MODIFIKASI PEMBANGKIT KUNCI ALGORITMA ELGAMAL DENGAN MENGGUNAKAN ALGORITMA DNA

SKRIPSI

OLEH:
IHDA UMDATUL KHOIROH
NIM. 19610002

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2023

MODIFIKASI PEMBANGKIT KUNCI ALGORITMA ELGAMAL DENGAN MENGGUNAKAN ALGORITMA DNA

SKRIPSI

Diajukan Kepada
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Matematika (S.Mat)

Oleh
Ihda Umdatul Khoiroh
NIM. 19610002

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2023

MODIFIKASI PEMBANGKIT KUNCI ALGORITMA ELGAMAL DENGAN MENGGUNAKAN ALGORITMA DNA

SKRIPSI

Oleh
Ihda Umdatul Khoiroh
NIM. 19610002

Telah Disetujui Untuk Diuji
Malang, 23 Juni 2023

Dosen Pembimbing I	Dosen Pembimbing II
Prof. Dr. H. Turmudi, M.Si., Ph.D.	Dr. H. Imam Sujarwo, M.Pd.
NIP. 19571005 198203 1 006	NIP. 19630502 198703 1 005

Mengetahui,
Ketua Program Studi Matematika

Dr. Elly Susanti, M.Sc.
NIP. 19741129 200012 2 005

MODIFIKASI PEMBANGKIT KUNCI ALGORITMA ELGAMAL DENGAN MENGGUNAKAN ALGORITMA DNA

SKRIPSI

Oleh
Ihda Umdatul Khoiroh
NIM. 19610002

Telah Dipertahankan di Depan Dewan Penguji
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S.Mat)
Tanggal 26 Juni 2023

Ketua Penguji	: Dr. Elly Susanti, M.Sc.	.....
Anggota Penguji I	: Muhammad Khudzaifah, M.Si.	.....
Anggota Penguji II	: Prof. Dr. H. Turmudi, M.Si., Ph.D.	.....
Anggota Penguji III	: Dr. H. Imam Sujarwo, M.Pd.	.....

Mengetahui,
Ketua Program Studi Matematika

Dr. Elly Susanti, M.Sc.
NIP. 19741129 200012 2 005

PERNYATAAN KEASLIAN TULISAN

Saya yang bertandatangan di bawah ini:

Nama : Ihda Umdatul Khoiroh

NIM : 19610002

Program Studi : Matematika

Fakultas : Sains dan Teknologi

Judul Skripsi : Modifikasi Pembangkit Kunci Algoritma Elgamal dengan Menggunakan Algoritma DNA

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan dan pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan atau daftar rujukan. Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 26 Juni 2023

Yang membuat pernyataan,

Ihda Umdatul Khoiroh

NIM. 19610002

MOTO

"Where There is A Will, There is A Way"

PERSEMBAHAN

Skripsi ini penulis persembahkan kepada:
Kedua orangtua tercinta Bapak Kusaini dan Ibu Mujayanah yang senantiasa mendoakan, mengerahkan waktu dan tenaganya untuk mendukung cita-cita penulis dan juga tak henti memberi semangat serta nasihat selama proses pengerjaan skripsi berlangsung. Kakek dan nenek penulis H.Sentot Syafi'i dan Hj.Tinah yang telah memberi dukungan kepada penulis selama proses perkuliahan berlangsung. Adik tersayang Za'imatul Qorir El-Farosyita yang telah mendoakan dan juga memberi semangat kepada penulis.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarokatuh

Segala puji dan syukur penulis panjatkan kehadirat Allah SWT yang telah memberikan berkah, rahmat serta hidayah-Nya sehingga penulis masih diberikan nikmat kesehatan, kesabaran, dan kesempatan dalam penulisan skripsi yang berjudul "Modifikasi Pembangkit Kunci Algoritma Elgamal dengan Menggunakan Algoritma DNA" dapat dilakukan dengan baik.

Sholawat serta salam selalu tercurahkan kepada Nabi Muhammad SAW yang telah menjadi panutan penulis agar menjadi pribadi yang cerdas dan berakhlak. Pada kesempatan kali ini, penulis ingin mengucapkan terima kasih kepada semua pihak yang telah mendukung, membantu, serta memotivasi penulis dalam penyusunan proposal skripsi ini, yakni kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Sri Harini, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Elly Susanti, S.Pd., M.Sc., selaku ketua Program Studi Matematika, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Prof. Dr. H. Turmudi, M.Si., Ph.D., selaku dosen pembimbing I yang telah memberi arahan, nasihat, serta motivasi kepada penulis sehingga dapat menyelesaikan proposal skripsi ini dengan baik.
5. Dr. H. Imam Sujarwo, M.Pd., selaku dosen pembimbing II yang telah memberikan bimbingan kepada penulis.
6. Seluruh dosen Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang yang telah memberikan ilmu selama dibangku perkuliahan.
7. Kedua orang tua penulis Bapak Kusaini dan Ibu Mujayanah yang senantiasa mendoakan dan mendukung penulis sehingga dapat menyelesaikan tugas akhir.
8. Kakek dan Nenek penulis H. Sentot Syafi'i dan Hj. Tinah yang mendukung cita-cita penulis.

9. Adik penulis Za’imatul Qorir El-Farosyita yang telah memberi semangat kepada penulis.
10. Seluruh teman-teman mahasiswa program studi matematika angkatan 2019, serta teman-teman yang telah menjadi pendengar keluh kesah penulis, membersamai proses pengerjaan tugas akhir serta telah memberikan dukungan dan semangat kepada penulis.

Semoga Allah melimpahkan pahala berlipat ganda bagi mereka. Penulis memohon maaf apabila skripsi ini masih terdapat kekurangan. Penulis berharap semoga skripsi ini dapat bermanfaat dan menambah wawasan bagi para pembaca.

Malang, 26 Juni 2023

Penulis

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR LAMPIRAN

ABSTRAK

Khoiroh, Ihda Umdatul. 2023. **Modifikasi Pembangkit Kunci Algoritma Elgamal dengan Menggunakan Algoritma DNA.** Skripsi. Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. H. Turmudi, M.Si., Ph.D. (II) Dr. H. Imam Sujarwo, M.Pd.

Kata Kunci: Algoritma Elgamal, Algoritma DNA, Modifikasi Pembangkit Kunci

Keamanan pesan rahasia merupakan hal penting yang harus dijaga agar informasi yang ada didalamnya tidak diketahui publik. Dalam penelitian ini digunakan algoritma DNA dalam proses pembangkitan kunci dan algoritma Elgamal untuk menyamarkan pesan sehingga pesan akan sulit dipahami oleh orang-orang yang tidak mengetahui kunci yang digunakan. Tujuan dari penelitian ini adalah untuk meningkatkan keamanan *plaintext* dengan memodifikasi proses pembangkitan kunci, sehingga pihak lain akan kesulitan untuk memecahkan kunci tersebut. Hasil dari penelitian ini adalah kunci privat dan kunci publik berupa akar primitif yang telah modifikasi. Kemudian dilakukan dekripsi pada kunci publik untuk melakukan proses enkripsi. Enkripsi dengan algoritma Elgamal menghasilkan blok-blok *ciphertext* berupa angka yang lebih panjang daripada *plaintext*. Untuk mengembalikan *plaintext* dilakukan dekripsi *ciphertext* dengan algoritma Elgamal menggunakan kunci privat yang telah dimodifikasi sehingga menghasilkan *plaintext* seperti semula.

ABSTRACT

Khoiroh, Ihda Umdatul. 2023. **Modification of Key Generator Elgamal Algorithm Using DNA Algorithm.** Thesis. Mathematics Study Program, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Advisors: (I) Prof. Dr. H. Turmudi, M.Si., Ph.D. (II) Dr. H. Imam Sujarwo, M.Pd.

Keywords: Elgamal Algorithm, DNA Algorithm, Modification of Key Generator

The security of secret messages is an important aspect that must be safeguarded to prevent public access to the information therein. In this research, the DNA algorithms are used in the key generation process, and the Elgamal algorithm is employed to obscure the message, making it difficult for individuals without knowledge of the key to understand the message. This research aims to enhance plaintext security by modifying the key generation process, thus making it challenging for others to decipher the key. The results of this study include the modified private key and the public key in the form of primitive roots. Subsequently, the public key is decrypted to initiate the encryption process. Encryption using the Elgamal algorithm produces ciphertext blocks with longer numbers than the plaintext. In order to recover the plaintext, the ciphertext is decrypted using the modified private key in the Elgamal algorithm, resulting in the original plaintext.

مستخلص البحث

الخيرة، إحدى عمدة.٢٠٢٣. **تعديل مولد الخوارزميات الرئيسية الجمل باستخدام الخوارزمية *DNA*.** البحث الجامعى. قسم الرياضيات،كلية العلوم والتكنولوجيا، جامعة مولان مالك إبراهيم الإسلامية الحكومية مالانج. المشرف: (١) البروفيسور، الدكتور، ترمودي، الماجستير، الحاج.(٢) الدكتور إمام سوجاروو، الماجستير، الحاج.

كلمات البحث: خوارزمية الجمل، خوارزمية *DNA* ، تعديل مولدالمفاتيح

الحفاظ على أمان الرسائل السرية هو أمر مهم يجب الحرص عليه لكي لا تصبح معلوماتها معروفة للجمهور. في هذه الدراسة، تم استخدام خوارزمية *DNA* في عملية توليد المفتاح وخوارزمية الجمل لإخفاء الرسالة بحيث تصعب فهم الرسالة بالنسبة للأشخاص الذين لا يعرفون المفتاح المستخدم.والهدف من هذه الدراسة هو تعزيز أمان النص العادي عن طريق تعديل عملية توليد المفتاح، بحيث يصعب على الآخرين كسر هذا المفتاح. ونتائج هذه الدراسة تتضمن المفتاح الخاصَ والمفتاحَ العامَ على شكل جذور بدائية تم تعديلها. ثم يتم فك تشفير المفتاح العام لإجراء عملية التشفير. تعمل غملية التشفير باستخدام خوارزمية الجمل على إنتاج كتل من النص المشفر على شكل أرقام أطول من النص العادي. لاستعادة النص العادي، يتم فك تشفير النص المشفر باستخدام خوارزمية الجمل واستخدام المفتاح الخاص المعدل للحصول على النص العادي كما كان.

BAB I
PENDAHULUAN

1.1 Latar Belakang

Keamanan informasi sudah menjadi masalah sejak zaman dahulu hingga saat ini. Pada zaman modern, informasi merupakan hal yang harus dipastikan keamanannya karena dengan kemajuan teknologi yang ada suatu informasi bisa didapatkan dengan mudah jika tingkat keamananya rendah. Informasi yang bersifat rahasia dan sensitif perlu dijaga keamanannya agar tidak dapat diakses oleh pihak lain yang tidak berhubungan atas informasi tersebut (Munir, 2019). Salah satu kejahatan yang diakibatkan oleh kurangnya tingkat keamanan informasi adalah penyadapan. Kejahatan seperti ini sering terjadi pada informasi yang didistribusikan melalui internet, sedangkan internet telah menjadi media dalam pendistribusian informasi dewasa ini, karena itu tingkat kejahatan seperti penyadapan akan semakin tinggi jika tidak diimbangi dengan tingkat keamanan yang tinggi pula. Salah satu contoh kasus penyadapan adalah pembobolan informasi mengenai data nasabah bank, pencurian dokumen negara, dan juga penyadapan surat-surat penting milik negara yang dilakukan oleh pihak-pihak yang tidak bertanggungjawab. Hal ini menjadikan peringatan bagi kita bahwa keamanan informasi pada saat ini sudah menjadi suatu kebutuhan (Munir, 2019). Oleh karena itu dibutuhkan sebuah alat untuk memperkuat tingkat keamanan terhadap informasi-informasi tersebut, salah satunya dengan memanfaatkan ilmu kriptografi.

Kriptografi merupakan salah satu cabang ilmu matematika yang dapat digunakan untuk meningkatkan keamanan informasi. Kriptografi merupakan

solusi yang tepat untuk mengatasi masalah keamanan (Satir & Kendirli, 2022). Sebuah ilmu dan seni untuk menjaga kerahasiaan sebuah pesan dengan cara mengubah pesan kedalam bentuk sandi hingga maknanya tidak dapat dipahami merupakan fungsi dari kriptografi (Munir, 2019). Penyandian pesan dalam kriptografi disebut dengan enkripsi, sedangkan proses mengembalikan pesan yang sudah disandi menjadi pesan asli disebut dekripsi. Kriptografi membentuk sebuah sistem yang didalamnya berisi himpunan yang terdiri dari algoritma enkripsi, algoritma dekripsi, ruang kunci, dan seluruh *plaintext* dan *ciphertext* yang terdapat di dalamnya (Munir, 2019). Kriptografi memiliki berbagai macam algoritma untuk menyandikan pesan, dalam penelitian ini digunakan algoritma DNA dan algoritma Elgamal.

Deoxyrbose Nucleic Acid (DNA) dalam ilmu biologi merupakan salah satu molekul asam nukleat, dimana asam nukleat adalah salah satu senyawa polimer utama yang berada didalam sel. Dalam DNA tersimpan informasi genetik spesifik yang dimiliki oleh individu dan spesies-spesies tertentu yang kemudian diwariskan ke generasi berikutnya (Gaffar, 2007). Pada awalnya hubungan kriptografi dan biologi molekuler memang tidak relevan, namun dengan adanya penelitian yang mendalami masalah terkait bioteknologi modern dan komputasi DNA, maka dua ilmu yang berbeda ini saling bekerjasama (Mahesa et al., 2019).

DNA dalam kriptografi merupakan pembawa sekaligus pengguna teknik biologi modern sebagai alat aplikasi. Algoritma DNA merupakan hal baru dalam dunia kriptografi. Algoritma DNA lahir setelah adanya penelitian dalam bidang komputasi DNA oleh Adlemen. DNA berbentuk heliks dengan panjang 2 untai nukleotida (Ahmed & Mohammed, 2017). Pada setiap DNA terdapat 4 macam

nukleotida yaitu *Adenine* (A), *Guanine* (G), *Cytosin* (C), dan *Thymine* (T). Teknik pada kriptografi DNA digunakan untuk mengintegrasikan operator DNA dan penyandian DNA kedalam struktur jaringan feistel (Satir & Kendirli, 2022).

Pada penelitian Ahmed dan Mohammed (2017) yang berjudul *"Developing a New Hybrid Cipher Algorithm using DNA and RC4"* menjelaskan cara penyembunyian data yang aman dan memiliki kompleksitas tinggi dalam lingkup steganografi. Penelitian dilakukan dengan mempertimbangkan tiga parameter yaitu entropi bersyarat, tes keacakan, serta waktu enkripsi. Hasil penelitian menunjukkan bahwa kinerja keamanan dengan menggunakan *cipher hybrid* menjadi lebih baik dibandingkan dengan hanya menggunakan RC4 saja. Selain itu, pada penelitian ini juga menyebutkan bahwa penggunaan dua algoritma simetris dapat menyembunyikan data dengan aman dan kompleksitas tinggi dalam lingkup steganografi. Kemudian pada penelitian Mahesa dkk (2019) yang berjudul *"Pemanfaatan Metode DNA Kriptografi dalam Meningkatkan Keamanan Citra Digital"* menyebutkan bahwa metode kriptografi DNA yang diterapkan pada aplikasi yang dibuat dengan menggunakan *Delphi 10.3* dalam rangka meningkatkan keamanan terhadap citra digital mampu menyelesaikan enkripsi dan dekripsi terhadap citra digital.

Algoritma kriptografi berikutnya adalah algoritma Elgamal. Algoritma Elgamal termasuk kedalam jenis algoritma kriptografi asimetris, karena mempunyai kunci untuk enkripsi dan dekripsi yang berbeda. Algoritma kunci publik merupakan sebutan lain untuk algoritma Elgamal (Solin & Ramadhani, 2020). Pada penelitian oleh Warnilah (2018) telah disebutkan bahwa algoritma Elgamal merupakan algoritma kriptografi yang cukup sulit untuk dipecahkan,

karena proses pembentukan kunci membutuhkan bilangan prima dan pemecahan masalah dilakukan dengan menggunakan logaritma diskrit. Kesulitan perhitungan logaritma diskrit merupakan titik keamanan pada algoritma Elgamal.

Menjaga kerahasiaan sebuah informasi merupakan hal yang penting, hingga Allah berfirman pada Surat An-Nisa' ayat 58 yang artinya :

"Sesungguhnya Allah menyuruh kamu menyampaikan amanat kepada yang berhak menerimanya, dan (menyuruh kamu) apabila menetapkan hukum di antara manusia supaya kamu menetapkan dengan adil. Sesungguhnya Allah memberi pengajaran yang sebaik-baiknya kepadamu. Sesungguhnya Allah Maha Mendengar lagi Maha Melihat". (QS. An-Nisa'/4:58).

