IMPLEMENTASI ALGORITMA SUPER ENKRIPSI PADA PENGAMAN PESAN BERBENTUK TEKS

SKRIPSI

OLEH
AHMAD ZAINI
NIM. 14610098

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2021

IMPLEMENTASI ALGORITMA SUPER ENKRIPSI PADA PENGAMAN PESAN BERBENTUK TEKS

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:
Ahmad Zaini
NIM. 14610098

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2021

IMPLEMENTASI ALGORITMA SUPER ENKRIPSI PADA PENGAMAN PESAN BERBENTUK TEKS

SKRIPSI

Oleh
Ahmad Zaini
NIM. 14610098

Telah diperiksa dan disetujui untuk diuji
Tanggal 7 Mei 2021

Pembimbing I,	Pembimbing II,
Muhammad Khudzaifah, M.Si	Juhari, M.Si
NIDT. 19900511 20160801 1 057	NIDT. 19840209 20160801 1 055

Mengetahui
Ketua Program Studi Matematika

Dr. Usman Pagalay, M.Si
NIP. 19650414 200312 1 001

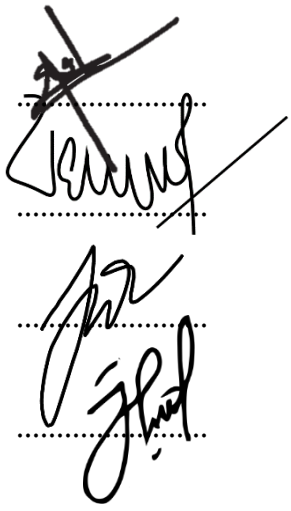
IMPLEMENTASI ALGORITMA SUPER ENKRIPSI PADA PENGAMAN PESAN BERBENTUK TEKS

SKRIPSI

Oleh
Ahmad Zaini
NIM. 14610098

Telah Dipertahankan di Depan Dewan Penguji Skripsi dan
Dinyatakan Diterima Sebagai Salah satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S.Mat)
Tanggal 15 Juni 2021

Penguji Utama	: Muhammad Nafie Jauhari, M.Si	.....
Ketua Penguji	: Evawati Alisah, M.Pd	.....
Sekretaris Penguji	: Muhammad Khudzaifah, M.Si	.....
Anggota Penguji	: Juhari, M.Si	.....



Mengetahui
Ketua Program Studi Matematika

Dr. Usman Pagalay, M.Si
NIP. 19650414 200312 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama : Ahmad Zaini

Nim : 14610098

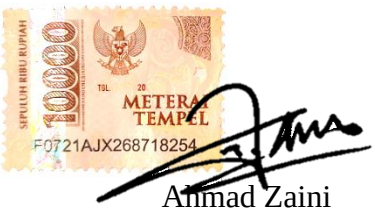
Program Studi : Matematika

Fakultas : Sains Dan Teknologi

Judul Skripsi : Implementasi Algoritma Super Enkripsi Pada Pengaman Pesan Berbentuk Teks

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 atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar rujukan. Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 15 Juni 2021
Yang membuat pernyataan,



Ahmad Zaini
NIM.14610098

MOTTO

“ **Memulai** Dengan Penuh **Keyakinan**”

“**Menjalankan** Dengan Penuh **Keikhlasan**”

“**Menyelesaikan** Dengan Penuh **Kebahagiaan** “

PERSEMBAHAN

Skripsi ini penulis persembahan untuk:

Bapak Abd. Wahab dan Ibu Yama, dan saudara perempuan Isnaini
Yang tidak lelah menyemangati, mengingatkan dan memberikan dukungan yang sebanyak-banyaknya sehingga penulis bisa menyelesaikan tugas akhir ini.

KATA PENGANTAR

Assalamua'alaikum Warahmatullahi Wabarakatuh

Segala puji bagi Allah Swt atas rahmat, taufik serta hidayah-Nya. Sehingga penulis mampu menyelesaikan penyusunan skripsi ini sebagai salah satu syarat untuk memperoleh gelar sarjana dalam bidang matematika di Fakultas Sains dan Teknologi, Universitas islam Negeri Maulana Malik Ibrahim Malang.

Dalam proses penyusunan skripsi ini, penulis banyak mendapat bimbingan dan arahan dari berbagai pihak. Untuk itu penulis ucapan terimakasih yang sebesar-besarnya dan penghargaan yang setinggi-tingginya sampaikan terutama kepada:

1. Prof. Dr. H. Abd. Haris, M.Ag, 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. Usman Pagalay, M.Si, Selaku Ketua Program Studi Matematika, Fakultas Sains Dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Muhammad Khudzaifah, M.Si, sebagai pembimbing satu skripsi atas arahan, nasihat, dan ilmu yang diberikan untuk penulis.
5. Juhari, M.Si, sebagai pembimbing dua skripsi atas saran dan arahan untuk penulis.
6. M. Nafie Jauhari, M.Si, sebagai penguji utama skrispi atas saran dan kritik untuk penulis.

