menggeser posisi pesan dalam tabel alfabetnya. Sehingga, dibutuhkan sejumlah kunci yang sama besarnya dengan besar *plaintext*. Tetapi bila kurang dari *plaintext*, bisa dilakukan perulangan kunci terhadap *plaintext* yang tidak mendapatkan kunci untuk menggeser. Lalu, uji terakhir adalah pengujian terhadap kapasitas *hiding*. Dari 3 algoritma yang digunakan, kapasitas *hiding* pesan tidak memiliki batasan kecuali pada teknik subtitusi. Dua teknik selain subtitusi memiliki inti operasi dengan menyisipkan langsung pesan di antara media kover. Sehingga, jumlah yang disisipkan tidak memiliki batasan panjang. Akan tetapi, semakin banyak data atau pesan yang disisipkan maka visibilitas dari pesan akan lebih mudah terlihat. Untuk teknik yang terakhir yaitu subtitusi, cara kerja algoritma tersebut yaitu mengganti atau subtitusi pada media kover dengan pesan yang ingin disisipkan ke dalam media kover. Sehingga, batas maksimal yang bisa diganti sebanyak besar dari media kover. Teknik ini juga memiliki Batasan seperti 2 teknik sebelumnya, yaitu semakin banyak data yang disubtitusikan terhadap media kover maka semakin mudah terlihat juga perubahan pada media kover.

Uji coba berikutnya yaitu uji ketahanan. Ketahanan berarti kompleksitas algoritma saat mencoba untuk membongkarnya. Pada algoritma enkripsi, besaran kompleksitas bervariasi. Untuk algoritma *Playfair cipher*, kompleksitasnya bernilai 25!. Nilai itu didapatkan dari jumlah kemungkinan pada tabel kunci. Tabel kunci berukuran 5x5 sehingga anggotanya ada 25. Sehingga kemungkinan variasi tabel kunci adalah 25! kali. Sedangkan pada algoritma *Vigenere cipher* dan *Porta cipher*. Kompleksitas sangat bergantung pada panjang kunci. Kompleksitasnya yaitu n\*13 untuk *Porta cipher* dan n\*26 untuk *Vigenere cipher* yang n adalah panjang kunci. 26 didapatkan dari jumlah alfabet yaitu 26 karakter. Sedangkan *Porta cipher* menggunakan setengahnya sehingga bernilai 13. Untuk AES, kompleksitasnya adalah 2256 karena ukuran kuncinya 256 bit. Lalu, pada uji coba kompleksitas algoritma stegaongrafi. Pengaman dari algoritma steganografi DNA adalah media kovernya dan posisi penempatan pesan rahasia. Kompleksitas media kover bila merujuk pada penelitian sebelumnya adalah sejumlah data DNA yang ada. Sedangkan untuk posisi penempatan tergantung pada ukuran pesan rahasia.

Uji yang terakhir adalah tingkat perubahan pada media kover setelah dilakukan penyisipan. Dari semua teknik yang diujikan, nilai kemiripan hasil penyisipan terhadap media kover sangat kecil. Berkisar di antara 0 sampai 0,6 persen. Nilai tersebut tidak tidak ideal bila perubahan media kover ingin tidak diketahui. Oleh karena itu, ketiga algoritma steganografi DNA yang diujikan tidak cocok untuk teknik yang perubahan media kovernya tidak diketahui.

.

# BAB V PENUTUP



## Kesimpulan

Kriptografi dan Steganografi DNA merupakan cabang ilmu baru yang masih perlu banyak pengembangan ke depannya. Kriptografi DNA masih memiliki potensi untuk dikembangkan dikarenakan sifat DNA yang unik.

Pada penilitian ini, diujikan sebuah metode penyembuniyan data dengan menggunakan metode kriptografi *Playfair cipher* dan steganografi Subtitusi DNA. Metode tersebut diujikan dengan teknik yang serupa yaitu metode kriptografi *Vigenere cipher*, *Porta cipher*, dan AES. Sedangkan untuk metode steganografi yang menjadi pembanding adalah metode *Insersertion* dan *Complementary Pair*. Yang diujikan adalah kecepatan, kapasitas, ketahanan, dan kemiripan.

Pada uji kecepatan, algoritma yang diusulkan mendapatkan hasil yang relatif cepat. Yaitu sekitar 0.0011 detik untuk proses *encode*, terpaut sedikit dengan metode tercepat yang diraih metode *Porta* dengan *Insertion* dengan waktu 0,0005 detik.

Pada uji kapasitas, kapasitas kunci metode yang diusulkan yaitu 25 karakter. Angka tersebut lebih kecil dibandingkan metode yang menggunakan enkripsi *Vigenere* dan *Porta* yaitu kapasitas kunci mengikuti panjang *plaintext*. Dari kapasitas panjang pesan rahasia juga lebih kecil dibandingkan dengan metode lain. Kapasitas panjang pesan rahasia dari algoritma yang diusulkan adalah lebih kecil atau sama dengan panjang media kover. Kapasitas tersebut lebih kecil dari metode yang menggunakan algoritma steganografi *Insertion* dan *Complementary Pair* dengan kapasitas pesan rahasia tidak memiliki batasan.

Uji ketahanan menunjukkan kompleksitas metode yang diusulkan cukup tinggi, yaitu 25! atau sekitar 1,55 x 1025. Angka ini lebih tinggi dari metode yang menggunakan metode kriptografi *Vigenere* dan *Porta* dengan nilai masing-masing n\*26 dan n\*13, dengan n adalah panjang kunci. Akan tetapi, kompleksitas metode yang diusulkan masih kalah dengan metode AES dengan kompleksitas 2256 atau sekitar 1,15 \*1077.

Untuk uji kemiripan atau tingkat kemiripan terhadap media kover sebelum dilakukan penyisipan, hasilnya mengecewakan. Nilai *similarity* dari metode yang diusulkan adalah 0,4 persen. Nilai tersebut juga tidak jauh berbeda dengan metode lain yang diujikan dengan rentang nilai 0 sampai 0,6 persen. Hasil itu menunjukkan metode tersebut tidak memiliki kemampuan untuk menyembunyikan kehadirannya pada media kover.

## Saran

Untuk meningkatkan kemampuan untuk menyembunyikan kehadiran pesan rahasia pada media kover, disarankan menggunakan media tambahan yang kasat mata seperti gambar, video, atau dokumen pdf. Selain itu, teknik DNA bisa diterapkan ke dalam kriptografi asimetrik seperti RSA, Elgamal, ECC, ataupun metode kriptografi POST quantum seperti NTRU, SIDH, dan McBits.

# DAFTAR PUSTAKA

# LAMPIRAN