Berdasarkan uraian yang telah disebutkan, maka penulis ingin melanjutkan penelitian dengan judul *"Modifikasi Pembangkit Kunci Algoritma Elgamal dengan Menggunakan Algoritma DNA".*

1.2 Rumusan Masalah

Berdasarkan latar belakamg yang telah dijelaskan, rumusan masalah pada penelitian ini adalah sebagai berikut:

1. Bagaimana proses modifikasi pembangkit kunci pada algoritma Elgamal dengan menggunakan algoritma DNA?
2. Bagaimana proses enkripsi pesan dengan menggunakan algoritma Elgamal?
3. Bagaimana proses dekripsi pesan menggunakan algoritma Elgamal?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah disebutkan, tujuan pada penelitian ini adalah sebagai berikut:

1. Untuk mendeskripsikan proses modifikasi pembangkit kunci pada algoritma Elgamal dengan menggunakan algoritma DNA.
2. Untuk mendeskripsikan proses enkripsi pesan dengan menggunakan algoritma Elgamal.
3. Untuk mendeskripsikan proses dekripsi pesan menggunakan algoritma Elgamal.

1.4 Manfaat Penelitian

Penelitian ini diharapkan memberi manfaat sebagai berikut:

1. Manfaat Teoritis

   Adanya penelitian ini diharapkan dapat digunakan untuk mengembangkan modifikasi kunci pada algoritma Elgamal agar lebih beragam dan tentunya agar memiliki tingkat keamanan yang lebih tinggi. Selain itu penelitian ini merupakan wujud dari penerapan pengetahuan dibidang matematika dalam lingkup kriptografi.

2. Manfaat Praktis
   a. Bagi Penulis
      i. Memperoleh ilmu dan wawasan yang baru ketika mempelajari kriptografi lebih dalam.
      ii. Menerapkan ilmu yang didapat dibangku perkuliahan, terutama dalam bidang kriptografi dan matematika.

b. Bagi Pembaca

i. Mengetahui beberapa algoritma yang terdapat dalam kriptografi.

ii. Memperoleh informasi yang dapat dikembangkan untuk penelitian dalam bidang kriptografi berikutnya.

1.5 Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Konversi bilangan biner dengan kombinasi pengkodean basa DNA dengan mengkonversi A menjadi 00, G menjadi 10, C menjadi 01, dan T menjadi 11.
2. Konversi *plaintext* menggunakan tabel ASCII 256.
3. Modifikasi kunci privat dan kunci publik g pada algoritma Elgamal.

1.6 Definisi Istilah

Berikut istilah-istilah yang digunakan dalam penelitian ini:

Algoritma Elgamal	: Algoritma yang dibuat Taher Elgamal dan merupakan algoritma kriptografi asimetris/kunci publik.
Algoritma DNA	: Algoritma kriptografi simetris yang memanfaatkan urutan DNA untuk menyembunyikan data.
Cipher	: Algoritma kriptografi yang berupa fungsi matematika yang digunakan pada proses enkripsi dan dekripsi.

Plaintext : Pesan asli yang berbentuk teks.

Ciphertext : Pesan berbentuk teks yang sudah disandi.

Enkripsi : Proses mengubah data atau informasi kedalam suatu bentuk yang sulit dikenali yang dibantu dengan penggunaan sebuah algoritma tertentu.

Dekripsi : Proses mengubah data atau informasi dari bentuk yang samar ke bentuk yang semula.

Kunci (*key*) : Parameter yang digunakan pada proses pengkodean pesan (*enciphering*) dan pengartian pesan (*deciphering*).

Modifikasi Pembangkit Kunci : Pengubahan proses pembentukan kunci pada suatu algoritma dari bentuk semula.

BAB II
KAJIAN TEORI

2.1 Kriptografi

Kriptografi merupakan gabungan dari kata *cryptos* dan *graphein* yang berasal dari bahasa Yunani yang memiliki arti rahasia dan tulisan. Kriptografi secara harfiah berarti tulisan rahasia (Munir, 2019). Kriptografi merupakan ilmu yang mempelajari metode untuk menyamarkan pesan sehingga pesan hanya dapat dibaca oleh penerima (Jamaludin et al., 2022). Menurut Menezes dkk (2014), kriptografi merupakan disiplin ilmu yang didalamnya mempelajari teknik-teknik matematika yang memiliki hubungan dengan keamanan suatu informasi.

Kriptografi menjadi syarat penting dalam keamanan teknologi informasi, terlebih dalam pengiriman pesan rahasia. Proses pengiriman pesan rahasia rentan mengalami penyerangan seperti penyadapan, pemutusan komunikasi, pengubahan isi pesan dan lain sebagainya. Keamanan dalam pengiriman pesan dapat meningkat dengan adanya kriptografi, pengamanan pesan dilakukan dengan mengubah pesan dalam bentuk sandi dengan menggunakan sebuah algoritma dan kunci tertentu yang hanya diketahui oleh pihak-pihak yang berwenang.

2.1.1 Sistem Kriptografi

Kriptografi membentuk sebuah sistem yang disebut sebagai sistem kriptografi (*cryptosystem*). Sistem kriptografi adalah himpunan yang terdiri dari lima bagian sebagai berikut (Sadikin, 2012):

1. *Plaintext*

   Plaintext atau teks asli adalah pesan asli yang dapat terbaca. *Plaintext* merupakan *input* untuk algoritma enkripsi.

2. *Secret Key*

   Secret key adalah kunci rahasia yang merupakan *input* bagi algoritma enkripsi pula. *Secret key* merupakan nilai yang bebas terhadap teks asli (*plaintext*) dan menentukan hasil *output* untuk algoritma enkripsi.

3. *Ciphertext*

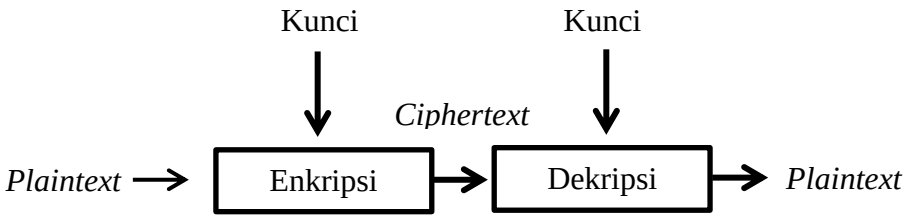
   Ciphertext merupakan *output* dari algoritma enkripsi. *Ciphertext* juga diartikan sebagai pesan yang tersembunyi karena sudah melalui proses enkripsi. *Ciphertext* yang acak dan sulit dipahami dapat dihasilkan dengan menggunakan algoritma enkripsi yang baik.

4. Algoritma Enkripsi

   Terdapat dua *input* yang dibutuhkan dalam algoritma enkripsi, yaitu *plaintext* dan *secret key*. Pada proses algoritma enkripsi *plaintext* ditransformasi sehingga menghasilkan *ciphertext*.

5. Algoritma Dekripsi

   Terdapat dua *input* yang dibutuhkan dalam algoritma dekripsi, yaitu *ciphertext* dan *secret key*. Algoritma dekripsi mengembalikan *ciphertext* menjadi *plaintext* jika *secret key* yang digunakan pada algoritma dekripsi dan algoritma enkripsi sama.



Gambar 2.1 Skema Enkripsi dan Dekripsi

Proses enkripsi menerima *input* berupa *plaintext* dan kunci sehingga mendapatkan sebuah hasil *ciphertext* yang merupakan bentuk sandi dari

plaintext. Sedangkan proses dekripsi menerima *input* berupa *ciphertext* dan kunci sehingga mendapatkan *output* berupa *plaintext* semula.

2.1.2 Tujuan Kriptografi

Sebagaimana cabang ilmu pada umumnya, kriptografi juga memiliki tujuan. Kriptografi bertujuan untuk memberi layanan keamanan antara lain:

1. Kerahasiaan (*confidentiality*)

   Kerahasiaan merupakan layanan kriptografi dengan tujuan menjaga pesan sehingga pihak lain yang tidak berkepentingan tidak dapat membaca pesan tersebut. Kerahasiaan juga disebut dengan *secrecy* atau *privacy*.

2. Integritas data (*data integrity*)

   Layanan kriptografi yang memastikan keaslian dan keutuhan pesan merupakan arti dari integritas data. Pesan yang telah dikirimkan dapat dipastikan tidak dimanipulasi selama pengiriman berlangsung. Sistem keamanan harus mampu untuk mendeteksi keaslian pesan untuk menjaga integritas pesan tersebut. Realisasi layanan data integritas dalam kriptografi berupa fungsi hash dan tanda-tanda digital (*digital signature).*

3. Otentikasi (*authentication*)

   Otentikasi adalah layanan kriptografi dengan cara mengidentifikasi kebenaran pihak-pihak yang melakukan komunikasi. Otentikasi harus dilakukan oleh seluruh pihak yang saling berkomunikasi, sehingga tidak ada keraguan dalam pengiriman maupun penerimaan pesan. Pesan yang dikirimkan melalui saluran komunikasi seperti internet rawan dipalsukan, sehingga pesan tersebut juga harus diidentifikasi terlebih dahulu darimana sumbernya.

4. Anti-penyangkalan (*non-repudiation*)

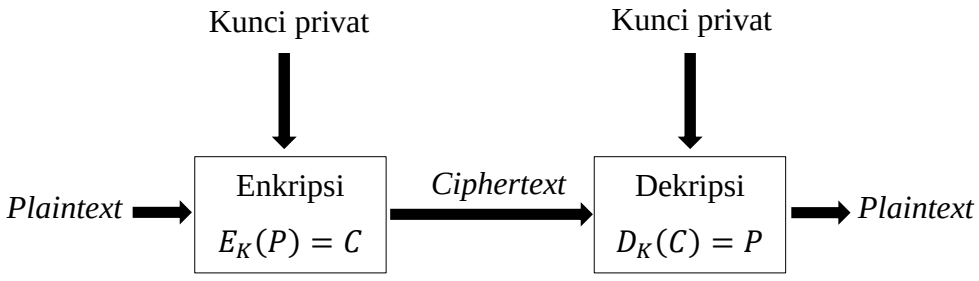
   Layanan kriptografi untuk mencegah pihak yang melakukan komunikasi melakukan penyangkalan terhadap apa yang telah dikirim adalah fungsi dari layanan kriptografi anti-penyangkalan. Tanda tangan pada lembar surat merupakan salah satu anti-penyangkalan dalam surat-menyurat, sedangkan dalam kriptografi hal ini dapat dilakukan dengan penggunaan tanda-tanda digital.

2.1.3 Algoritma Kriptografi

Algoritma kriptografi merupakan langkah-langkah yang disusun secara sistematis dan logis yang digunakan untuk menyembunyikan pesan. Dengan adanya algoritma, proses enkripsi pesan akan menjadi lebih mudah. Algoritma kriptografi dibagi menjadi dua jika dibedakan berdasarkan jenis kunci pada proses enkripsi dan dekripsi antara lain:

1. Kriptografi Kunci Simetri

   Kriptografi kunci simetri (*symmetric-key cryptography*) atau kriptografi konvensional merupakan algoritma kriptografi dimana proses enkripsi dan dekripsinya memiliki kunci yang sama. Letak keamanan pada kriptografi kunci simetri adalah pada kerahasiaan kuncinya (Munir, 2019).



Gambar 2.2 Skema Kriptografi Simetri

Keterangan :

P : *Plaintext*

C :C*iphertext*

K : Kunci privat

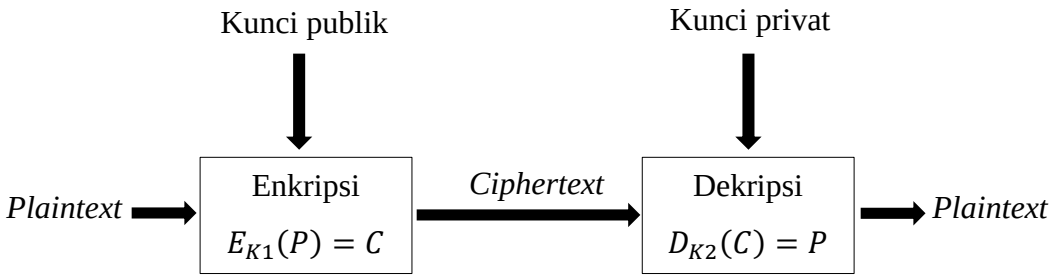
E_K : Enkripsi dengan kunci *K*

D_K : Dekripsi dengan kunci *K*

Proses enkripsi pada kriptografi simetri menerima *input* berupa *plaintext P* dan kunci privat *K* sehingga mendapatkan *output ciphertext C*. Sedangkan proses dekripsi pada kriptografi simetri menerima *input ciphertext C* dan kunci privat *K* yang merupakan kunci yang digunakan pada proses enkripsi sehingga mendapatkan hasil *plaintext P*.

2. Kriptografi Asimetri

   Kriptografi kunci asimetri (*asymmetric-key cryptography*) atau yang memiliki nama lain kriptografi kunci nirsimetri merupakan algoritma kriptografi yang memiliki kunci yang berbeda pada saat proses enkripsi dan dekripsi. Kriptografi asimetri disebut juga dengan kriptografi kunci publik (*public-key cryptography*). Kunci pada proses enkripsi dapat dibagikan kepada semua orang yang tidak mempunyai otoritas atas pesan tersebut, kunci ini disebut sebagai kunci publik (*public key*), sedangkan kunci pada proses dekripsi tidak disebarkan dan hanya digunakan oleh pengirim pesan, kunci ini disebut dengan kunci pribadi (*private key*) (Jamaludin et al., 2022).



Gambar 2.3 Skema Kriptografi Asimetri

Keterangan :

P : *Plaintext*

C : *Ciphertext*

K_1 : Kunci publik

K_2 : Kunci privat

E_{K1} : Enkripsi dengan kunci publik K_1

D_{K2} : Dekripsi dengan kunci privat K_2

Proses enkripsi pada kriptografi simetri menerima *input* berupa *plaintext P* dan kunci publik K_1 sehingga mendapatkan *output ciphertext C*. Sedangkan proses dekripsi pada kriptografi simetri menerima *input ciphertext C* dan kunci privat K_2 sehingga mendapatkan hasil *plaintext P*.

2.1.4 Algoritma Elgamal

Algoritma Elgamal merupakan algoritma yang dibuat oleh Taher Elgamal pada tahun 1984. Pada awal keberadaannya algoritma ini digunakan untuk *digital signature*, seiring berjalannya waktu dilakukan modifikasi pada algoritma Elgamal agar dapat digunakan untuk proses enkripsi dan dekripsi (Munir, 2019).

Algoritma Elgamal merupakan algoritma kriptografi asimetris yang memiliki kunci berbeda untuk proses enkripsi dan dekripsi. Sistem kriptografi

Elgamal bekerja pada sebuah grup perkalian $(\mathbb{G},\times)$ yang pada grup itu persoalan logaritma diskrit sulit dipecahkan. Grup perkalian $\mathbb{G}$ dapat berupa grup perkalian siklik $\langle\alpha\rangle$ dengan α adalah akar primitif pada $(\mathbb{Z}_p^*,\times)$ dengan p merupakan bilangan prima besar.

Secara umum besaran-besaran yang digunakan dalam algoritma Elgamal terdapat pada tabel 2.1 berikut:

Tabel 2.1 Besaran Pada Algoritma Elgamal

No	Besaran	Keterangan
1	Bilangan prima, p	Publik
2	Bilangan bulat g (akar primitif dari p)	Publik
3	Bilangan acak, x	Kunci privat
4	$y = g^x \ mod \ p$	Kunci publik
5	*Plaintext*, m	Privat
6	*Ciphertext*, a dan b	Publik

Prosedur penyandian pesan menggunakan algoritma Elgamal dimulai dengan pembangkitan kunci privat dan kunci publik oleh pihak pengirim dan pihak penerima pesan, kemudian dilanjutkan dengan enkripsi pesan dan diakhiri dengan dekripsi pesan. Penjelasan lebih lengkapnya adalah sebagai berikut:

1. Pembangkitan kunci privat dan kunci publik

   Sebelum masuk kedalam proses enkripsi dilakukan pembangkitan kunci privat dan kunci publik dengan cara pengirim dan penerima pesan menentukan bilangan prima p dan bilangan bulat g yang merupakan akar primitif dari p. p setidaknya memiliki 1 faktor prima besar, jika p hanya memiliki 1 faktor prima kecil maka akan memudahkan penghitungan

logaritma diskrit (ElGamal, 1985). Kemudian pengirim dan penerima pesan membangkitkan kunci privat dan kunci publik dengan memilih sebuah bilangan acak x dengan syarat $1 < x < p - 1$. Kemudian membangkitkan kunci publik dengan persamaan

$$y = g^x \bmod p \tag{2.1}$$

Keterangan:

y : Kunci publik

g : Kunci publik (akar primitif dari p)

x : Bilangan acak, kunci privat

p : Bilangan prima

Sehingga dari hasil penghitungan, diperoleh kunci privat x dan kunci publik (y, g, p).