7. Evawati Alisah, M.Pd, sebagai ketua penguji skripsi atas kritik, saran dan dukungan untuk penulis.
8. Segenap sivitas akademika Program Studi Matematika, Fakultas Sains dan teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang, terutama seluruh Dosen terimakasih atas segala ilmu dan bimbingannya.
9. Bapak Abd. Wahab dan Ibu Yama yang selalu memberikan do'a, semangat serta motivasi kepada penulis sampai saat ini.
10. Sahabat-sahabat terbaik penulis yang selalu menemani, membantu, dan memberikan dukungan sehingga penulis dapat menyelesaikan skripsi ini.
11. Seluruh teman-teman di program studi matematika angkatan 2014 (MATH EIGEN) khususnya matematika-C, Teman-teman Pejuang Kripto, teman KBMB angkatan 2014, Terimakasih atas segala pengalaman berharga, kerja sama dan kebersamaan atas kenang-kenangan indah yang dirajut bersama dalam menggapai impian.

Semoga Allah SWT melimpahkan rahmat dan karunia-Nya kepada kita semua. Akhirnya penulis berharap semoga dengan rahmat dan izin-Nya mudah-mudahan skripsi ini bermanfaat bagi penulis dan bagi pembaca. *Amiin*

Wassalamualaikum Warahmatullahi Wabarakatuh

Malang, 02 Juni 2020

Penulis

DAFTAR ISI

DAFTAR GAMBAR

ABSTRAK

Zaini, Ahmad. 2021. **Implementasi Algoritma Super Enkripsi Pada Pengaman Pesan Berbentuk Teks.** Skripsi. Program Studi Matematika Fakultas Sains dan Teknologi. Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Muhammad Khudzaifah, M.Si. (II) Juhari, M.Si

Kata kunci : Kriptografi, Enkripsi, Dekripsi, Super Enkripsi, *Playfair Cipher* , *Rail Fence Cipher*

Ilmu dan seni merahasiakan pesan menggunakan teknik penyandian dengan menyamarkan pesan sehingga tidak mudah dipahami adalah makna dari Kriptografi. Metode yang digunakan dalam mengamankan pesan salah satunya adalah Super Enkripsi. Super Enkripsi adalah penggabungan beberapa dari algoritma kriptografi klasik menjadi sebuah algoritma yang kuat. *Playfair* adalah salah satu algoritma Kriptografi klasik yang secara teknik penggunaannya dengan mensubstitusikan karakter atau huruf menggunakan matriks persegi yang berukuran 5×5. Pesan yang dikirimkan akan diproses dahulu dengan menggunakan algoritma *Playfair*, pada proses ini plainteks akan diubah menjadi digraphs atau berpasangan dari sejumlah karakter pesan plainteks. Proses ini memerlukan kunci sehingga plainteks tidak dengan mudah ditebak. Kunci berada pada baris pertama pada kolom pertama matriks persegi, menghilangkan karakter atau huruf ganda. Dalam memproses enkripsi dan dekripsi algoritma ini adalah memindahkan posisi karakter huruf, memindahkannya tidak seperti Cipher lain melainkan karakter huruf berpasangan sehingga analisis frekuensinya semakin banyak dan sulit dipecahkan karena makin banyak percobaan yang harus dilakukan.

Rail fence merupakan salah satu dari algoritma kriptografi klasik transposisi. Cara meng enkripsi dan dekripsinya membutuhkan kunci dan *offset*. Jika kunci = 4 maka offset <= 3, cara mengoperasikan algoritma *Rail Fence* adalah diagonal dari atas ke bawah, kunci dan *offset* menjadi hal penting untuk pertama memulai menuliskan karakter huruf ke dalam matriks. Keunggulan dari algoritma ini adalah pesan atau plainteks yang akan dikirimkan akan diproses menggunakan kunci dan (*offset*). Kunci dan *offset*-nya saling berhubungan, jika hanya memiliki salah satunya maka proses enkripsi dan dekripsi pun akan gagal dan tidak akan menemukan isi pesan yang asli.