2. Proses Enkripsi

Langkah pertama sebelum melakukan enkripsi adalah menyatakan pesan sebagai bilangan bulat m dan harus terletak dalam interval $[0, p - 1]$. Untuk m yang besar, bagi m menjadi blok-blok m_1, m_2, ..., yang berukuran lebih kecil sehingga nilai didalam interval $[0, p - 1]$ direpresentasikan oleh setiap blok. Misalkan pengirim pesan sudah mengetahui kunci publik penerima pesan yaitu (y, p, g). Berikut langkah-langkah enkripsi:

a. Pengirim memilih bilangan bulat acak k, dengan $1 \leq k \leq p - 1$.

b. Pengirim mengenkripsi pesan m menjadi pasangan nilai (a, b) dengan persamaan

$$a = g^k \bmod p \tag{2.2}$$

$$b = y^k m \bmod p \tag{2.3}$$

Keterangan:

a : *Ciphertext* 1

b : *Ciphertext* 2

g : Kunci publik (akar primitif p)

k : Bilangan acak

y : Kunci publik

m : *Plaintext*

p : Bilangan prima

Pasangan a dan b adalah *ciphertext* untuk pesan m. Jadi ukuran *ciphertext* dua kali ukuran *plaintext*nya. Pengirim mengirim (a, b) kepada penerima.

3. Proses Dekripsi

Penerima pesan menggunakan kunci privat x untuk mendekripsi a dan b menjadi *plaintext* m dengan persamaan

$$m = \left(\frac{b}{a^x}\right) \bmod p \tag{2.4}$$

$$= b(a^x)^{-1} \bmod p \tag{2.5}$$

Keterangan:

m : *Plaintext*

a : *Ciphertext* 1

b : *Ciphertext* 2

x : Kunci privat

p : Bilangan prima

Contoh :

1. Untuk pembangkitan kunci :

   Misalkan Alice ingin membangkitkan pasangan kuncinya. Alice memilih bilangan prima $p = 2273$, akar primitif dari p yaitu $g = 3$, dan $x = 243$. Alice kemudian menghitung

   $$y = g^x \bmod p = 3^{243} mod\ 2273 = 461$$

   Jadi, kunci publik privat Alice adalah $x = 243$ dan kunci publiknya adalah $y = 461, g = 3, p = 2273$.

2. Untuk enkripsi:

   Misalkan Bob mengirim *plaintext* "HALO" kepada Alice dengan memisalkan pesan-pesan yang akan Bob kirim dengan menggunakan tabel ASCII 256 sehingga didapatkan nilai desimal $\mathrm{A} = 65, \mathrm{B} = 66, \ldots \mathrm{Z} = 90$. Sehingga didapatkan $\mathrm{H} = 72, \mathrm{A} = 65, \mathrm{L} = 76, \mathrm{O} = 79$, maka pesan m dalam *integer* adalah

   $$m = 72657679$$

   Karena nilai m terlalu besar, maka m dibagi menjadi blok-blok yang lebih kecil dalam interval $[0,2273 - 1]$, sehingga didapatkan

   $m_1 = 72, m_2 = 65, m_3 = 76, m_4 = 72$

   - Enkripsi $m_1 = 72$

     Bob memilih $k_1 = 2020$, kemudian menghitung:

     $a_1 = g^{k_1}\ mod\ p = 3^{2020}\ mod\ 2273 = 532$

     $b_1 = y^{k_1} m_1\ mod\ p = 461^{2020}\ \cdot 72\ mod\ 2273 = 179$

     Jadi *ciphertext* yang dihasilkan untuk m_1 adalah $c_1 = (532,179)$.

- Enkripsi $m_2 = 65$

  Bob memilih $k_2 = 62$, kemudian menghitung:

  $a_2 = g^{k_2} \, mod \, p = 3^{62} \, mod \, 2273 = 1649$

  $b_2 = y^{k_2} m_2 \, mod \, p = 461^{62} \cdot 65 \, mod \, 2273 = 1369$

  Jadi *ciphertext* yang dihasilkan untuk m_2 adalah $c_2 = (1649,1369)$.

- Enkripsi $m_3 = 76$

  Bob memilih $k_3 = 69$, kemudian menghitung:

  $a_3 = g^{k_3} \, mod \, p = 3^{69} \, mod \, 2273 = 1385$

  $b_3 = y^{k_3} m_3 \, mod \, p = 461^{69} \cdot 76 \, mod \, 2273 = 2073$

  Jadi *ciphertext* yang dihasilkan untuk m_3 adalah $c_3 = (1385,2073)$.

- Enkripsi $m_4 = 79$

  Bob memilih $k_4 = 60$, kemudian menghitung:

  $a_4 = g^{k_4} \, mod \, p = 3^{60} \, mod \, 2273 = 1446$

  $b_4 = y^{k_4} m_4 \, mod \, p = 461^{60} \cdot 79 \, mod \, 2273 = 928$

  Jadi *ciphertext* yang dihasilkan untuk m_4 adalah $c_4 = (1446,928)$.

Bob mengirim *ciphertext* $(532,179); (1649,1369); (1385,2073)$ dan $(1446,928)$ kepada Alice.

3. Untuk dekripsi

   Alice mendekripsi *ciphertext* dari Bob dengan melakukan perhitungan sebagai berikut:

   - Dekripsi $c_1 = (532,179)$

     $(a_1^x)^{-1} = a_1^{p-1-x} \, mod \, p = 532^{2029} \, mod \, 2273 = 775$

     $m_1 = \frac{b_1}{a_1^x} mod \, p = b_1 (a_1^x)^{-1} \, mod \, p = 179 \cdot 775 \, mod \, 2273 = 72$

- Dekripsi $c_2 = (1649,1369)$

  $(a_2^x)^{-1} = a_2^{p-1-x} \; mod \; p = 1649^{2029} \; mod \; 2273 = 1149$

  $m_2 = \frac{b_2}{a_2^x} mod \; p = b_2(a_2^x)^{-1} \; mod \; p = 1369 \cdot 1149 \; mod \; 2273 = 65$

- Dekripsi $c_3 = (1385,2073)$

  $(a_3^x)^{-1} = a_3^{p-1-x} \; mod \; p = 1385^{2029} \; mod \; 2273 = 136$

  $m_3 = \frac{b_3}{a_3^x} mod \; p = b_3(a_3^x)^{-1} \; mod \; p = 2073 \cdot 136 \; mod \; 2273 = 76$

- Dekripsi $c_4 = (1446,928)$

  $(a_4^x)^{-1} = a_4^{p-1-x} \; mod \; p = 1446^{2029} \; mod \; 2273 = 512$

  $m_4 = \frac{b_4}{a_4^x} mod \; p = b_4(a_4^x)^{-1} \; mod \; p = 928 \cdot 512 \; mod \; 2273 = 79$

Plaintext yang didekripsi adalah $m_1 m_2 m_3 m_4 = 72657679$, dan jika kode desimal tersebut diubah menjadi teks empat digit dengan menggunakan tabel ASCII 256 berbunyi “HALO” yang sama dengan *plaintext* yang dikirim oleh Bob.

2.1.5 Algoritma DNA

Kriptografi DNA merupakan hal yang baru dalam lingkup kriptografi. Kriptografi DNA didefinisikan sebagai alat untuk menyembunyikan data kedalam urutan DNA (Raj et al., 2016). Setiap himpunan DNA terdiri dari 4 macam nukleotida, antara lain : *Adenine* (A), *Guanine* (G), *Cytosin* (C), dan *Thymine* (T). Pada struktur DNA *Adenine* dipasangkan dengan *Thymine*, sedangkan *Guanine* dipasangkan dengan *Cytoshin*. Dalam kriptografi DNA, pasangan DNA digunakan sebagai pembawa informasi (Satir & Kendirli, 2022). *De-Oxy Ribo* merupakan asam nukleat yang dijadikan acuan dalam komputasi

DNA. *De-Oxy Ribo* merupakan asam nukleat yang mengandung informasi genetik yang dibutuhkan pada saat proses pertumbuhan serta fungsi pasa organisme hidup. Komputasi DNA memiliki beberapa keuntungan diantaranya adalah kecepatan dan persyaratan daya minimal. Satu gram DNA mengandung kurang lebih 1021 basis DNA atau setara dengan 108 *tera-byte*. Oleh karena itu beberapa gram DNA berpotensi untuk menyimpan semua data yang terdapat didunia (Mahesa et al., 2019).

Dalam sistem biner 0 dan 1 saling berkomplemen, sehingga 00 dan 11, begitu pula dengan 10 dan 01 juga saling berkomplemen. Apabila 00, 11, 10, dan 01 dikodekan dengan basa nukleat A, T, G, dan C maka diperoleh 24 macam skema penyandian. Karena antar basa DNA memiliki hubungan komplemen, maka terdapat delapan macam kombinasi penyandi yang memenuhi prinsip pasangan basa komplementer.

Tabel 2.2 Bentuk Kombinasi Pengkodean DNA

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
A	00	00	01	01	10	10	11	11
T	11	11	10	10	01	01	00	00
G	01	10	00	11	00	11	01	10
C	10	01	11	00	11	00	10	01

Tahapan yang dilakukan dalam algoritma DNA meliputi enkripsi, pembangkitan kunci acak, dan dekripsi. Berikut langkah-langkah yang dilakukan dalam setiap tahapan tersebut:

1. Enkripsi

   Untuk mendapatkan sebuah *ciphertext*, maka terdapat langkah-langkah berikut dalam proses enkripsi:

   Langkah 1 : input data teks (*plaintext*).

   Langkah 2 : setiap karakter dari data diubah kedalam nilai ASCII desimal.

   Langkah 3 : setelah didapatkan nilai ASCII desimal, kemudian dikonversi ke dalam nilai biner.

   Langkah 4 : nilai biner dikonversi menjadi kode DNA.

   Langkah 5 : penugasan acak kode DNA kedalam kode ASCII yang diperluas berdasarkan Pk kunci privat.

   Langkah 6 : enkripsi pesan menggunakan kode DNA.

   Langkah 7 : permutasi secara siklis pesan yang terenkripsi dengan $Pk \bmod 26$.

2. Pembangkitan Kunci Acak

   Pada tahap ini, kunci acak Pk terdapat dalam interval 1 hingga 256 dihasilkan untuk enkripsi. Sesuai dengan masing-masing nilai Pk, dihasilkan tabel indeks berukuran 256 dimana setiap nilai dari tabel sesuai dengan kombinasi A, T, G, dan C. Berikut langkah-langkah yang terdapat dalam pembangkitan kunci acak:

   Langkah 1 : input data teks (*plaintex*t).

   Langkah 2 : data input dianggap sebagai bentuk pasangan, setiap pasangan digantikan oleh nukleotida DNA yaitu A untuk 00, T untuk 11, G untuk 10, dan C untuk 01.

Langkah 3 : setelah dihasilkan Pk, kemudian sesuaikan dengan nilai dari tabel indeks yang dihasilkan Pk.

Langkah 4 : A, T, G, C yang diperoleh dari langkah 2 digunakan untuk menemukan kunci enkripsi akhir dari tabel indeks yang dihasilkan.

3. Dekripsi

Tahap dekripsi dilakukan untuk mendapatkan kembali *plaintext* seperti semula. Berikut langkah-langkah dekripsi algoritma kriptografi DNA:

Langkah 1 : input kunci privat Pk dan pesan yang terenkripsi.

Langkah 2 : permutasi secara siklis pesan yang terenkripsi dengan kode DNA dengan $Pk\ 26$.

Langkah 3 : kode DNA dikonversi ke nilai binernya.

Langkah 4 : pengacakan kode biner ke kode ASCII yang diperluas.

Langkah 5 : dekripsi *ciphertext* ke teks asli.

Contoh:

Diberikan sebuah *plaintext* "**GIRL**", kemudian akan dilakukan enkripsi dan dekripsi dengan menggunakan algoritma kriptografi DNA.

1. Untuk proses enkripsi

Langkah pertama, tentukan nilai ASCII dari *plaintext*, yaitu $\mathrm{G} = 71, \mathrm{I} = 73, \mathrm{R} = 82, \mathrm{L} = 76$ Sehingga didapatkan nilai biner dan kode DNA sebagai berikut:

$$\mathrm{G}(71) \rightarrow 01000111 \rightarrow \mathrm{CACT}$$

$$\mathrm{I}(73) \rightarrow 01001001 \rightarrow \mathrm{CAGC}$$

$$\mathrm{R}(82) \rightarrow 01010010 \rightarrow \mathrm{CCAG}$$

$$L(76) \rightarrow 01001100 \rightarrow CATA$$

Sehingga pesan "**GIRL**" berubah menjadi **CACT-CAGC-CCAG-CATA**.

Kemudian dari kode DNA yang didapatkan dan berdasarkan pada tabel substitusi kunci acak, didapatkan *ciphertext* yaitu **92, 82, 73, 70**.

2. Untuk proses dekripsi

   Dari proses enkripsi, didapatkan *ciphertext* 92, 82, 73, dan 70. Kemudian lakukan substitusi pembangkit kunci acak sehingga didapatkan kode DNA, setelah mendapatkan kode DNA, kemudian lakukan substitusi kembali untuk mendapatkan kode binernya, sehingga:

$$92 \rightarrow CACT \rightarrow 01000111$$

$$82 \rightarrow CAGC \rightarrow 01001001$$

$$73 \rightarrow CCAG \rightarrow 01010010$$

$$70 \rightarrow CATA \rightarrow 01001100$$

   Selanjutnya dilakukan pengacakan kode biner ke dalam kode ASCII yang diperluas sebagai berikut:

$$01000111 \rightarrow 71 \rightarrow G$$

$$01001001 \rightarrow 73 \rightarrow I$$

$$01010010 \rightarrow 82 \rightarrow R$$

$$01001100 \rightarrow 76 \rightarrow L$$

   Sehingga dari *ciphertext* **92, 82, 73 dan 70** didapatkan kembali *plaintext* "**GIRL**".

2.2 Teori Bilangan

Teori bilangan merupakan cabang dari matematika murni yang mempelajari bilangan bulat dan fungsi yang bernilai bilangan bulat. Selain itu teori bilangan juga membahas mengenai berbagai masalah terbuka yang dengan mudah dapat dipahami oleh orang yang tidak memiliki keahlian dalam bidang matematika.

Persoalan yang terdapat pada teori bilangan merupakan dasar dari kriptografi kunci asimetri atau kriptografi kunci publik (Sadikin, 2012). Pada algoritma Elgamal hal tersebut diterapkan pada persoalan logaritma diskrit yang merupakan persoalan pada teori bilangan.

2.2.1 Keterbagian

Definisi 2.2.1

Misalkan $x, y \in \mathbb{Z}$, dengan $x \neq 0$, maka x disebut membagi y (ditulis $x| y$) jika $y = x\,k$, untuk suatu $k \in \mathbb{Z}$.

Berdasarkan definisi yang telah disebutkan, bilangan bulat x dengan $x \neq 0$, membagi y jika ada suatu bilangan bulat k sedemikian hingga $y = xk$. Notasi $x| y$ dibaca "x membagi y" atau "x faktor dari y" atau "y kelipatan dari x" atau "x pembagi y". Jika x tidak membagi y, maka ditulis dengan notasi $x \nmid y$.

2.2.2 Aritmatika Modulo

Misalkan a adalah bilangan bulat dan m adalah bilangan bulat > 0. Operasi $a \bmod m$ (dibaca "a modulo m") akan memiliki sisa jika a dibagi dengan m. Bilangan m disebut modulus atau modulo, sedangkan hasil dari operasi modulo m terdapat pada himpunan $\{0, 1, 2, \dots, m-1\}$.

Notasi : $a \bmod m = r$ sedemikin sehingga $a = mq + r$, dengan $0 \leq r \leq m$.

Contoh :

Berikut beberapa contoh operasi dengan operator modulo

1. $23\ mod\ 5 = 3 \qquad (23 = 5 \cdot 4 + 3)$
2. $27\ mod\ 3 = 0 \qquad (27 = 3 \cdot 9 + 0)$
3. $6\ mod\ 8 = 6 \qquad (6 = 8 \cdot 0 + 6)$
4. $0\ mod\ 12 = 0 \qquad (0 = 12 \cdot 0 + 0)$

2.2.3 Bilangan Prima

Bilangan prima adalah bilangan bulat positif lebih dari satu yang hanya habis dibagi oleh satu dan bilangan itu sendiri. Bilangan prima memiliki peran penting dalam matematika diskrit. Jumlah bilangan prima adalah tak terhingga, Semakin besar bilangan bulat tersebut semakin jarang ada bilangan prima. Dalam kriptografi bilangan prima merupakan bilangan yang penting, karena beberapa algoritma kunci publik mendasarkan algoritmanya pada bilangan prima salah satunya adalah algoritma Elgamal (Munir, 2019).

Cara menguji apakah n adalah bilangan prima atau bilangan komposit adalah dengan cara membagi n dengan bilangan prima mulai dari $2, 3,$ ..., hingga bilangan prima $\leq \sqrt{n}$. Jika n habis dibagi dengan salah satu dari bilangan prima tersebut, maka n adalah bilangan komposit, akan tetapi jika n tidak habis dibagi oleh semua bilangan prima tersebut, maka n adalah bilangan prima (Sadikin, 2012).

Contoh :

Dipilih sebuah bilangan yaitu 4271, buktikan bahwa bilangan tersebut merupakan bilangan prima!