Penelitian ini bertujuan untuk mengetahui proses enkripsi dan dekripsi pesan teks menggunakan algoritma Super Enkripsi. Hasil dari penelitian ini dalam menggabungkan dua algoritma klasik algoritma *Playfair* dan *Rail Fence* akan menghasilkan tingkat keamanan dalam mengenkripsi dan mendeskripsikan pesan yang dikirimkan, sehingga pesan yang dikirimkan tidak mudah untuk dibajak oleh orang-orang yang tidak bertanggung jawab. Pembajak pun memerlukan waktu yang cukup lama dalam memecahkan sandi karena harus mengulang percobaan dan penggunaan *Rail Fence* menjadi lebih sulit karena offset dan kunci saling berhubungan atau bahkan tidak bisa mengakses pesan tersebut.

ABSTRACT

Zaini, Ahmad. 2021. **On The Implementation of Super Encryption Algorithm on Text Message Security.** Thesis. Mathematics Study Program, Faculty of Science and Technology. Maulana Malik Ibrahim State Islamic University, Malang. Advisors: (1) Muhammad Khudzaifah , M.Si (2) Juhari , M.Si

Keywords: Cryptography, Encryption, Decryption, Super Encryption, Playfair Cipher , Rail Fence Cipher

The science and art of concealing messages using encryption techniques by disguising the message so that it is not easily understood is the meaning of Cryptography. One of the methods used in securing messages is Super Encryption. Super Encryption is the amalgamation of several of the classic cryptographic algorithms into a powerful algorithm. Playfair is one of the classical Cryptography algorithms which is technically used by substituting characters or letters using a 5x5 square matrix. The message sent will be processed first using the algorithm Playfair, in this process the plaintext will be converted into digraphs or in pairs of a number of plaintext message characters. This process requires a key so that the plaintext is not easily guessed. The key is in the first row of the first column of the square matrix, eliminating double characters or letters. In processing encryption and decryption this algorithm is to move the position of the character letters, moving them not like other Ciphers but character pairs of letters so that the frequency analysis is getting more and more difficult to solve because more experiments must be done.

The Rail fence is one of the classic transposition cryptographic algorithms. How to encrypt and decrypt it requires a key and an offset. If key = 4 then offset <= 3, how to operate the algorithm Rail Fence is diagonal from top to bottom, key and offset are important things to first start writing letters into the matrix. The advantage of this algorithm is that the message or plaintext that will be sent will be processed using the key and offset. The key and the offsets are related, if you only have one of them then the encryption and decryption process will fail and will not find the original message content.

This study aims to determine the process of encryption and decryption of text messages using the Super Encryption algorithm. The results of this study in combining the two classical algorithms, the algorithm Playfair and Rail Fence, will produce a level of security in encrypting and describing the messages sent, so that the messages sent are not easy to be hijacked by irresponsible people. The hijacker also takes a long time to crack the password because he has to repeat the experiment and using Rail Fence becomes more difficult because the offset and key are interconnected or can't even access the message.

ملخص

زيني ,أحمد. ٢٠٢١. تنفيذ خوارزمية التشفير الفائق على أمن الرسائل النصية.
البحث العلمي.قسم الرياضيات، جامعة مولانا مالك إبراهيم الإسلامية الحكومية
بمالانج.المشرف الأول (١) محمد حذيفة الماجستير، المشرف الثاني (٢) جوهري،
الماجستير.

كلمات : تشفير، تشفير، فك التشفير، سوبر التشفير، *Rail Fence Cipher*،*Playfair Cipher*

علم وفن إخفاء الرسائل باستخدام تقنيات التشفير عن طريق إخفاء الرسالة بحيث لا يسهل فهمها هو معنى التشفير. إحدى الطرق المستخدمة في تأمين الرسائل هي Super Super enkripsi.enkripsi هو اندماج العديد من خوارزميات التشفير الكلاسيكية في خوارزمية قوية. Playfair هي إحدى خوارزميات التشفير التقليدية التي تُستخدم تقنيًا عن طريق استبدال الأحرف أو الأحرف باستخدام مصفوفة مربعة قياسها خمسة أضعاف خمسة. ستتم معالجة الرسالة المرسلة أولاً باستخدام خوارزمية Playfair ، في هذه العملية سيتم تحويل النص العادي إلى رسومات رقمية أو في أزواج من عدد من أحرف رسالة النص العادي. تتطلب هذه العملية مفتاحًا حتى لا يتم تخمين النص العادي بسهولة. المفتاح موجود في الصف الأول من العمود الأول من المصفوفة المربعة ، مما يلغي الأحرف أو الحروف المزدوجة. في معالجة التشفير وفك التشفير ، تعمل هذه الخوارزمية على تحريك موضع أحرف الأحرف ، ونقلها ليس مثل الأصفار الأخرى ولكن أزواج الأحرف من الأحرف بحيث يصبح حل تحليل التردد أكثر وأكثر صعوبة لأنه يجب إجراء المزيد من التجارب

Rail Fence هي واحدة من خوارزميات نقل التشفير الكلاسيكية. كيفية تشفيرها وفك تشفيرها يتطلب مفتاح وإزاحة. إذا كان المفتاح = ٤ ثم الإزاحة <= ٣ ، فإن كيفية تشغيل Rail Fence سياج السكك الحديدية تكون قطرية من أعلى إلى أسفل ، يعد المفتاح والإزاحة من الأشياء المهمة لبدء كتابة الأحرف أولاً في المصفوفة. ميزة هذه الخوارزمية هي أن الرسالة أو النص العادي الذي سيتم إرساله ستتم معالجته باستخدام المفتاح و (الإزاحة). المفتاح والإزاحة

مرتبطان ، إذا كان لديك واحد منهما فقط ، فستفشل عملية التشفير وفك التشفير ولن تعثر على محتوى الرسالة الأصلية.

تهدف هذه الدراسة إلى تحديد عملية تشفير وفك تشفير الرسائل النصية باستخدام خوارزمية Super enkripsi.نتائج هذه الدراسة في الجمع بين الخوارزميات الكلاسيكية Playfair و Rail Fence سيؤدي إلى مستوى من الأمان في تشفير ووصف الرسائل المرسلة ، بحيث لا يسهل اختراق الرسائل المرسلة من قبل أشخاص غير مسؤولين. كسر كلمة المرور بسبب الاضطرار إلى تكرار التجربة واستخدام Rail Fence يصبح أكثر صعوبة لأن الإزاحة والمفتاح مترابطان أو لا يمكن حتى الوصول إلى الرسالة.

BAB I
PENDAHULUAN

1.1 Latar Belakang

Playfair merupakan *digraphs cipher* artinya setiap proses enkripsi maupun dekripsi dilakukan menggunakan pasangan karakter huruf (Setyaningsih, Jurnal Teknologi, 2009). Dalam penelitiannya Rina Candra Noer Santi (2010), metode Sandi *Playfair* yang di implementasikan pada pesan teks menggunakan matriks dengan ukuran 5×5 seperti yang digunakan pada matrik kunci *playfair cipher*. Batasannya hanya menggunakan pesan teks huruf alfabet tanpa memasukkan bilangan, dan menambahkan bilangan 1 sampai 5 untuk mempermudah memproses karakter dengan matriks 5×5 agar mudah dipahami. Penelitiannya Muhammad Sufyan Tsauri (2019), metode sandi yang dikembangkan dalam mengenkripsi dokumen dalam bentuk digital untuk mengamankan pesan teks ujian. Dengan menggunakan algoritma *Rail Fence Cipher* dalam mengenkripsi database teks ujian.

Zaman milenial sekarang terdapat banyak penemuan teknologi yang bisa mempermudah kegiatan manusia, teknologi yang sering digunakan manusia saat ini adalah teknologi komunikasi. Bentuk teknologi komunikasi pun beragam seperti pesan teks, suara, gambar, video, dan *sharing* data digital. Pengiriman pesan hakikatnya ada yang berbentuk pesan *public* (umum) dan ada yang berbentuk pesan (*privacy*) rahasia. Pesan yang berbentuk (*privacy)* rahasia biasanya memerlukan keamanan dalam proses pengiriman agar pesan tersebut bisa sampai kepada penerima pesan dengan catatan isi dari pesan tidak bisa diketahui oleh pihak manapun. Ilmu dan seni merahasiakan pesan menggunakan teknik penyandian

dengan menyamarkan pesan sehingga tidak mudah dipahami adalah makna dari Kriptografi. (Meyer, C., 1982). Pesan yang dikirimkan memungkinkan tidak ada seorangpun mengetahui pesan asli kecuali hanya yang bersangkutan. Al-Qur'an Surat An-Nisa': 58 dijelaskan:

Artinya :

Sesungguhnya Allah menyuruh kamu menyampaikan amanah kepada pemiliknya. Apabila kamu menetapkan hukum di antara manusia, hendaklah kamu tetapkan secara adil. Sesungguhnya Allah memberi pengajaran yang paling baik kepadamu. Sesungguhnya Allah Maha Mendengar lagi Maha Melihat. (Qs. An-Nisaa'/4:58).

Dalam ayat ini menjelaskan "*Sesungguhnya Allah menyuruh kamu menyampaikan amanah kepada pemiliknya*" tentang perintah untuk menyampaikan amanah kepada orang yang berhak, dan seruan untuk tidak menyebarkan informasi atau pesan yang kepada orang-orang yang tidak dikehendaki-Nya. Kriptografi adalah salah satu metode untuk mengamankan pesan supaya tetap terjaga kerahasiaannya dengan cara enkripsi dan dekripsi pada pesan. Enkripsi adalah proses penyandian pesan asli (*plaintext*) menjadi pesan tersandi (*ciphertext*). Dalam proses enkripsi dan dekripsi membutuhkan parameter untuk transformasi yang dinamakan kunci (Munir R. , 2004). Metode yang pada saat ini sudah banyak digunakan yaitu metode *Playfair Cipher*.