Bukti:

Untuk membuktikan bahwa 4271 adalah bilangan prima langkah pertama yang dilakukan adalah mencari akar dari 4271, sehingga diperoleh $\sqrt{4271} = 65{,}359$. Kemudian tentukan bilangan prima $\leq \sqrt{4271}$ dan didapatkan bilangan prima 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, dan 61. Selanjutnya bagi 4271 dengan masing-masing bilangan prima $\leq \sqrt{4271}$ sebagai berikut:

4271 : 2 = 2.135,5	4271 : 19 = 224,8	4271 : 47 = 90,9
4271 : 3 = 1.423,7	4271 : 23 = 185,7	4271 : 53 = 80,6
4271 : 5 = 845,2	4271 : 29 = 147,3	4271 : 59 = 72,4
4271 : 7 = 610,1	4271 : 31 = 137,8	4271 : 61 = 70,1
4271 : 11 = 388,3	4271 : 37 = 115,4	
4271 : 13 = 328,5	4271 : 41 = 104,2	
4271 : 17 = 251,2	4271 : 43 = 99,3	

Karena 4271 tidak habis dibagi dengan bilangan prima $\leq \sqrt{4271}$, maka terbukti bahwa 4271 adalah bilangan prima.

2.2.4 Teorema Fermat

Misalkan p prima maka:

1. Untuk setiap bilangan bulat b,

$$b^p - b \equiv 0 \ (mod\ p) \tag{2.6}$$

2. Jika $b \neq 0 \ (mod\ p)$,

$$b^{p-1} \equiv 1 (mod\ p) \tag{2.7}$$

(Kraft & Washington, 2015).

Contoh:

Akan diuji apakah 19 bilangan prima atau bukan dengan menggunakan Teorema Fermat. Ambil nilai $b = 2$ karena PBB$(19,2) = 1$. Selanjutnya hitung $2^{19-1} = 262144 \equiv 1\ (mod\ 19)$. Karena 19 habis membagi $262144 - 1 = 262143$, maka 19 merupakan bilangan prima.

2.2.5 Teorema Euler

Misalkan a dan n adalah dua bilangan yang relatif prima dan $\phi(n)$ adalah fungsi totient euler, maka berlaku teorema euler berikut

$$a^{\phi(n)} \equiv 1(mod\ n) \tag{2.8}$$

Contoh :

Misalkan $a = 7$ dan $n = 10$, keduanya relatif prima, $\phi(10) = 4$, maka $7^4 = 2041 \equiv 1(mod\ 10)$.

2.2.6 Akar Primitif

Nilai a adalah akar primitif modulo bilangan bulat positif n jika pangkatnya menghasilkan seluruh kelompok unit modulo n (Batten, 2012). Definisi lain menyebutkan misalkan n adalah bilangan bulat, maka a adalah akar primitif dari n jika perpangkatan dari $a, a^2, \dots, a^{\phi(n)}$ (dalam sebuah modulus) menghasilkan nilai yang berbeda dan semuanya relatif prima dengan n. Khusus untuk bilangan prima, jika p adalah bilangan prima, maka a disebut akar primitif dari p jika perpangkatan $a, a^2, \dots, a^{p-1}$ (dalam modulus p) menghasilkan nilai-nilai yang berbeda (Munir, 2019a).

Contoh :

Misalkan $p = 5$, maka $a = 2$ adalah akar primitif dari p karena

$2^1(mod\ 5) = 2$

$2^2(mod\ 5) = 4$

$2^3(mod\ 5) = 3$

$2^4(mod\ 5) = 1$

Karena semua perpangkatan dari 2 menghasilkan nilai-nilai yang berbeda yaitu 2,4,3,1 dan semua bilangan didalam modulus 5 terjadi satu kali. Perpangkatan berikutnya akan kembali berulang menghasilkan nilai-nilai tersebut. Sehingga untuk bilangan prima 5 maka panjang siklus tidak lebih dari $5 - 1 = 4$.

2.2.7 Eksponensial

Dalam sebuah medan terbatas mencari eksponensial $g^a = h$ mudah. Eksponensial merupakan operasi modular yang sering dijumpai pada sistem kriptografi. Eksponensial modular adalah operasi pada persamaan

$$y = a^x mod\ n \tag{2.9}$$

Untuk menghitung eksponensial agar lebih cepat dan efisien, maka digunakan sebuah algoritma yang bernama *Square and Multiply.* Algoritma *Square and Multiply* mengasumsikan eksponen x dalam bentuk biner sehingga:

$$x = \sum_{i=0}^{l-1} x_i 2^i \tag{2.10}$$

dengan l adalah panjang x dalam biner. Dengan x_i bernilai 0 atau 1, $0 \leq i \leq l - 1$.

2.2.8 Permasalahan Logaritma Diskrit

Logaritma disebut juga sebagai invers eksponensial. Pada sebuah grup, logaritma adalah masalah yang sulit untuk diselesaikan. Sistem kriptografi Elgamal merupakan salah satu sistem dalam kriptografi yang mengalami kesulitan dalam penyelesaian logaritma diskrit.

Logaritma diskrit memiliki fungsi yang sama dengan logaritma biasa, yaitu logaritma sebuah bilangan y untuk basis g adalah nilai eksponensial x terhadap basis g agar menghasilkan bilangan tersebut. Fungsi logaritma dapat direpresentasikan sebagai fungsi berikut

$$log_g(y) = x \rightarrow g^x = y \tag{2.11}$$

2.4 ASCII

ASCII (*American Standard Code for Information Interchange*) atau dalam bahasa Indonesia berarti kode standar Amerika untuk pertukaran informasi merupakan standar internasional dalam kode huruf dan simbol. Pada awalnya kode ASCII memiliki komposisi bilangan biner sebanyak 7 bit. Namun ASCII disimpan sebagai sandi 8 bit dengan menambahkan 0 sebagai bit *significant* yang paling tinggi (Tantoni & Zaen, 2018). Komposisi bilangan biner sebanyak 8 bit dimulai dari 0000 0000 hingga 1111 1111. Kode ASCII memiliki jumlah sebanyak 255 kode. Kode ASCII yang digunakan untuk memanipulasi teks adalah kode ASCII 0..127. Kode ASCII dikelompokkan menjadi beberapa bagian, yaitu kode yang tidak terlihat simbolnya seperti 32 (space), kemudian kode yang terlihat simbolnya seperti abjad dari A-Z, angka 0-9, dan juga beberapa karakter khusus, dan yang terakhir adalah kode yang tidak ada di keyboard tetapi dapat ditampilkan, kode ini biasanya digunakan untuk kode-kode grafik. Saat ini standart yang banyak digunakan pada komputer maupun perangkat komunikasi adalah ASCII.

2.5 Kajian Integrasi Topik dengan Al-Qur'an/Hadist

Konsep dalam Al-Qur'an yang berhubungan dengan ilmu kriptografi terdapat pada surat An-Nisa' ayat 58 yang berbunyi:

إِنَّ ٱللَّهَ يَأۡمُرُكُمۡ أَن تُؤَدُّواْ ٱلۡأَمَٰنَٰتِ إِلَىٰٓ أَهۡلِهَا وَإِذَا حَكَمۡتُم بَيۡنَ ٱلنَّاسِ أَن تَحۡكُمُواْ بِٱلۡعَدۡلِۚ إِنَّ ٱللَّهَ نِعِمَّا يَعِظُكُم بِهِۦٓۗ إِنَّ ٱللَّهَ كَانَ سَمِيعَۢا بَصِيرٗا

Artinya : "Sungguh, Allah menyuruhmu menyampaikan amanat kepada yang berhak menerimanya, dan apabila kamu menetapkan hukum di antara manusia hendaknya kamu menetapkannya dengan adil. Sungguh Allah sebaik-baik yang memberi pengajaran kepadamu. Sungguh, Allah Maha Mendengar, Maha Melihat". (QS. An-Nisa'/4:58).

Dalam ayat ini dijelaskan bahwa amanah merupakan sesuatu yang diberikan kepada satu pihak ke pihak yang lain untuk dipelihara dan dikembalikan jika sudah waktunya atau jika diminta kembali oleh pemiliknya. Amanah harus disampaikan kepada orang yang berhak menerimanya, baik amanah itu berasal dari Allah maupun dari manusia. Dalam islam, amanah merupakan asas keimanan, hal ini didasarkan pada sabda Nabi Muhammad SAW yang memiliki arti "Selanjutnya, amanah yang merupakan lawan dari khianat adalah sendi utama interaksi. Amanah tersebut membutuhkan kepercayaan dan kepercayaan itu melahirkan ketenangan batin yang selanjutnya melahirkan keyakinan". Amanah memiliki banyak macam, seperti amanah antara manusia dengan Allah, amanah antar manusia, amanah antara manusia dan lingkungan, bahkan amanah kepada dirinya sendiri (Shihab, 2002).

Ayat ini juga memerintahkan kita untuk berbuat adil dalam menetapkan suatu hukum, karena pada dasarnya masing-masing manusia sudah memiliki amanah sejak sebelum kelahirannya. Penetapan suatu hukum tidak bisa dilakukan oleh sembarang orang. Karena itu dalam menetapkan suatu hukum perlu adanya sebuah pengetahuan yang berhubungan dengan hukum dan cara untuk menetapkannya. Sehingga perintah berbuat adil tersebut ditujukan untuk orang-orang yang tersebut. Sifat amanah dan adil harus ditegakkan oleh semua orang tanpa memandang agama, keturunan, maupun ras. Dengan kata lain kedua sifat tersebut ditujukan kepada seluruh umat manusia yang ada di muka bumi ini.

Perintah untuk kewajiban memiliki sifat amanah juga disebutkan dalam sebuah hadist Nabi riwayat Bukhari dan Muslim yang berbunyi:

حَدَّثَنَا إِسْمَاعِيلُ حَدَّثَنِي مَالِكٌ عَنْ عَبْدِ اللَّهِ بْنِ دِينَارٍ عَنْ عَبْدِ اللَّهِ بْنِ عُمَرَ رَضِيَ اللَّهُ عَنْهُمَا أَنَّ رَسُولَ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ قَالَ أَلَا كُلُّكُمْ رَاعٍ وَكُلُّكُمْ مَسْئُولٌ عَنْ رَعِيَّتِهِ فَالْإِمَامُ الَّذِي عَلَى النَّاسِ رَاعٍ وَهُوَ مَسْئُولٌ عَنْ رَعِيَّتِهِ وَالرَّجُلُ رَاعٍ عَلَى أَهْلِ بَيْتِهِ وَهُوَ مَسْئُولٌ عَنْ رَعِيَّتِهِ وَالْمَرْأَةُ رَاعِيَةٌ عَلَى أَهْلِ بَيْتِ زَوْجِهَا وَوَلَدِهِ وَهِيَ مَسْئُولَةٌ عَنْهُمْ وَعَبْدُ الرَّجُلِ رَاعٍ عَلَى مَالِ سَيِّدِهِ وَهُوَ مَسْئُولٌ عَنْهُ أَلَا فَكُلُّكُمْ رَاعٍ وَكُلُّكُمْ مَسْئُولٌ عَنْ رَعِيَّتِهِ

Artinya :"Telah menceritakan kepada kami [Ismail] Telah menceritakan kepadaku [Malik] dari [Abdullah bin Dinar] dari [Abdullah bin Umar] radliallahu 'anhuma, Rasulullah shallallahu 'alaihi wasallam bersabda: "ketahuilah Setiap kalian adalah pemimpin, dan setiap kalian akan dimintai pertanggungjawabannya atas yang di pimpin, penguasa yang memimpin rakyat banyak dia akan dimintai pertanggungjawaban atas yang dipimpinnya, setiap

kepala keluarga adalah pemimpin anggota keluarganya dan dia dimintai pertanggungjawaban atas yang dipimpinnya, dan isteri pemimpin terhadap keluarga rumah suaminya dan juga anak-anaknya, dan dia akan dimintai pertanggungjawabannya terhadap mereka, dan budak seseorang juga pemimpin terhadap harta tuannya dan akan dimintai pertanggungjawaban terhadapnya, ketahuilah, setiap kalian adalah bertanggung jawab atas yang dipimpinnya."” (HR. Bukhari Muslim).

Dalam hadist diatas dijelaskan bahwa kita sebagai manusia adalah seorang pemimpin, dan setiap pemimpin memiliki tanggung jawab atas apa yang dipimpin. Setiap orang memiliki bawahan dan para bawahan tersebut harus diperlakukan dengan adil baik dalam urusan agama maupun dunia. Oleh karenanya orang yang sudah diangkat sebagai pengemban amanat (penanggung jawab) terhadap sesuatu, maka dia harus melakukan yang terbaik dan bersungguh-sungguh dalam mengurusnya. Selain itu, hadist ini juga memerintahkan kita untuk melakukan kewajiban dan memenuhi hak dalam memimpin sesuatu. Kata pertanggungjawaban dimaksudkan sebagai bukti tindakan kepada orang-orang yang dipimpin apakah sang pemimpin dapat melakukan tugas serta kewajibannya dengan baik atau tidak.

2.6 Kajian Topik dengan Teori Pendukung

Seiring dengan perkembangan zaman, informasi semakin mudah untuk diakses dimanapun dan kapanpun. Akan tetapi dengan kemudahan akses tersebut tidak menutup kemungkinan bahwa suatu informasi juga akan lebih mudah untuk dicuri dan disalahgunakan. Terlebih lagi dengan semakin maraknya penggunaan

media sosial dalam kegiatan sehari-hari akan memudahkan penyadap untuk mencuri informasi yang didistribusikan melalui internet khususnya media sosial. Untuk mengantisipasi hal-hal yang tidak diinginkan maka perlu adanya bantuan dengan memanfaatkan cabang dari ilmu matematika yaitu kriptografi.

Kriptografi dapat membantu menyamarkan sebuah informasi agar tidak diketahui orang lain selain yang berhak menerimanya. Penerapan ilmu kriptografi tidak terlepas dari konsep matematika dalam bidang aljabar. Seperti penggunaan teori bilangan dalam penghitungan kunci dalam algoritma kriptografi, penggunaan bilangan prima untuk menentukan nilai p pada algoritma Elgamal, kemudian penggunaan teori tentang akar primitif untuk mencari nilai g dalam proses pembangkitan kunci. Dalam penelitian ini digunakan algoritma DNA dan algoritma Elgamal untuk mengamankan pesan teks. Kedua algoritma tersebut digunakan agar mendapatkan tingkat keamanan yang lebih tinggi, karena algoritma DNA merupakan algoritma kunci simetri dan algoritma Elgamal merupakan algoritma kunci asimetri. Algoritma kunci asimetri memiliki tingkat keamanan yang lebih tinggi daripada algoritma kunci simetri, sehingga gabungan antara keduanya diharapkan memiliki tingkat keamanan yang lebih.

Peneliti membangkitkan kunci pada algoritma Elgamal dengan memodifikasinya menggunakan algoritma kriptografi DNA, sehingga pembangkit kunci tersebut memiliki formula yang berbeda dengan pembangkit kunci yang asli. Kunci publik dan kunci privat adalah dua kunci yang dimiliki algoritma Elgamal, dalam penelitian ini peneliti hanya membangkitkan kunci publik. Setelah proses modifikasi kunci sudah selesai, kemudian dilanjutkan dengan mengenkripsi *plaintext* dengan menggunakan algoritma Elgamal hingga

mendapatkan *ciphertext*. Selanjutnya, untuk mengembalikan kembali *ciphertext* hingga menjadi *plaintext* seperti semula, peneliti melakukan proses dekripsi menggunakan algoritma Elgamal.

BAB III
METODE PENELITIAN

3.1 Jenis Penelitian

Penelitian ini merupakan jenis penelitian kualitatif. Penelitian kualitatif digunakan untuk meneliti suatu kondisi objek alamiah, dimana penelitian berangkat dari sebuah data yang kemudian memanfaatkan teori yang ada sebagai bahan untuk penjelas dan berakhir dengan sebuah teori (Harahap, 2020). Metode yang digunakan pada penelitian ini adalah metode studi literatur, dimana untuk mengkaji hal-hal yang akan dibahas dalam penelitian menggunakan bantuan buku, jurnal, artikel ilmiah, serta tugas akhir yang berhubungan dengan kriptografi.

3.2 Pra Penelitian

Pra penelitian pada penelitian ini adalah dengan mempelajari buku, jurnal, artikel ilmiah, serta referensi lainnya yang berhubungan dengan kriptografi, algoritma DNA, serta algoritma Elgamal. Selain itu penulis juga mempelajari bagaimana cara untuk memodifikasi pembangkit kunci yang terdapat pada algoritma Elgamal.