Berdasarkan penelitian sebelumnya, teknik enkripsi yang digunakan dalam penelitian ini menggunakan dua teknik algortima klasik yaitu algoritma *Playfair* yang termasuk dalam teknik substitusi *Cipher* dan algoritma *Rail Fence* yang termasuk dalam teknik tranposisi *Cipher* dalam mengamankan pesan teks.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang tersebut maka rumusan masalah dalam penelitian ini " Bagaimana proses enkripsi dan dekripsi pesan teks menggunakan metode super enkripsi ? "

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah maka tujuan penelitian adalah mengetahui proses enkripsi dan dekripsi pesan teks menggunakan metode super enkripsi.

1.4 Manfaat Penelitian

Hasil penulisan ini di harapkan mampu memberikan manfaat pada pembaca umumnya dan penulis khususnya, selain itu diharapkan:

1. Sebagai pembelajaran dan penelitian pengamanan data berupa pesan teks.
2. Sebagai bahan referensi dalam pengembangan penelitian lebih lanjut.
3. Sebagai sarana pengambangan keilmuan dibidang matematika terapan.
4. Sebagai aplikasi dalam melindungi privasi pesan teks.

1.5 Metode Penelitian

Metode Penelitian yang dilakukan dalam penelitian ini dengan cara studi literatur. Penulisan dimulai dengan mempelajari jurnal-jurnal, tugas akhir, artikel dan buku-buku tentang enkripsi dan dekripsi beserta algoritma-algoritmanya. Adapun langkah-langkah untuk menyelesaikan penelitian ini, sebagai berikut.

1. Menyusun enkripsi algoritma super enkripsi *(Playfair Cipher – Rail Fence Cipher)* pada pesan teks dengan langkah;

a. Menentukan 10 karakter atau lebih yang digunakan sebagai *plaintext*
b. Mengururtkan plainteks ke bentuk bigram (dua karakter/berpasangan)
c. Memasukkan karakter kunci yang digunakan kedalam matriks bujur sangkar 5x5 dengan tidak mengulang karakter yang sama kedalam matriks dari baris pertama horisontal dan dilanjutkan pada baris kolom kedua.
d. Mengoperasikan karakter plainteks dengan matriks 5x5 sehingga menghasilkan sebuah *ciphertext* hasil operasi bigram dari algoritma *Playfair Cipher*.
e. Menyusun hasil enkripsi *playfair cipher* dan dioperasikan dengan Algoritma *Rail Fence* dengan menuliskan dan membagi *plaintext* menjadi baris diagonal naik turun, disesuaikan dengan jumlah kunci dan *offset*.
f. Menyusun hasil enkripsi dari Algoritma *Rail Fence* secara berbaris horisontal sehingga menghasilkan sebuah *Ciphertext*

2. Menyusun dekripsi menggunakan algoritma super enkripsi pada pesan teks *Playfair Cipher – Rail Fence Cipher* dengan langkah sebagai berikut
   a. Proses Dekripsi ini menggunakan hasil dalam proses enkripsi yang disebut dengan Cipherteks. Cipherteks akan diproses menggunakan algortima *Playfair Cipher* dan Algoritma *Rail Fence Cipher* dengan menggunakan kunci (*Key*) dan *Offset* yang sama, menghasilkan karekater huruf berupa *Plaintext*.
   b. *Plaintext* dari hasil *Rail Fence Cipher* di operasikan dengan algoritma *Playfair Cipher* dengan matrik 5x5, menggunakan kunci(*key*) *Playfair*

Cipher pada langkah enkripsi pesan, dilanjutkan dengan mengurutkan karakter.

c. Menghasilkan karakter plainteks dari operasi algoritma *Playfair Cipher*.

3. Implementasi sebagai ketepatan hasil perhitungan dengan komputer menggunakan aplikasi.

1.6 Sistematika Penulisan

Dalam penulisan dalam penelitian ini, penulis menggunakan sistematika yang terdiri dari empat bab, dan masing-masing bab dibagi dalam sub-bab dengan sistematika penulisan sebagai berikut.

Bab I Pendahuluan

Membahas tentang latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, tinjauan pustaka, metode penulisan dan sistematika penulisan yang menggambarkan secara singkat isi laporan penelitian ini.