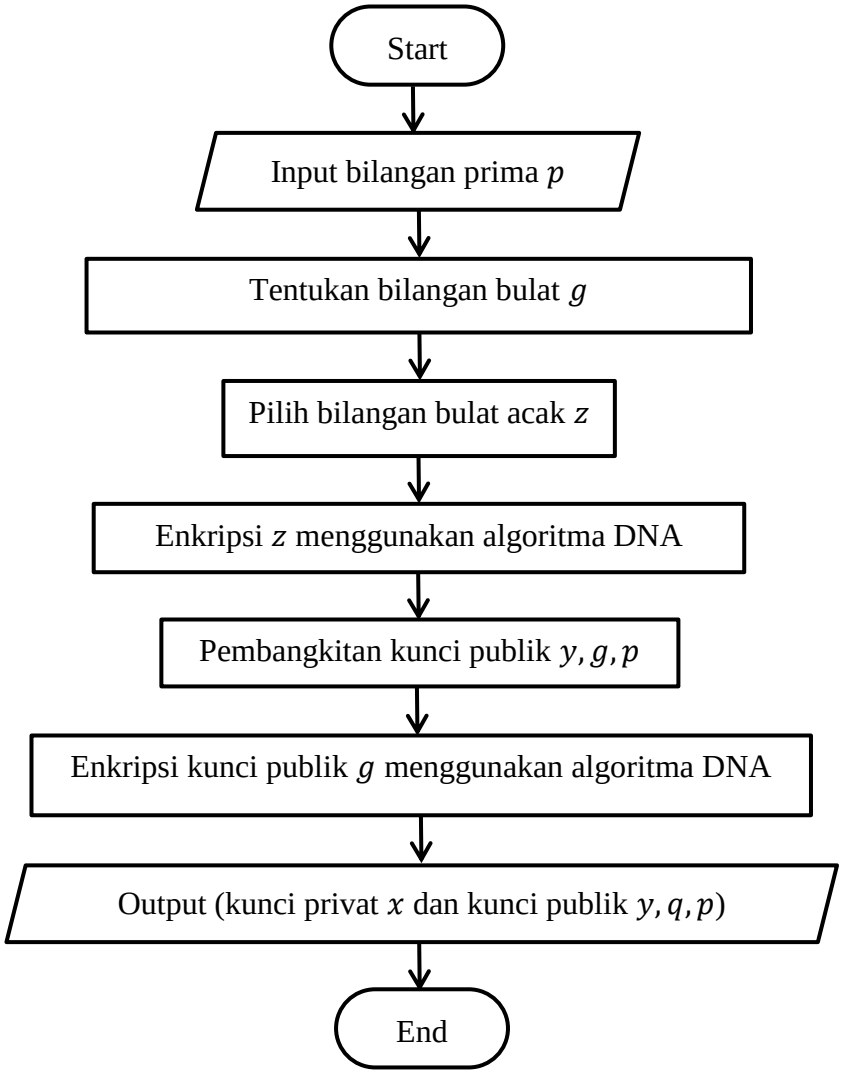
3.3 Tahapan Penelitian

3.3.1 Langkah Modifikasi Pembangkit Kunci pada Algoritma Elgamal dengan Menggunakan Algoritma DNA

Langkah-langkah modifikasi pembangkitan kunci pada algoritma Elgamal dengan menggunakan algoritma DNA adalah sebagai berikut:

1. Penerima pesan menentukan bilangan prima p dan bilangan bulat g.

2. Pembangkitan kunci privat dengan menentukan bilangan bulat acak z.
3. Enkripsi z dengan menggunakan algoritma DNA sehingga hasil enkripsi dari z digunakan sebagai kunci privat x.
4. Pembangkitan kunci publik dengan menggunakan persamaan (2.1) sehingga didapatkan kunci publik (y, g, p).
5. Enkripsi kunci publik g dengan menggunakan algoritma DNA dan menghasilkan q.
6. Penerima pesan mengirimkan kunci publik(y, q, p) ke pengirim pesan.

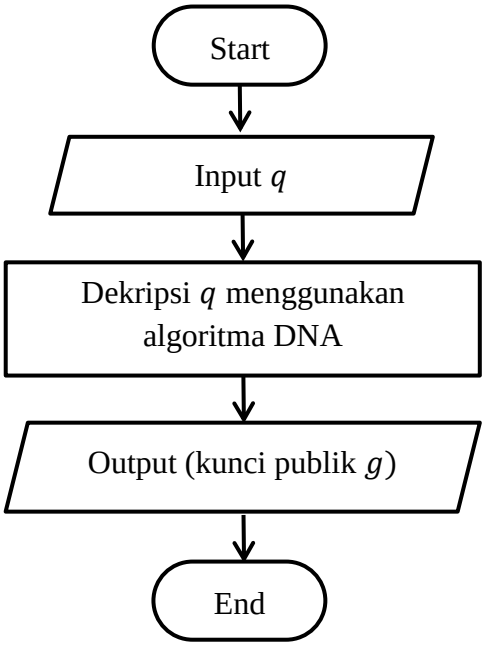


Gambar 3.1
Skema Proses Modifikasi Pembangkitan Kunci pada Algoritma Elgamal

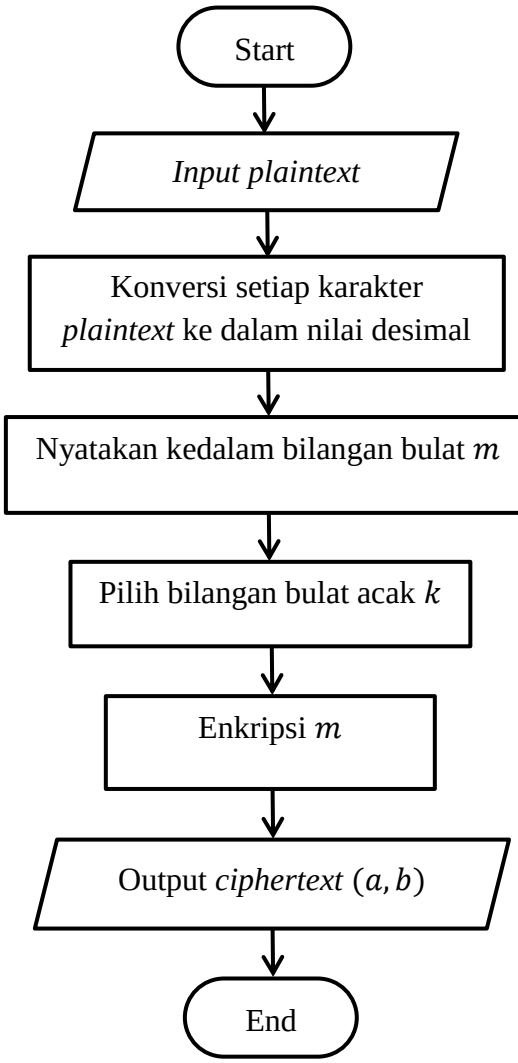
3.3.2 Langkah Enkripsi Pesan dengan Menggunakan Algoritma Elgamal

Langkah-langkah enkripsi pesan dengan menggunakan algoritma Elgamal adalah sebagai berikut:

1. Dekripsi kunci publik q dengan menggunakan algoritma DNA.
2. Tentukan *plaintext* yang akan dienkripsi.
3. Mengkonversi setiap karakter pada *plaintext* ke dalam nilai desimal
4. Kemudian nyatakan *plaintext* kedalam bentuk bilangan bulat m.
5. Pilih bilangan bulat acak k untuk digunakan dalam proses enkripsi m.
6. Enkripsi pesan m menjadi pasangan nilai (a, b) dengan menggunakan persamaan (2.2) dan (2.3).
7. Pasangan (a, b) merupakan *ciphertext* dari m.



Gambar 3.2
Dekripsi Kunci Publik q

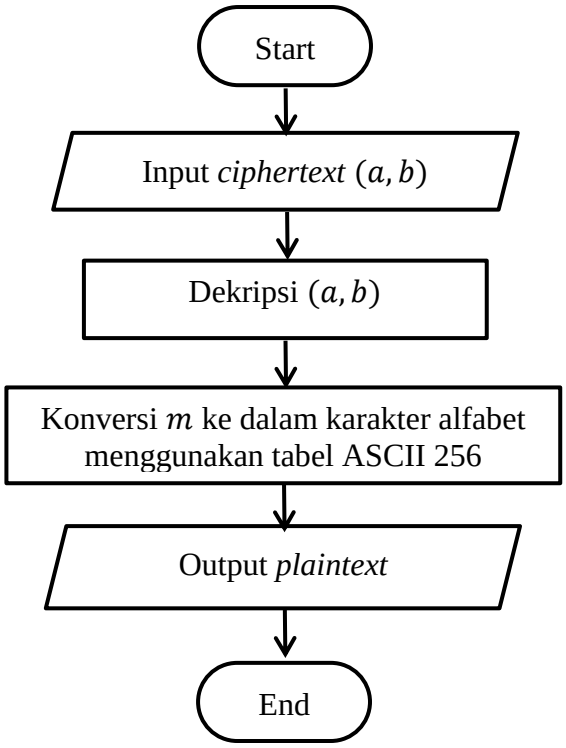


Gambar 3.3
Skema Proses EnkripsiMenggunakan Algoritma Elgamal

3.3.3 Langkah Dekripsi Pesan dengan Menggunakan Algoritma Elgamal

Proses dekripsi pesan menggunakan algoritma Elgamal dengan kunci yang telah dimodifikasi dengan langkah sebagai berikut:

1. *Input ciphertext* (a, b) yang merupakan hasil dari proses enkripsi.
2. Dekripsi *ciphertext* (a, b) dengan menggunakan persamaan (2.4).
3. Kemudian konversi m ke dalam karakter yang sesuai.
4. Setelah itu didapatkan *output* berupa *plaintext*.



Gambar 3.4
Skema Proses Dekripsi Menggunakan Algoritma Elgamal

BAB IV
HASIL DAN PEMBAHASAN

4.1 Algoritma Elgamal dan Algoritma DNA

4.1.1 Modifikasi Pembangkit Kunci Algoritma Elgamal dengan Menggunakan Algoritma DNA

Pembangkitan kunci pada algoritma Elgamal dilakukan oleh penerima pesan. Adapun kunci yang dibangkitkan merupakan kunci privat dan salah satu kunci publik. Langkah pertama untuk membangkitkan kunci privat adalah ambil sembarang bilangan prima p besar kemudian tentukan bilangan bulat g yang merupakan akar primitif dari p. Selanjutnya pilih bilangan bulat acak z dengan syarat $1 < z < 255$ karena nilai desimal tertinggi dalam tabel ASCII 256 adalah 255. Dilanjutkan dengan mengenkripsi z dengan menggunakan algoritma DNA dengan cara mengkonversi z ke dalam kode biner. Setelah didapatkan kode biner kemudian konversi kode biner kedalam kode DNA, data input biner dianggap sebagai bentuk pasangan yang digantikan oleh nukleotida DNA yaitu A untuk 00, T untuk 11, G untuk 10 dan C untuk 01. Kemudian konversi kode DNA yang telah diperoleh dengan menggunakan tabel kunci pembangun acak pada kriptografi DNA. Dari hasil konversi menggunakan kunci pembangun acak tersebut didapatkan sebuah kunci privat x.

Langkah selanjutnya dilakukan pembangkitan kunci publik dengan menggunakan persamaan

$$y = g^x mod\ p$$

Dari penghitungan tersebut didapatkan kunci publik (y, g, p). Sebelum penerima pesan mengirimkan kunci publik ke pengirim pesan, akan dilakukan enkripsi

kunci publik g dengan menggunakan algoritma DNA. Adapun langkah langkah untuk mengenkripsi kunci publik g adalah dengan mongkonversi g kedalam kode biner. Setelah didapatkan kode biner kemudian konversi kode biner kedalam kode DNA, data input biner dianggap sebagai bentuk pasangan yang digantikan oleh nukleotida DNA yaitu A untuk 00, T untuk 11, G untuk 10 dan C untuk 01. Kemudian konversi kode DNA yang telah diperoleh dengan menggunakan tabel kunci pembangun acak pada kriptografi DNA. Hasil enkripsi kunci publik g yaitu q akan dikirimkan penerima pesan pada pengirim pesan bersamaan dengan kunci publik (y, p).

4.1.2 Enkripsi Pesan dengan Menggunakan Algoritma Elgamal

Pengirim pesan menerima kunci publik (y, q, p) yang dikirim oleh penerima pesan. Sebelum mengenkripsi *plaintext* yang akan dikirimkan, pengirim pesan mendekripsikan kunci publik q terlebih dahulu dengan langkah sebagai berikut:

1. Mengkonversi nilai q dengan menggunakan tabel kunci pembangun acak pada kriptografi DNA sehingga mendapatkan sebuah kode DNA yang sesuai.
2. Kemudian kemudian konversi kode DNA yang diperoleh kedalam kode biner dengan mengkonversi 00 untuk A, 11 untuk T, 10 untuk G dan 01 untuk C.
3. Konversi kode biner ke dalam nilai desimal yang sesuai.
4. Hasil dari proses dekripsi tersebut merupakan kunci publik g yang merupakan akar primitif dari p yang kemudian digunakan pada proses enkripsi menggunakan algoritma Elgamal.

Setelah proses dekripsi pada q, langkah selanjutnya adalah enkripsi *plaintext* dengan menggunakan algoritma Elgamal dengan langkah sebagai berikut:

1. Menentukan *plaintext* yang akan dienkripsi.
2. Kemudian konversi setiap karakter pada *plaintext* ke dalam nilai desimal.
3. Selanjutnya nyatakan ke dalam bentuk bilangan bulat m yang terletak dalam interval $[0, p-1]$. Jika nilai m terlalu besar, maka bagi m menjadi blok-blok $m_1, m_2, \ldots$ yang berukuran lebih kecil dan setiap bloknya merepresentasikan nilai dalam interval $[0, p-1]$.
4. Pilih bilangan bulat acak k dengan syarat $1 \leq k \leq p-1$.
5. Enkripsi pesan m menjadi pasangan nilai (a, b) dengan menggunakan persamaan

$$a = g^k \bmod p$$

$$b = y^k m \bmod p$$

   Pasangan a, b merupakan *ciphertext* untuk pesan m.
6. Mengirimkan *ciphertext* kepada penerima pesan.

4.1.3 Dekripsi Pesan dengan Menggunakan Algoritma DNA

Dekripsi dilakukan oleh penerima pesan yang sudah dienkripsi. Adapun proses dekripsi pesan dengan menggunakan algoritma Elgamal adalah sebagai berikut:

1. Input *ciphertext* (a, b) yang sebelumnya sudah dikirim oleh pengirim pesan.
2. Dekripsi *ciphertext* (a, b) menggunakan persamaan

$$m = \left(\frac{b}{a^x}\right) \bmod p$$

$$= b(a^x)^{-1} \bmod p$$

3. Kemudian dari penghitungan tersebut didapatkan nilai m.
4. Selanjutnya konversi nilai m menjadi karakter yang sesuai.
5. Setelah itu penerima pesan mendapatkan *output* berupa *plaintext* yang merupakan pesan rahasia yang dikirimkan oleh pengirim pesan.

4.2 Simulasi Modifikasi Pembangkit Kunci Algoritma Elgamal dengan Menggunakan Algoritma DNA

4.2.1 Proses Modifikasi Pembangkitan Kunci Algoritma Elgamal dengan Menggunakan Algoritma DNA

Pada proses ini akan dilakukan pembangkitan kunci privat x dan kunci publik g pada algoritma Elgamal oleh penerima pesan dengan menggunakan algoritma DNA. Langkah pertama pilih sembarang bilangan prima besar, misal $p = 4271$. Selanjutnya cari akar primitif dari 4271, pilih salah satu yaitu 7, sehingga $g = 7$. Kemudian tentukan nilai z dimana $1 < z < 255$ dan diperoleh nilai $z = 212$. Enkripsi nilai z menggunakan algoritma DNA dengan cara mengkonversi $z = 212$ ke dalam kode biner sehingga diperoleh kode biner 11010100. Kemudian ubah kode biner 11010100 ke dalam kode DNA sebagai berikut

$$11 \rightarrow \mathrm{T}$$

$$01 \rightarrow \mathrm{C}$$

$$01 \rightarrow \mathrm{C}$$

$$00 \rightarrow \mathrm{A}$$

Sehingga didapatkan kode DNA TCCA, dari kode DNA tersebut kemudian dikonversi kembali dengan menggunakan kunci pembangun acak pada kriptografi

DNA sehingga didapatkan nilai 101. Dari nilai tersebut didapatkan kunci privat x, sehingga $x = 101$. Kemudian hitung nilai y sebagai berikut

$$y = g^x mod\ p$$
$$= 7^{101} mod\ 4271$$
$$= 763$$

Dari penghitungan tersebut, didapatkan $y = 763$. Sehingga dari proses pembangkitan kunci yang telah dijelaskan didapatkan kunci publik $y = 763$, $g = 7$ dan $p = 4271$. Sebelum penerima pesan mengirim kunci publik ke pengirim pesan, dilakukan enkripsi terhadap kunci publik g terlebih dahulu dengan menggunakan algoritma DNA dengan cara mengkonversi nilai $g = 7$ ke dalam kode biner sehingga didapatkan kode biner 00000111. Kemudian konversi kode biner kedalam kode DNA

$$00 \rightarrow \mathrm{A}$$
$$00 \rightarrow \mathrm{A}$$
$$01 \rightarrow \mathrm{C}$$
$$11 \rightarrow \mathrm{T}$$

Sehingga didapatkan kode DNA AACT. Dan langkah terakhir adalah konversi kode DNA menggunakan kunci pembangun acak pada kriptografi DNA dan didapatkan nilai 28, sehingga $q = 28$. Karena kunci publik g sudah di enkripsi menjadi q, jadi penerima pesan mengirimkan kunci publik $y = 763, q = 28$ dan $p = 4271$ ke pengirim pesan.

4.3 Simulasi Enkripsi Pesan dengan Menggunakan Algoritma Elgamal

4.3.1 Proses Dekripsi Kunci Publik

Dekripsi kunci publik $q = 28$ menggunakan algoritma DNA dengan cara mengkonversi $q = 28$ menggunakan kunci pembangkit acak pada kriptografi DNA, sehingga didapatkan kode DNA AACT. Selanjutnya konversi kode DNA ke dalam kode biner

$$\mathrm{A} \rightarrow 00$$

$$\mathrm{A} \rightarrow 00$$

$$\mathrm{C} \rightarrow 01$$

$$\mathrm{T} \rightarrow 11$$

Didapatkan kode biner 00000111. Kemudian konversi kode biner ke dalam nilai desimal dan diperoleh nilai desimal 7, jadi $g = 7$. Nilai g yang selanjutnya digunakan pada proses enkripsi. Sehingga pengirim pesan mempunyai kunci publik $y = 763, g = 7 \text{ dan } p = 4271$ untuk mengenkripsi pesan yang akan dikirimkan.

4.3.2 Proses Enkripsi Pesan

Plaintext : CHANGE YOUR THOUGHTS AND YOU CHANGE YOUR WORLD

Sebelum melakukan enkripsi pada *plaintext*, konversi masing-masing karakter pada *plaintext* ke dalam nilai desimal dengan menggunakan tabel ASCII 256 seperti pada tabel 4.1 berikut:

Tabel 4.1
Konversi Karakter Alfabet Menjadi Kode ASCII (desimal)

Karakter	**Kode ASCII (desimal)**
C	67
H	72
A	65
N	78
G	71
E	69
Space	32
Y	89
O	79
U	85
R	82
Space	32
T	84
H	72
O	79
U	85
G	71
H	72
T	84
S	83
Space	32
A	65
N	78

Karakter	**Kode ASCII (desimal)**
D	68
Space	32
Y	89
O	79
U	85
Space	32
C	67
H	72
A	65
N	78
G	71
E	69
Space	32
Y	89
O	79
U	85
R	82
Space	32
W	87
O	79
R	82
L	76
D	68

Dari tabel di atas didapatkan

$$m = 677265787169328979858232847279857172848332657868328979853267726578716932897985823287798276 68$$

Karena nilai m terlalu besar, maka nilai m dibagi menjadi blok-blok yang lebih kecil dalam interval $[0{,}4271-1]$, sehingga didapatkan beberapa nilai m yaitu, $m_1 = 677, m_2 = 265, m_3 = 787, m_4 = 169, m_5 = 328, m_6 = 979, m_7 = 858, m_8 = 232, \ m_9 = 847, \ m_{10} = 279, \ m_{11} = 857, \ m_{12} = 172, \ m_{13} = 848, m_{14} = 332, \ m_{15} = 657, \ m_{16} = 868, \ m_{17} = 328, \ m_{18} = 979, \ m_{19} = 853, m_{20} = 267, \ m_{21} = 726, \ m_{22} = 578, \ m_{23} = 716, \ m_{24} = 932, \ m_{25} = 897, m_{26} = 985, m_{27} = 823, m_{28} = 287, m_{29} = 798, m_{30} = 276, m_{31} = 68.$

Dari nilai m yang diperoleh, akan dilakukan proses enkripsi terhadap masing-masing nilai m dengan cara sebagai berikut:

1. $m_1 = 677$

   Pilih $k_1 = 208$

   $a_1 = g^{k_1} \bmod p = 7^{208} \bmod 4271 = 3880$

   $b_1 = y^{k_1} m_1 \bmod p = 763^{208} \cdot 677 \bmod 4271 = 320$

   Sehingga didapatkan *ciphertext* $c_1 = (3880{,}320)$.

2. $m_2 = 265$

   Pilih $k_2 = 30$

   $a_2 = g^{k_2} \bmod p = 7^{30} \bmod 4271 = 2076$

   $b_2 = y^{k_2} m_2 \bmod p = 763^{30} \cdot 265 \bmod 4271 = 923$

   Sehingga didapatkan *ciphertext* $c_2 = (2076{,}923)$.

3. $m_3 = 787$

   Pilih $k_3 = 335$

   $a_3 = g^{k_3} \bmod p = 7^{335} \bmod 4271 = 3340$

   $b_3 = y^{k_3} m_3 \bmod p = 763^{335} \cdot 787 \bmod 4271 = 2539$

   Sehingga didapatkan *ciphertext* $c_3 = (3340{,}2539)$.

4. $m_4 = 169$

   Pilih $k_4 = 45$

   $a_4 = g^{k_4} \bmod p = 7^{45} \bmod 4271 = 2217$

   $b_4 = y^{k_4} m_4 \bmod p = 763^{45} \cdot 169 \bmod 4271 = 4197$

   Sehingga didapatkan *ciphertext* $c_4 = (2217,4197)$.

5. $m_5 = 328$

   Pilih$k_5 = 82$

   $a_5 = g^{k_5} \bmod p = 7^{82} \bmod 4271 = 3828$

   $b_5 = y^{k_5} m_5 \bmod p = 763^{82} \cdot 328 \bmod 4271 = 1890$

   Sehingga didapatkan *ciphertext* $c_5 = (3828,1890)$.

6. $m_6 = 979$

   Pilih $k_6 = 330$

   $a_6 = g^{k_6} \bmod p = 7^{330} \bmod 4271 = 1021$

   $b_6 = y^{k_6} m_6 \bmod p = 763^{330} \cdot 979 \bmod 4271 = 4203$

   Sehingga didapatkan *ciphertext* $c_6 = (1021,4203)$.

7. $m_7 = 858$

   Pilih $k_7 = 130$

   $a_7 = g^{k_7} \bmod p = 7^{130} \bmod 4271 = 4192$

   $b_7 = y^{k_7} m_7 \bmod p = 763^{130} \cdot 858 \bmod 4271 = 2892$

   Sehingga didapatkan *ciphertext* $c_7 = (4192,2892)$.

8. $m_8 = 232$

   Pilih $k_8 = 22$

   $a_8 = g^{k_8} \bmod p = 7^{22} \bmod 4271 = 3015$

   $b_8 = y^{k_8} m_8 \bmod p = 763^{22} \cdot 232 \bmod 4271 = 1534$

Sehingga didapatkan *ciphertext* $c_8 = (3015,1534)$.

9. $m_9 = 847$

   Pilih $k_9 = 280$

   $a_9 = g^{k_9} \bmod p = 7^{280} \bmod 4271 = 930$

   $b_9 = y^{k_9} m_9 \bmod p = 763^{280} \cdot 847 \bmod 4271 = 1538$

   Sehingga didapatkan *ciphertext* $c_9 = (930,1538)$.

10. $m_{10} = 279$

    Pilih $k_{10} = 273$

    $a_{10} = g^{k_{10}} \bmod p = 7^{273} \bmod 4271 = 3764$

    $b_{10} = y^{k_{10}} m_{10} \bmod p = 763^{273} \cdot 571 \bmod 279 = 409409$

    Sehingga didapatkan *ciphertext* $c_{10} = (3764,3516)$.

11. $m_{11} = 857$

    Pilih $k_{11} = 630$

    $a_{11} = g^{k_{11}} \bmod p = 7^{630} \bmod 4271 = 4158$

    $b_{11} = y^{k_{11}} m_{11} \bmod p = 763^{630} \cdot 857 \bmod 4271 = 2357$

    Sehingga didapatkan *ciphertext* $c_{11} = (4158,2357)$.

12. $m_{12} = 172$

    Pilih $k_{12} = 3070$

    $a_{12} = g^{k_{12}} \bmod p = 7^{3070} \bmod 4271 = 3778$

    $b_{12} = y^{k_{12}} m_{12} \bmod p = 763^{3070} \cdot 172 \bmod 4271 = 781$

    Sehingga didapatkan *ciphertext* $c_{12} = (3778,781)$.

13. $m_{13} = 848$

    Pilih $k_{13} = 2247$

    $a_{13} = g^{k_{13}} \bmod p = 7^{2247} \bmod 4271 = 1403$

$b_{13} = y^{k_{13}} m_{13} \bmod p = 763^{2247} \cdot 848 \bmod 4271 = 1419$

Sehingga didapatkan *ciphertext* $c_{13} = (1403,1419)$.

14. $m_{14} = 332$

    Pilih $k_{14} = 81$

    $a_{14} = g^{k_{14}} \bmod p = 7^{81} \bmod 4271 = 1157$

    $b_{14} = y^{k_{14}} m_{14} \bmod p = 763^{81} \cdot 332 \bmod 4271 = 639$

    Sehingga didapatkan *ciphertext* $c_{14} = (1157,639)$.

15. $m_{15} = 657$

    Pilih $k_{15} = 2434$

    $a_{15} = g^{k_{15}} \bmod p = 7^{2434} \bmod 4271 = 3530$

    $b_{15} = y^{k_{15}} m_{15} \bmod p = 763^{2434} \cdot 657 \bmod 4271 = 49$

    Sehingga didapatkan *ciphertext* $c_{15} = (3530,49)$.

16. $m_{16} = 868$

    Pilih $k_{16} = 4195$

    $a_{16} = g^{k_{16}} \bmod p = 7^{4195} \bmod 4271 = 397$

    $b_{16} = y^{k_{16}} m_{16} \bmod p = 763^{4195} \cdot 868 \bmod 4271 = 2224$

    Sehingga didapatkan *ciphertext* $c_{16} = (397,2224)$.

17. $m_{17} = 328$

    Pilih $k_{17} = 2316$

    $a_{17} = g^{k_{17}} \bmod p = 7^{2316} \bmod 4271 = 2017$

    $b_{17} = y^{k_{17}} m_{17} \bmod p = 763^{2316} \cdot 328 \bmod 4271 = 2543$

    Sehingga didapatkan *ciphertext* $c_{17} = (2017,2543)$.

18. $m_{18} = 979$

    Pilih $k_{18} = 1881$

$a_{18} = g^{k_{18}} \bmod p = 7^{1881} \bmod 4271 = 827$

$b_{18} = y^{k_{18}} m_{18} \bmod p = 763^{1881} \cdot 979 \bmod 4271 = 4013$

Sehingga didapatkan *ciphertext* $c_{18} = (827,4013)$.

19. $m_{19} = 853$

    Pilih $k_{19} = 1623$

    $a_{19} = g^{k_{19}} \bmod p = 7^{1623} \bmod 4271 = 1510$

    $b_{19} = y^{k_{19}} m_{19} \bmod p = 763^{1623} \cdot 853 \bmod 4271 = 3216$

    Sehingga didapatkan *ciphertext* $c_{19} = (1510,3216)$.

20. $m_{20} = 267$

    Pilih $k_{20} = 1047$

    $a_{20} = g^{k_{20}} \bmod p = 7^{1047} \bmod 4271 = 223$

    $b_{20} = y^{k_{20}} m_{20} \bmod p = 763^{1047} \cdot 267 \bmod 4271 = 174$

    Sehingga didapatkan *ciphertext* $c_{20} = (223,174)$.

21. $m_{21} = 726$

    Pilih $k_{21} = 552$

    $a_{21} = g^{k_{21}} \bmod p = 7^{552} \bmod 4271 = 3418$

    $b_{21} = y^{k_{21}} m_{21} \bmod p = 763^{552} \cdot 726 \bmod 4271 = 2022$

    Sehingga didapatkan *ciphertext* $c_{21} = (3418,2022)$.

22. $m_{22} = 578$

    Pilih $k_{22} = 492$

    $a_{22} = g^{k_{22}} \bmod p = 7^{492} \bmod 4271 = 2190$

    $b_{22} = y^{k_{22}} m_{22} \bmod p = 763^{492} \cdot 578 \bmod 4271 = 3607$

    Sehingga didapatkan *ciphertext* $c_{22} = (2190,3607)$.

23. $m_{23} = 716$

    Pilih $k_{23} = 4052$

    $a_{23} = g^{k_{23}} \bmod p = 7^{4052} \bmod 4271 = 601$

    $b_{23} = y^{k_{23}} m_{23} \bmod p = 763^{4052} \cdot 716 \bmod 4271 = 1759$

    Sehingga didapatkan *ciphertext* $c_{23} = (601,1759)$.

24. $m_{24} = 932$

    Pilih $k_{24} = 819$

    $a_{24} = g^{k_{24}} \bmod p = 7^{819} \bmod 4271 = 1451$

    $b_{24} = y^{k_{24}} m_{24} \bmod p = 763^{819} \cdot 932 \bmod 4271 = 1141$

    Sehingga didapatkan *ciphertext* $c_{24} = (1451,1141)$.

25. $m_{25} = 897$

    Pilih $k_{25} = 306$

    $a_{25} = g^{k_{25}} \bmod p = 7^{306} \bmod 4271 = 612$

    $b_{25} = y^{k_{25}} m_{25} \bmod p = 763^{306} \cdot 897 \bmod 4271 = 981$

    Sehingga didapatkan *ciphertext* $c_{25} = (612,981)$.

26. $m_{26} = 985$

    Pilih $k_{26} = 155$

    $a_{26} = g^{k_{26}} \bmod p = 7^{155} \bmod 4271 = 2504$

    $b_{26} = y^{k_{26}} m_{26} \bmod p = 763^{155} \cdot 985 \bmod 4271 = 99$

    Sehingga didapatkan *ciphertext* $c_{26} = (2504,99)$.

27. $m_{27} = 823$

    Pilih $k_{27} = 4125$

    $a_{27} = g^{k_{27}} \bmod p = 7^{4125} \bmod 4271 = 469$

    $b_{27} = y^{k_{27}} m_{27} \bmod p = 763^{4125} \cdot 823 \bmod 4271 = 2099$

Sehingga didapatkan *ciphertext* $c_{27} = (469,2099)$.

28. $m_{28} = 287$

    Pilih $k_{28} = 1819$

    $a_{28} = g^{k_{28}} \bmod p = 7^{1819} \bmod 4271 = 1689$

    $b_{28} = y^{k_{28}} m_{28} \bmod p = 763^{1819} \cdot 287 \bmod 4271 = 395$

    Sehingga didapatkan *ciphertext* $c_{28} = (1689,395)$.

29. $m_{29} = 798$

    Pilih $k_{29} = 3785$

    $a_{29} = g^{k_{29}} \bmod p = 7^{3785} \bmod 4271 = 2964$

    $b_{29} = y^{k_{29}} m_{29} \bmod p = 763^{3785} \cdot 798 \bmod 4271 = 2558$

    Sehingga didapatkan *ciphertext* $c_{29} = (2504,99)$.

30. $m_{30} = 276$

    Pilih $k_{30} = 178$

    $a_{30} = g^{k_{30}} \bmod p = 7^{178} \bmod 4271 = 1837$

    $b_{30} = y^{k_{30}} m_{30} \bmod p = 763^{178} \cdot 276 \bmod 4271 = 1538$

    Sehingga didapatkan *ciphertext* $c_{30} = (1837,1538)$.

31. $m_{31} = 68$

    Pilih $k_{31} = 2987$

    $a_{31} = g^{k_{31}} \bmod p = 7^{2987} \bmod 4271 = 4026$

    $b_{31} = y^{k_{31}} m_{31} \bmod p = 763^{2987} \cdot 68 \bmod 4271 = 1356$

    Sehingga didapatkan *ciphertext* $c_{31} = (4026,1356)$.