Bab II Kajian Pustaka

Membahas tentang teori-teori penunjang yang digunakan dalam bab selanjutnya, meliputi Pesan teks, keamanan Pesan teks, Algoritma kriptografi, Algorima kriptografi klasik, Algoritma Kriptografi modern, Enkripsi, Dekripsi, Super Enkripsi, Algoritma *Playfair - Railfence*

Bab III Pembahasan

Bab ini berisi tentang langkah-langkah pemebentukan *Ciphertext* yang melalui tahap super enkripsi substitusi dan transposisi yang dilakukan melalui metode Algoritma *Playfair Cipher – Railfence Cipher* sehingga didapatkan suatu *Ciphertext* yang telah terenkripsi dan juga berisi implementasi algoritma kedalam aplikasi *Python*.

Bab IV Penutup

Bab ini berisi kesimpulan dari pembahasan yang sesuai dengan hasil penelitian, yang selanjutnya dapat digunakan sebagai sarana bagi pembaca dan peneliti selanjutnya.

BAB II
KAJIAN PUSTAKA

2.1 Kombinatoarial

Kombinatorial (*Combinatotic*) adalah ilmu matematika yang mempelajari pengaturan objek-objek. Solusi yang ingin kita peroleh dengan kombinatorial ini adalah jumlah cara pengaturan objek-objek tertentu di dalam himpunannya. Di dalam kriptografi, kombinatorial berkaitan dengan hal-hal seperti: berapa banyak kunci yang bisa dibentuk, berapa banyak kombinasi plainteks dan cipherteks yang mungkin, berapa banyak bit yang mempunyai paritas ganjil (Munir R. , 2019).

2.1.1 Kaidah Dasar Menghitung

Kombinatorial didasarkan pada hasil yang diperoleh dari suatu percobaan (*experiment*). Percobaan adalah proses fisik yang hasilnya dapat di amati. Dua kaidah dasar yang digunakan sebagai teknik menghitung dalam kombinatorial adalah kaidah perkalian (*rule of product*) dan kaidah penjumlahan (*rule of sum*). Kedua kaidah ini dapat digunakan untuk memecahkan banyak persoalan menghitung (Munir R. , 2019).

a. Kaidah perkalian (*rule of product*)

   Jika n buah percobaan masing-masing mempunyai $P_1, P_2, \dots P_n$ hasil yang mungkin, maka bila semua percobaan dilakukan secara serempak, maka terdapat $P_1 \times P_2 \times \dots \times P_n$ hasil percobaan yang mungkin.

b. Kaidah penjumlah (*rule of sum*).

   Jika n buah percobaan masing-masing mempunyai $P_1, P_2, \dots P_n$ hasil yang mungkin, maka bila hanya salah satu percobaan yang dilakukan, maka terdapat $P_1 \times P_2 \times \dots \times P_n$ hasil percobaan yang mungkin.

Contoh

Sebuah kata-sandi (*password*) sistem komputer panjangnya enam sampai delapan karakter. Tiap karakter boleh berupa huruf besar dan kecil tidak dibedakan. Kita akan menghitung berapa jumlah kata sandi yang dapat dibentuk. Banyaknya huruf alfabet adalah 26 (A-Z) dan banyak angka desimal adalah 10 (0-9), jadi seluruhnya 36 karakter. Masing-masing huruf atau angka menjadi pilihan untuk posisi karakter didalam *password.*

Jawaban:

untuk kata-sandi dengan panjang karakter 6 karakter, jumlah kemungkinan kata sandi adalah (36) (36) (36) (36) (36) (36) = 36^6 = 2.176.782.336. Untuk kata-sandi dengan panjang 7 karakter, jumlah kemungkinan kata-sandi adalah (36) (36) (36) (36) (36) (36) (36) = 36^7 = 78.364.164.096. dan untuk kata sandi dengan panjang 8 karakter, jumlah kemungkinan kata-sandi adalah(36) (36) (36) (36) (36) (36) (36) (36)= 36^8 = 2.821.109.907.456. Dengan menggunakan kaidah penjumlahan, jumlah seluruh kata-sandi adalah 2.176.782.336 + 78.364.164.096 + 2.821.109.907.456 = 2.901.650.833.888 buah.

2.1.2 Permutasi

Permutasi adalah jumlah urutan berbeda dari pengaturan objek-objek. bentuk khusus aplikasi aturan perkalian. Misalkan jumlah objek adalah n, maka urutan pertama dipilih dari n objek, urutan kedua dipilih ari $n-1$ objek, urutan ketiga dari $n-2$ objek. Begitu seterusnya, dan urutan terakhir dipilih dari 1 objek tersisa. Menurut kaidah perkalian permutasi dari n objek adalah (Munir R. , 2016).

$$n(n-1)(n-2)\dots(2)(1) = n! \qquad (2.1)$$

Permutasi r dari n objek, disimbolkan dengan $P(n,r)$, adalah jumlah kemungkinan urutan r buah objek yang dipilih dari n buah objek, dengan $r \leq n$, yang dalam hal ini, pada setiap kemungkinan urutan tidak ada objek yang sama, jumlah urutan yang berbeda adalah (Munir R. , 2019)

$$P(n,r) = \frac{n!}{(n-r)!} \tag{2.2}$$

Contoh

Ada 26 huruf dalam alfabet. Jika huruf-huruf tersebut disusun, maka terdapat 26! Urutan susunan yang dapat dihasilkan. Jika kita menyusun 5 buah huruf dari alfabet, maka jumlah kemungkinan susunan huruf yang dapat dibentuk adalah $P\ (26{,}5) = \frac{26!}{(26-5)!} = 7893600$ buah

Kita akan menghitung beberapa jumlah *string* yang dapat dibentuk yang terdiri dari 4 huruf berbeda dan di ikuti dengan 3 angka yang berbeda pula. Ada $P(26{,}4)$ cara mengisi posisi 4 huruf dan $P(10{,}3)$ cara untuk mengisi posisi 3 buah angka. Karena *string* disusun oleh 4 huruf dan 3 angka, maka jumlah *string* yang dapat dibuat adalah $P(26{,}4) \times P(10{,}3) = 258.336.000$

2.1.3 Kombinasi

Bentuk khusus dari permutasi adalah kombinasi. Jika pada permutasi urutan kemunculan diperhitungkan, maka pada kombinasi, urutan kemunculan diperhitungkan, maka pada kombinasi, urutan kemunculan di abaikan. Sebagai contoh, urutan $acb, bca\ dan\ acb$ di anggap sama dan dihitung sekali.

Kombinasi r elemen dari n elemen, disimbolkan dengan $C(n,r)$ atau $\binom{n}{r}$ adalah jumlah pemilihan yang tidak terurut r elemen yang di ambil dari n buah elemen, yang banyaknya adalah (Munir R. , 2019)

$$C(n,r) = \frac{n!}{r!\,(n-r)!} \tag{2.3}$$

Contoh

Setiap karakter ASCII panjangnya 1 *byte* (1 *byte* = 8 bit). Jumlah *byte* yang mengandung 3 buah bit 1 adalah $C(8,3) = 8!/(3!\,5!)$ =56 buah.

Contoh

Banyaknya cara membagikan 5 buah kartu remi yang di ambil dari tumpukan 52 buah kartu ke masing-masing dari 4 orang adalah $C(52,5) \times C(47,5) \times C(42,5) \times C(37,5)$.

2.2 Pesan

Pesan (*message*) adalah data atau informasi yang dapat dibaca dan dimengerti maknanya. Nama lain untuk pesan adalah plainteks (*plaintext*) atau teks-jelas (*cleartext*). Pesan dapat berupa data informasi yang dikirim (melalui kurir, saluran telekomunikasi, dsb) atau yang disimpan didalam media perekam (kertas, *storage*, dsb). Pesan yang tersimpan tidak hanya berupa teks, tetapi juga dapat berbentuk citra (*image*) suara/bunyi (*audio*) dan video, atau berkas biner lainnya (Munir R. , 2006).

Komunikasi dalam kehidupan manusia terasa sangat penting, karena dengan komunikasi dapat menjembatani segala bentuk ide yang akan disampaikan seseorang. Dalam setiap melakukan komunikasi unsur penting di antaranya adalah pesan, karena pesan disampaikan melalui media yang tepat, bahasa yang

dimengerti, kata-kata yang sederhana dan sesuai dengan maksud, serta tujuan pesan itu akan disampaikan dan mudah dicerna oleh komunikan. Adapun pesan itu menurut Onong Effendy, menyatakan bahwa pesan adalah : “suatu komponen dalam proses komunikasi berupa paduan dari pikiran dan perasaan seseorang dengan menggunakan lambang, bahasa/lambang-lambang lainnya disampaikan kepada orang lain. (Effendy, 1989)

2.3 Kriptografi

Pengamanan terhadap data (informasi) dapat dilakukan dengan beberapa cara, yaitu steganografi, *watermarking*, dan kriptografi. Steganografi adalah ilmu menyembunyikan pesan tersembunyi dalam suatu cara sehingga selain si pengirim dan si penerima, tidak ada seorang pun yang mengetahui atau menyadari keberadaan suatu pesan rahasia. Steganografi berbeda dengan *watermarking* dan kriptografi, yang menyamarkan arti pesan, namun tidak menyembunyikan keberadaan pesan. Steganografi dan kriptografi sering kali digunakan secara bersamaan untuk menjamin keamanan pesan rahasia.