Jadi dari proses enkripsi yang telah diuraikan didapatkan *ciphertext c* sebagai berikut:

$c = (3880,320); (2076,923); (3340,2539); (2217,4197); (3828,1890); (1021, 4203); (4192,2892); (3015,1534); (930,1538); (3764,3516); (4158,2357); (3778,781); (1403,1419);(1157,639);(3530,49);(397,2224); (2017, 2543); (827,4013); (1510,3216); (223,174); (3418,2022); (2190,3607); (601,1759); (1451,1141); (612,981); (2504,99););(469,2099); (1689, 395);(2504,99);(1837,1538);(4026,1356).$

4.4 Simulasi Dekripsi Pesan dengan Menggunakan Algoritma Elgamal

4.4.1 Proses Dekripsi Pesan

Ciphertext :

$c = (3880,320); (2076,923); (3340,2539); (2217,4197); (3828,1890); (1021, 4203); (4192,2892); (3015,1534); (930,1538); (3764,3516); (4158,2357); (3778,781); (1403,1419);(1157,639);(3530,49);(397,2224); (2017, 2543); (827,4013); (1510,3216); (223,174); (3418,2022); (2190,3607); (601,1759); (1451,1141); (612,981); (2504,99););(469,2099); (1689, 395);(2504,99);(1837,1538);(4026,1356)$

Untuk mendekripsikan *ciphertext* yang telah diperoleh, maka dilakukan proses dekripsi pada masing-masing nilai *c* dengan cara sebagai berikut:

1. $c_1 = (3880,320)$

$$m_1 = \frac{b_1}{a_1^x} \; mod \; p$$

$$= b_1 (a_1)^{-x} mod \; p$$

$$= b_1 (a_1)^{p-1-x} mod$$

$$= 320 \cdot 3880^{4271-1-101} \; mod \; 4271 = 677$$

2. $c_2 = (2076,923)$

$$m_2 = \frac{b_2}{a_2^x} \; mod \; p$$

$$= b_2(a_2)^{-x} mod \; p$$

$$= b_2(a_2)^{p-1-x} mod$$

$$= 923 \cdot 2076^{4271-1-101} \; mod \; 4271 = 265$$

3. $c_3 = (3340,2539)$

$$m_3 = \frac{b_3}{a_3^x} \; mod \; p$$

$$= b_3(a_3)^{-x} mod \; p$$

$$= b_3(a_3)^{p-1-x} mod$$

$$= 2539 \cdot 3340^{4271-1-101} \; mod \; 4271 = 787$$

4. $c_4 = (2217,4197)$

$$m_4 = \frac{b_4}{a_4^x} \; mod \; p$$

$$= b_4(a_4)^{-x} mod \; p$$

$$= b_4(a_4)^{p-1-x} mod$$

$$= 4197 \cdot 2217^{4271-1-101} \; mod \; 4271 = 169$$

5. $c_5 = (3828,1890)$

$$m_5 = \frac{b_5}{a_5^x} \; mod \; p$$

$$= b_5(a_5)^{-x} mod \; p$$

$$= b_5(a_5)^{p-1-x} mod$$

$$= 1890 \cdot 3828^{4271-1-101} \; mod \; 4271 = 328$$

6. $c_6 = (1021,4203)$

$$m_6 = \frac{b_6}{a_6^x} \; mod \; p$$

$$= b_6(a_6)^{-x} mod \; p$$

$$= b_6(a_6)^{p-1-x} mod$$

$$= 4203 \cdot 1021^{4271-1-101} \; mod \; 4271 = 858$$

7. $c_7 = (4192,2892)$

$$m_7 = \frac{b_7}{a_7^x} \; mod \; p$$

$$= b_7(a_7)^{-x} mod \; p$$

$$= b_7(a_7)^{p-1-x} mod$$

$$= 2892 \cdot 4192^{4271-1-101} \; mod \; 4271 = 858$$

8. $c_8 = (3015,1534)$

$$m_8 = \frac{b_8}{a_8^x} \; mod \; p$$

$$= b_8(a_8)^{-x} mod \; p$$

$$= b_8(a_8)^{p-1-x} mod$$

$$= 1534 \cdot 3015^{4271-1-101} \; mod \; 4271 = 232$$

9. $c_9 = (930,1538)$

$$m_9 = \frac{b_9}{a_9^x} \; mod \; p$$

$$= b_9(a_9)^{-x} mod \; p$$

$$= b_9(a_9)^{p-1-x} mod$$

$$= 1538 \cdot 930^{4271-1-101} \; mod \; 4271 = 847$$

10. $c_{10} = (3764,409)$

$$m_{10} = \frac{b_{10}}{a_{10}^{x}} \; mod \; p$$

$$= b_{10}(a_{10})^{-x} mod \; p$$

$$= b_{10}(a_{10})^{p-1-x} mod$$

$$= 409 \cdot 3764^{4271-1-101} \; mod \; 4271 = 279$$

11. $c_{11} = (4158,2357)$

$$m_{11} = \frac{b_{11}}{a_{11}^{x}} \; mod \; p$$

$$= b_{11}(a_{11})^{-x} mod \; p$$

$$= b_{11}(a_{11})^{p-1-x} mod$$

$$= 2357 \cdot 4158^{4271-1-101} \; mod \; 4271 = 857$$

12. $c_{12} = (3778,781)$

$$m_{12} = \frac{b_{12}}{a_{12}^{x}} \; mod \; p$$

$$= b_{12}(a_{12})^{-x} mod \; p$$

$$= b_{12}(a_{12})^{p-1-x} mod$$

$$= 781 \cdot 3778^{4271-1-101} \; mod \; 4271 = 172$$

13. $c_{13} = (1403,1419)$

$$m_{13} = \frac{b_{13}}{a_{13}^{x}} \; mod \; p$$

$$= b_{13}(a_{13})^{-x} mod \; p$$

$$= b_{13}(a_{13})^{p-1-x} mod$$

$$= 1419 \cdot 1403^{4271-1-101} \; mod \; 4271 = 848$$

14. $c_{14} = (1157,639)$

$$m_{14} = \frac{b_{14}}{a_{14}^{x}} \; mod \; p$$

$$= b_{14}(a_{14})^{-x} mod \; p$$

$$= b_{14}(a_{14})^{p-1-x} mod$$

$$= 639 \cdot 1157^{4271-1-101} \; mod \; 4271 = 332$$

15. $c_{15} = (3530,49)$

$$m_{15} = \frac{b_{15}}{a_{15}^{x}} \; mod \; p$$

$$= b_{15}(a_{15})^{-x} mod \; p$$

$$= b_{15}(a_{15})^{p-1-x} mod$$

$$= 49 \cdot 3530^{4271-1-101} \; mod \; 4271 = 657$$

16. $c_{16} = (397,2224)$

$$m_{16} = \frac{b_{16}}{a_{16}^{x}} \; mod \; p$$

$$= b_{16}(a_{16})^{-x} mod \; p$$

$$= b_{16}(a_{16})^{p-1-x} mod$$

$$= 2224 \cdot 397^{4271-1-101} \; mod \; 4271 = 868$$

17. $c_{17} = (2017,2543)$

$$m_{17} = \frac{b_{17}}{a_{17}^{x}} \; mod \; p$$

$$= b_{17}(a_{17})^{-x} mod \; p$$

$$= b_{17}(a_{17})^{p-1-x} mod$$

$$= 2543 \cdot 2017^{4271-1-101} \; mod \; 4271 = 328$$

18. $c_{18} = (827,4013)$

$$m_{18} = \frac{b_{18}}{a_{18}^{x}} \; mod \; p$$

$$= b_{18}(a_{18})^{-x} mod \; p$$

$$= b_{18}(a_{18})^{p-1-x} mod$$

$$= 4013 \cdot 827^{4271-1-101} \; mod \; 4271 = 979$$

19. $c_{19} = (1510,3216)$

$$m_{19} = \frac{b_{19}}{a_{19}^{x}} \; mod \; p$$

$$= b_{19}(a_{19})^{-x} mod \; p$$

$$= b_{19}(a_{19})^{p-1-x} mod$$

$$= 3216 \cdot 1510^{4271-1-101} \; mod \; 4271 = 853$$

20. $c_{20} = (223,174)$

$$m_{20} = \frac{b_{20}}{a_{20}^{x}} \; mod \; p$$

$$= b_{20}(a_{20})^{-x} mod \; p$$

$$= b_{20}(a_{20})^{p-1-x} mod$$

$$= 174 \cdot 223^{4271-1-101} \; mod \; 4271 = 267$$

21. $c_{21} = (3418,2022)$

$$m_{21} = \frac{b_{21}}{a_{21}^{x}} \; mod \; p$$

$$= b_{21}(a_{21})^{-x} mod \; p$$

$$= b_{21}(a_{21})^{p-1-x} mod$$

$$= 2022 \cdot 3418^{4271-1-101} \; mod \; 4271 = 726$$

22. $c_{22} = (2190,3607)$

$$m_{22} = \frac{b_{22}}{a_{22}^{x}} \; mod \; p$$

$$= b_{22}(a_{22})^{-x} mod \; p$$

$$= b_{22}(a_{22})^{p-1-x} mod$$

$$= 3607 \cdot 2190^{4271-1-101} \; mod \; 4271 = 578$$

23. $c_{23} = (601,1759)$

$$m_{23} = \frac{b_{23}}{a_{23}^{x}} \; mod \; p$$

$$= b_{23}(a_{23})^{-x} mod \; p$$

$$= b_{23}(a_{23})^{p-1-x} mod$$

$$= 1759 \cdot 601^{4271-1-101} \; mod \; 4271 = 716$$

24. $c_{24} = (1451,1141)$

$$m_{24} = \frac{b_{24}}{a_{24}^{x}} \; mod \; p$$

$$= b_{24}(a_{24})^{-x} mod \; p$$

$$= b_{24}(a_{24})^{p-1-x} mod$$

$$= 1141 \cdot 1451^{4271-1-101} \; mod \; 4271 = 932$$

25. $c_{25} = (612,981)$

$$m_{25} = \frac{b_{25}}{a_{25}^{x}} \; mod \; p$$

$$= b_{25}(a_{25})^{-x} mod \; p$$

$$= b_{25}(a_{25})^{p-1-x} mod$$

$$= 612 \cdot 981^{4271-1-101} \; mod \; 4271 = 897$$

26. $c_{26} = (2504,99)$

$$m_{26} = \frac{b_{26}}{a_{26}^{x}} \; mod \; p$$

$$= b_{26}(a_{26})^{-x} mod \; p$$

$$= b_{26}(a_{26})^{p-1-x} mod$$

$$= 99 \cdot 2504^{4271-1-101} \; mod \; 4271 = 985$$

27. $c_{27} = (469,2099)$

$$m_{27} = \frac{b_{27}}{a_{27}^{x}} \; mod \; p$$

$$= b_{27}(a_{27})^{-x} mod \; p$$

$$= b_{27}(a_{27})^{p-1-x} mod$$

$$= 2099 \cdot 469^{4271-1-101} \; mod \; 4271 = 823$$

28. $c_{28} = (1689,395)$

$$m_{28} = \frac{b_{28}}{a_{28}^{x}} \; mod \; p$$

$$= b_{28}(a_{28})^{-x} mod \; p$$

$$= b_{28}(a_{28})^{p-1-x} mod$$

$$= 395 \cdot 1689^{4271-1-101} \; mod \; 4271 = 287$$

29. $c_{29} = (2964,2558)$

$$m_{29} = \frac{b_{29}}{a_{29}^{x}} \; mod \; p$$

$$= b_{29}(a_{29})^{-x} mod \; p$$

$$= b_{29}(a_{29})^{p-1-x} mod$$

$$= 2558 \cdot 2964^{4271-1-101} \; mod \; 4271 = 798$$

30. $c_{30} = (1837,1538)$

$$m_{30} = \frac{b_{30}}{a_{30}^{x}} \; mod \; p$$

$$= b_{30}(a_{30})^{-x} mod \; p$$

$$= b_{30}(a_{30})^{p-1-x} mod$$

$$= 1538 \cdot 1837^{4271-1-101} \; mod \; 4271 = 276$$

31. $c_{31} = (4026,1356)$

$$m_{31} = \frac{b_{31}}{a_{31}^{x}} \; mod \; p$$

$$= b_{31}(a_{31})^{-x} mod \; p$$

$$= b_{31}(a_{31})^{p-1-x} mod$$

$$= 1356 \cdot 4026^{4271-1-101} \; mod \; 4271 = 68$$

Dari penghitungan pada proses dekripsi, didapatkan masing-masing nilai m sebagai berikut:

$m_1 = 677, m_2 = 265, m_3 = 787, m_4 = 169, m_5 = 328, m_6 = 979, m_7 = 858,$
$m_8 = 232, \quad m_9 = 847, \quad m_{10} = 279, \quad m_{11} = 857, \quad m_{12} = 172, \quad m_{13} = 848,$
$m_{14} = 332, \quad m_{15} = 657, \quad m_{16} = 868, \quad m_{17} = 328, \quad m_{18} = 979, \quad m_{19} = 853,$
$m_{20} = 267, \quad m_{21} = 726, \quad m_{22} = 578, \quad m_{23} = 716, \quad m_{24} = 932, \quad m_{25} = 897,$
$m_{26} = 985, m_{27} = 823, m_{28} = 287, m_{29} = 798, m_{30} = 276, m_{31} = 68.$

Kemudian blok-blok m tersebut digabungkan dan didapatkan nilai

$m = 677265787169328979858232847279857172848332657868328979853267726578716932897985823287798276 68$

Setelah itu bagi nilai m menjadi beberapa bagian dengan syarat untuk masing-masing m harus berada pada interval $[0, 255]$. Kemudian konversi ke dalam karakter yang sesuai dengan menggunakan tabel ASCII 256, sehingga

menghasilkan *plaintext* “CHANGE YOUR THOUHGTS AND YOU CHANGE YOUR WORLD” yang merupakan *plaintext* yang sama dengan *plaintext* pada proses enkripsi.

4.5 Kajian Tentang Amanah dan Hubungannya dengan Keamanan Pesan

Pengiriman informasi rahasia merupakan aktivitas yang harus diperhatikan keamanannya. Keamanan diperlukan agar informasi tidak diketahui oleh pihak yang tidak memiliki hak atas informasi tersebut. Terlebih pada zaman modern saat ini informasi rawan untuk dicuri oleh pihak-pihak yang tidak bertanggung jawab, karena itu diperlukan tingkat keamanan yang tinggi untuk melindunginya. Informasi sama halnya dengan sebuah amanah yang harus dijaga dengan bersungguh-sungguh. Salah satu cara untuk menjaga keamanan informasi adalah menggunakan bantuan ilmu kriptografi. Dalam kriptografi terdapat algoritma-algoritma yang dapat digunakan untuk menyandikan pesan sehingga pesan tersebut menjadi samar. Dengan bantuan kriptografi pesan akan dienkripsi menggunakan sebuah algoritma dimana untuk memecahkan algoritma tersebut dibutuhkan sebuah kunci agar pesan dapat kembali kedalam bentuk semula. Kunci hanya dimiliki oleh pengirim dan penerima pesan sehingga orang lain akan kesulitan memecahkan isi pesan jika tidak memiliki kuncinya. Menjaga keamanan informasi ini berkaitan dengan adanya perintah untuk amanah. Orang yang memiliki sifat amanah merupakan orang-orang yang dapat dipercaya dan segan untuk menyebarkan informasi yang dimiliki terlebih jika bersifat rahasia. Selain itu masalah ini juga berkaitan dengan sifat tanggung jawab bagi manusia. Menjaga keamanan informasi merupakan sebuah tanggung jawab yang diberikan

kepada manusia yang telah dipilih agar informasi tidak jatuh ke tangan orang yang salah.

BAB V
PENUTUP

5.1 Kesimpulan

Berdasarkan pembahasan dari proses modifikasi kunci pada algoritma Elgamal, proses enkripsi serta dekripsinya maka:

1. Modifikasi pembangkit kunci pada algoritma Elgamal dilakukan dengan menggunakan bantuan algoritma DNA dengan cara menentukan bilangan prima p dan bilangan bulat g yang merupakan akar primitif dari p. Dilanjutkan dengan menentukan bilangan bulat acak z yang akan digunakan untuk pembangkitan kunci. Kemudian mengenkripsi z menggunakan algoritma DNA sehingga didapatkan bilangan lain yang kemudian digunakan sebagai kunci privat x. Kemudian menentukan kunci publik y menggunakan persamaan yang telah ditentukan. Selanjutnya enkripsi g menggunakan algoritma Elgamal sehingga mendapatkan nilai q. Sehingga didapatkan kunci privat x serta kunci publik (y, q, p).
2. Dekripsi q menggunakan algoritma DNA untuk mendapatkan g dimana g merupakan kunci publik yang digunakan pada proses enkripsi. Enkripsi *plaintext* dengan menggunakan algoritma Elgamal diawali dengan mengubah *plaintext* ke dalam nilai desimal sehingga didapatkan sebuah nilai m. Jika nilai m terlalu besar, maka m dibagi menjadi blok-blok yang lebih kecil yang berada dalam interval $[0, p-1]$. Setelah itu dilakukan enkripsi pada m dengan menentukan bilangan bulat acak k terlebih dahulu, kemudian menghitung nilai a dan b. Nilai a dan b ini yang kemudian

menjadi *ciphertext* bagi m. *Ciphertext* pada algoritma Elgamal lebih panjang daripada *plaintext* yang diberikan.

3. Proses dekripsi *ciphertext* pada algoritma Elgamal diawali dengan menghitung nilai m dengan menggunakan *ciphertext* a dan b yang telah diperoleh. Setelah didapatkan nilai m kemudian dilanjutkan dengan menggabungkan nilai m yang terpisah. Langkah selanjutnya adalah mengubah nilai m menjadi karakter yang sesuai sehingga didapatkan teks yang sesuai dengan *plaintext* yang di*input*kan.

5.2 Saran untuk Penelitian Lanjutan

Berdasarkan hasil penelitian penulis memberikan saran terhadap pengembangan penelitian selanjutnya sebagai berikut:

1. Memodifikasi pembangkit kunci pada algoritma Elgamal dengan menggunakan algoritma lain.
2. Mengkombinasikan algoritma Elgamal dengan algoritma lain atau algoritma DNA dengan algoritma lain agar didapatkan *ciphertext* yang lebih sulit dipecahkan untuk mendapatkan tingkat keamanan yang lebih tinggi.

DAFTAR PUSTAKA

Ahmed, R. K. dan Mohammed, I. J. (2017). *Developing a New Hybrid Cipher Algorithm using DNA and RC4*. International Journal of Advanced Computer Science and Applications, 8(10), 171–176. www.ijacsa.thesai.org

Al-Qahira Mushaf Terjemah Tajwid Warna. (2017). UD. Nur Ilmu

Batten, L. M. (2012). *Public Key Cryptography Applications and Attacks*. IEEE Press.

ElGamal, T. (1985). A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. *IEE Transactions on Information Theory, IT-31*(4), 469–472. https://doi.org/10.1007/3-540-39568-7_2

Gaffar, S. (2007). *Buku Ajar Bioteknologi Molekul*.

Harahap, N. (2020). *Penelitian Kualitatif* (H. Sazali (ed.)). Wal Ashri Publishing.

Jamaludin, Sulaiman, O. K., Tandungan, S., Putra, L. M., Yuswardi, Yulianti, N., Sidabutar, J., Aisa, S., Tantriawan, H., Arizal, Mardalius, dan Pakpahan, A. F. (2022). *Kriptografi: Teknik Keamanan Data*. Yayasan Kita Menulis.

Kitab Hadist Online. (2011). Kementrian Agama Aceh

Mahesa, K., Sugiantoro, B., dan Prayudi, Y. (2019). *Pemanfaatan Metode DNA Kriptografi dalam Meningkatkan Keamanan Citra Digital*. Jurnal Ilmiah Informatika, 7(02), 108–113.

Menezes, A. J., Oorschoot, P. C., dan Vanstone, S. A., (1996), *Handbook of Applied Cryptography, CRC Press*.

Munir, R. (2019). *Kriptografi* (2nd ed.). Informatika Bandung.

Raj, B. B., Vijay, J. F., dan Mahalakshmi, T. (2016). *Secure Data Transfer through DNA Cryptography using Symmetric Algorithm*. International Journal of Computer Applications, 133(2), 19–23. https://doi.org/10.5120/ijca2016907719

Sadikin, R. (2012). *Kriptografi untuk Keamanan Jaringan dan Implementasinya dalam Bahasa Java* (T. A. Prabawati (ed.)). CV. Andi Offset.

Satir, E. dan Kendirli, O. (2022). *A Symmetric DNA Encryption Process with a Biotechnical Hardware*. Journal of King Saud University - Science, 34(3), 1–10. https://doi.org/10.1016/j.jksus.2022.101838

Shihab, M. Q. (2002). *Tafsir Al-Misbah Pesan, Kesan dan Keserasian Al-Qur'an*

(1st ed., Vol. 13, Issue 1). Penerbit Lentera Hati.

Solin, R. dan Ramadhani, P. (2020). *Modifikasi Pembangkit Kunci Algoritma Elgamal dengan Menerapkan Algoritma Freivalds.* KOMIK (Konferensi Nasional Teknologi Informasi Dan Komputer), *4*(1), 351–356. https://doi.org/10.30865/komik.v4i1.2719

Tantoni, A. dan Zaen, M. T. A. (2018). *Implementasi Double Caesar Cipher Menggunakan ASCII.* Jurnal Informatika Dan Rekayasa Elektronik, 1(2), 24–32. https://doi.org/10.36595/jire.v1i2.56

Warnilah, A. I. dan Nugraha, S. N. (2018). *Komparasi Algoritma Kriptografi Elgamal dan Caesar Cipher untuk Enkripsi dan Dekripsi Pesan.* IJCIT (Indonesian Journal on Computer and Information Technology), 3(2), 243–252. https://repository.bsi.ac.id/index.php/unduh/item/231067/1.-Komparasi-Algoritma-Kriptografi-Elgamal-Dan-Caesar.pdf

LAMPIRAN-LAMPIRAN

Lampiran 1 Tabel ASCII 256

Kode ASCII	Kode Biner	Karakter
00	00000000	NULL
01	00000001	SOH
02	00000010	STX
03	00000011	ETX
04	00000100	EOT
05	00000101	ENQ
06	00000110	ACK
07	00000111	BEL
08	00001000	BS
09	00001000	HT
10	00001010	LF
11	00001011	VT
12	00001100	FF
13	00001101	CR
14	00001110	SOH
15	00001111	SI
16	00010000	DLE
17	00010001	DC1
18	00010010	DC2
19	00010011	DC3
20	00010100	DC4
21	00010101	NAK
22	00010110	SYN
23	00010111	ETB
24	00011000	CAN
25	00011001	EM
26	00011010	SUB
27	00011011	ESC
28	00011100	FS

29	00011101	GS
30	00011110	RS
31	00011111	US
32	00100000	Space
33	00100001	!
34	00100010	"
35	00100011	#
36	00100100	$
37	00100101	%
38	00100110	&
39	00100111	'
40	00101000	(
41	00101001	)
42	00101010	*
43	00101011	+
44	00101100	,
45	00101101	-
46	00101110	.
47	00101111	/
48	00110000	0
49	00110001	1
50	00110010	2
51	00110011	3
52	00110100	4
53	00110101	5
54	00110110	6
55	00110111	7
56	00111000	8
57	00111001	9
58	00111010	:
59	00111011	;
60	00111100	<
61	00111101	=

62	00111110	>
63	00111111	?
64	01000000	@
65	01000001	A
66	01000010	B
67	01000011	CAN
68	01000100	D
69	01000101	E
70	01000110	FS
71	01000111	GS
72	01001000	H
73	01001001	I
74	01001010	J
75	01001011	K
76	01001100	L
77	01001101	M
78	01001110	N
79	01001111	O
80	01010000	P
81	01010001	Q
82	01010010	R
83	01010011	S
84	01010100	T
85	01010101	U
86	01010110	V
87	01010111	W
88	01011000	X
89	01011001	Y
90	01011010	Z
91	01011011	[
92	01011100	\
93	01011101	]
94	01011110	^

95	01011111	_
96	01100000	`
97	01100001	A
98	01100010	B
99	01100011	CAN
100	01100100	D
101	01100101	E
102	01100110	FS
103	01100111	GS
104	01101000	H
105	01101001	I
106	01101010	J
107	01101011	K
108	01101100	L
109	01101101	M
110	01101110	N
111	01101111	O
112	01110000	P
113	01110001	Q
114	01110010	R
115	01110011	S
116	01110100	T
117	01110101	U
118	01110110	V
119	01110111	W
120	01111000	X
121	01111001	Y
122	01111010	Z
123	01111011	{
124	01111100	|
125	01111101	}
126	01111110	~
127	01111111	DEL

128	10000000	Ç
129	10000001	Ü
130	10000010	È
131	10000011	Â
132	10000100	Ä
133	10000101	À
134	10000110	Å
135	10000111	Ç
136	10001000	Ê
137	10001001	Ë
138	10001010	È
139	10001011	Ï
140	10001100	Î
141	10001101	ì
142	10001110	Ä
143	10001111	Å
144	10010000	É
145	10010001	æ
146	10010010	Æ
147	10010011	ô
148	10010100	ö
149	10010101	ò
150	10010110	û
151	10010111	ù
152	10011000	ÿ
153	10011001	Ö
154	10011010	Ü
155	10011011	ø
156	10011100	£
157	10011101	Ø
158	10011110	×
159	10011111	*ƒ*
160	10100000	á

161	10100001	í
162	10100010	ó
163	10100011	ú
164	10100100	Ñ
165	10100101	Ñ
166	10100110	ª
167	10100111	º
168	10101000	¿
169	10101001	®
170	10101010	¬
171	10101011	½
172	10101100	¼
173	10101101	¡
174	10101110	«
175	10101111	»
176	10110000	░
177	10110001	▒
178	10110010	▓
179	10110011	│
180	10110100	┤
181	10110101	Á
182	10110110	Â
183	10110111	À
184	10111000	©
185	10111001	╣
186	10111010	║
187	10111011	╗
188	10111100	╝
189	10111101	¢
190	10111110	¥
191	10111111	┐
192	11000000	└
193	11000001	┴

194	11000010	┬
195	11000011	├
196	11000100	─
197	11000101	┼
198	11000110	Ã
199	11000111	ã
200	11001000	╚
201	11001001	╔
202	11001010	╩
203	11001011	╦
204	11001100	╠
205	11001101	═
206	11001110	╬
207	11001111	¤
208	11010000	Đ
209	11010001	đ
210	11010010	Ê
211	11010011	Ë
212	11010100	È
213	11010101	ı
214	11010110	Í
215	11010111	Î
216	11011000	Ï
217	11011001	┘
218	11011010	┌
219	11011011	█
220	11011100	▄
221	11011101	¦
222	11011110	Ì
223	11011111	▀
224	11100000	Ó
225	11100001	ß
226	11100010	Ô

227	11100011	Ò
228	11100100	Õ
229	11100101	Õ
230	11100110	µ
231	11100111	Þ
232	11101000	Þ
233	11101001	Ú
234	11101010	Û
235	11101011	Ù
236	11101100	Ý
237	11101101	Ý
238	11101110	¯
239	11101111	'
240	11110000	≡
241	11110001	±
242	11110010	‗
243	11110011	¾
244	11110100	¶
245	11110101	§
246	11110110	÷
247	11110111	¸
248	11111000	°
249	11111001	¨
250	11111010	·
251	11111011	¹
252	11111100	³
253	11111101	²
254	11111110	▪
255	11111111	nsbp

Lampiran 2 Tabel Kunci Pembangun Acak pada Kriptografi DNA

0	AAAA	43	GCGG	86	CATC	129	ATAA	172	GGCG	215	CTTC
1	ACAA	44	GACG	87	CCTC	130	AGGA	173	GTCG	216	CGAT
2	AAGA	45	GCCG	88	CAAT	131	ATGA	174	GGTG	217	CTAT
3	ACGA	46	GATG	89	CCAT	132	AGCA	175	GTTG	218	CGGT
4	AACA	47	GCTG	90	CAGT	133	ATCA	176	GGAC	219	CTGT
5	ACCA	48	GAAC	91	CCGT	134	AGTA	177	GTAC	220	CGCT
6	AATA	49	GCAC	92	CACT	135	ATTA	178	GGGC	221	CTCT
7	ACTA	50	GAGC	93	CCCT	136	AGAG	179	GTGC	222	CGTT
8	AAAG	51	GCGC	94	CATT	137	ATAG	180	GGCC	223	CTTT
9	ACAG	52	GACC	95	CCTT	138	AGGG	181	GTCC	224	TGAA
10	AAGG	53	GCCC	96	TAAA	139	ATGG	182	GGTC	225	TTAA
11	ACGG	54	GATC	97	TCAA	140	AGCG	183	GTTC	226	TGGA
12	AACG	55	GCTC	98	TAGA	141	ATCG	184	GGAT	227	TTGA
13	ACCG	56	GAAT	99	TCGA	142	AGTG	185	GTAT	228	TGCA
14	AATG	57	GCAT	100	TACA	143	ATTG	186	GGGT	229	TTCA
15	ACTG	58	GAGT	101	TCCA	144	AGAC	187	GTGT	230	TGTA
16	AAAC	59	GCGT	102	TATA	145	ATAC	188	GGCT	231	TTTA
17	ACAC	60	GACT	103	TCTA	146	AGGC	189	GTCT	232	TGAG
18	AAGC	61	GCCT	104	TAAG	147	ATGC	190	GGTT	233	TTAG
19	ACGC	62	GATT	105	TCAG	148	AGCC	191	GTTT	234	TGGG
20	AACC	63	GCTT	106	TAGG	149	ATCC	192	CGAA	235	TTGG
21	ACCC	64	CAAA	107	TCGG	150	AGTC	193	CTAA	236	TGCG
22	AATC	65	CCAA	108	TACG	151	ATTC	194	CGGA	237	TTCG
23	ACTC	66	CAGA	109	TCCG	152	AGAT	195	CTGA	238	TGTG
24	AAAT	67	CCGA	110	TATG	153	ATAT	196	CGCA	239	TTTG
25	ACAT	68	CACA	111	TCTG	154	AGGT	197	CTCA	240	TGAC
26	AAGT	69	CCCA	112	TAAC	155	ATGT	198	CGTA	241	TTAC
27	ACGT	70	CATA	113	TCAC	156	AGCT	199	CTTA	242	TGGC
28	AACT	71	CCTA	114	TAGC	157	ATCT	200	CGAG	243	TTGC
29	ACCT	72	CAAG	115	TCGC	158	AGTT	201	CTAG	244	TGCC
30	AATT	73	CCAG	116	TACC	159	ATTT	202	CGGG	245	TTCC
31	ACTT	74	CAGG	117	TCCC	160	GGAA	203	CTGG	246	TGTC
32	GAAA	75	CCGG	118	TATC	161	GTAA	204	CGCG	247	TTTC
33	GCAA	76	CACG	119	TCTC	162	GGGA	205	CTCG	248	TGAT
34	GAGA	77	CCCG	120	TAAT	163	GTGA	206	CGTG	249	TTAT

35	GCGA	78	CATG	121	TCAT	164	GGCA	207	CTTG	250	TGGT
36	GACA	79	CCTG	122	TAGT	165	GTCA	208	CGAC	251	TTGT
37	GCCA	80	CAAC	123	TCGT	166	GGTA	209	CTAC	252	TGCT
38	GATA	81	CCAC	124	TACT	167	GTTA	210	CGGC	253	TTCT
39	GCTA	82	CAGC	125	TCCT	168	GGAG	211	CTGC	254	TGTT
40	GAAG	83	CCGC	126	TATT	169	GTAG	212	CGCC	255	TTTT
41	GCAG	84	CACC	127	TCTT	170	GGGG	213	CTCC		
42	GAGG	85	CCCC	128	AGAA	171	GTGG	214	CGTC		

RIWAYAT HIDUP

Ihda Umdatul Khoiroh, lahir di Malang pada 25 Agustus 2000. Bertempat tinggal di Desa Pandansari Lor, Kecamatan Jabung, Kabupaten Malang. Merupakan anak pertama dari Bapak Kusaini dan Ibu Mujayanah. Pendidikan yang pernah ditempuh yaitu TK An-Nur Tumpang yang lulus pada tahun 2007, kemudian melanjutkan pendidikan sekolah dasar di MI Islamiyah Sukopuro selama 6 tahun dan lulus pada tahun 2013. Kemudian melanjutkan pendidikan sekolah menengah pertama di MTs Al-Ittihad Poncokusumo lulus pada tahun 2016. Dan menempuh pendidikan sekolah menengah atas di MA Al-Ittihad Poncokusumo lulus pada tahun 2019. Selain pendidikan formal penulis juga menempuh pendidikan non formal di Pondok Pesantren Putri Al-ittihad Poncokusumo pada tahun 2013 hingga tahun 2019.

Pada tahun yang sama penulis melanjutkan studi di Universitas Islam Negeri Maulana Malik Ibrahim Malang dengan mengambil Program Studi Matematika, Fakultas Sains dan Teknologi. Aktif mengikuti organisasi KSR PMI Unit UIN Malang pada 2020-2022. Kegiatan-kegiatan yang pernah diikuti selama menjadi mahasiswa yaitu Panitia Kompetisi Matematika (KOMET) XIX pada tahun 2020, Panitia KOMET XX pada tahun 2021 dan Panitia BARAPAMERA XVI pada tahun 2023.

KEMENTERIAN AGAMA RI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
FAKULTAS SAINS DAN TEKNOLOGI
Jl. Gajayana No.50 Dinoyo Malang Telp. / Fax. (0341)558933

BUKTI KONSULTASI SKRIPSI

Nama : Ihda Umdatul Khoiroh
NIM : 19610002
Fakultas / Program Studi : Sains dan Teknologi / Matematika
Judul Skripsi : Modifikasi Pembangkit Kunci Algoritma Elgamal dengan Menggunakan Algoritma DNA
Pembimbing I : Prof. Dr. H. Turmudi, M.Si., Ph.D.
Pembimbing II : Dr. H. Imam Sujarwo, M.Pd.

No	Tanggal	Hal	Tanda Tangan
1.	07 Desember 2022	Konsultasi Judul	1.
2.	25 Januari 2023	Konsultasi Revisi Judul dan Konsultasi BAB I	2.
3.	31 Januari 2023	Konsultasi Revisi BAB I dan Konsultasi BAB II, III	3.
4.	07 Februari 2023	Konsultasi Revisi BAB I, II dan III	4.
5.	10 Februari 2023	Konsultasi Revisi BAB III	5.
6.	01 Maret 2023	Konsultasi Kajian Agama BAB I dan II	6.
7.	07 Maret 2023	Konsultasi Revisi Kajian Agama BAB I dan II	7.
8.	08 Maret 2023	ACC Seminar Proposal	8.
9.	16 Mei 2023	Konsultasi Revisi Seminar Proposal dan Konsultasi BAB IV	9.
10.	18 Mei 2023	Konsultasi Revisi BAB IV	10.
11.	20 Mei 2023	Konsultasi BAB V	11.
12.	22 Mei 2023	Konsultasi Revisi BAB V	12.
13.	24 Mei 2023	Konsultasi Kajian Agama BAB IV	13.
14.	25 Mei 2023	Konsultasi Revisi BAB V	14.
15.	26 Mei 2023	ACC Seminar Hasil	15.

KEMENTERIAN AGAMA RI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
FAKULTAS SAINS DAN TEKNOLOGI
Jl. Gajayana No 50 Dinoyo Malang Telp. / Fax. (0341)558933

16.	21 Juni 2023	Konsultasi Revisi Kajian Agama BAB IV	16.
17.	21 Juni 2023	Konsultasi BAB IV dan ACC Sidang Skripsi	17.
18.	26 Juni 2023	ACC Keseluruhan	18.

Malang, 26 Juni 2023
Mengetahui,
Ketua Program Studi Matematika

Dr. Elly Susanti, M.Sc.
NIP. 19741129 200012 2 005