Menurut B. Schneier dalam bukunya Emy Setyaningsih (2015) Kriptografi (*cryptography*) berasal dari bahasa Yunani, yaitu dari kata *crypto* dan *graphia* yang berarti ‘penulisan rahasia’. Kriptografi adalah ilmu ataupun seni yang mempelajari bagaimana membuat suatu pesan yang dikirimkan oleh pengirim dapat disampaikan kepada penerima dengan aman. Kriptografi merupakan bagian dari suatu cabang ilmu matematika yang disebut dengan kriptologi (cryptology).

2.4.1 Mekanisme Kriptografi

Sistem kriptografi (*cryptosystem*) bekerja dengan cara menyandikan suatu pesan menjadi kode rahasia yang dimengerti oleh pelaku sistem informasi. Beberapa istilah atau terminologi dalam kriptografi yaitu :

a. Pesan, Plainteks dan Cipherteks

Pesan (*message*) adalah data atau informasi yang dapat dibaca, dipersepsi dan dimengerti artinya pesan dapat berupa teks, citra (*image*), suara/bunyi (*audio*), video atau bentuk-bentuk biner lainnya, barik berupa bentuk digital maupun analog. Pesan berupa teks sering disebut juga plainteks (*plaintext*) atau teks-jelas (*cleartext*). Agara pesan tidak dapat dipahami isinya oleh pihak lain, maka pesan perlu disandikan menjadi pesan yang tidak dapat dimengerti lagi maknanya. Pesan teks yang tersandi disebut Cipherteks (*Ciphertext*).

b. Pengirim dan penerima

Komunikasi data melibatkan pertukaran pesan antara dua entitas. Pengirim (*Sender*) adalah entitas yang mengirim pesan kepada entitas lainnya. Penerima (*Receiver*) adalah entitas yang menerima pesan. Pengirim dan penerima tidak harus berupa orang, tetapi juga dapat berupa mesin, robot atau komputer. Jadi, orang bisa berkomunikasi dengan orang lainnya. Untuk pesan yang disimpan didalam memori, istilah pengirim dan penerima pesan sudah tidak relevan lagi. Dengan kata lain, untuk pesan yang tersimpan pengirim dan penerima adalah pemilik pesan itu sendiri.

c. Enkripsi dan dekripsi

Proses penyandian plainteks menjadi cipherteks disebut enkripsi (*encryption*). Sedangkan proses mengembalikan cipherteks menjadi plainteks semula dinamakan dekripsi (*decryption*).

d. Cipher, kode dan kunci

Algoritma Kriptografi untuk enkripsi dan dekripsi disebut juga *cipher*. *Cipher* dapat di artikan sebagai aturan untuk *enchipering* atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Beberapa *Cipher* memerlukan algoritma yang berbeda untuk *enciphering* dan *deciphering*. *Cipher* seharusnya tidak perlu rahasia, namun pada zaman dulu (dan mungkin saja masih ada pada zaman sekarang) *Cipher* dirahasiakan oleh pembuatnya.

Algoritma kriptografi yang keamanannya terletak pada kerahasiaan algoritmanya dinamakan algoritma *restricted.* Algoritma ini pada zaman dulu dilakukan sekolompok orang untuk bertukar pesan sesama anggota kelompok. Mereka membuat sendiri algortima enkripsi dan dekripsi sendiri dan yang mengetahui algoritma tersebut hanya anggota kelompok dan tidak boleh dibocorkan diluar kelompok tersebut.

Disaat salah satu anggota keluar dari kelompok maka algoritmanya kriptografi harus segera diganti dengan algoritma yang baru. Algoritma *restricted* ini terkesan tidak praktis. Pada zaman modern dalam mengatasi masalah di atas dipergunakan sebuah kunci (*key*). Algoritma kriptografi tidak perlu harus rahasia tetapi kunci harus rahasia.

Hal ini bersesuaian dengan prinsip Kerckhoff yang mengatakan "*Semua algoritma kriptografi harus publik. Hanya kunci yang rahasia"* (Schneier, 1996). Kunci adalah parameter yang digunakan di dalam *enciphering* dan *deciphering.* Kunci umumnya berupa string pendek atau berupa deretan bilangan.

Istilah *cipher* sering disalahpahami dengan Kode (*code*). *Cipher* tidak sama dengan kode. Jika *Cipher* adalah transformasi karakter-ke-karakter atau bit-ke-bit tanpa memperlihatkan struktur bahasa pesan, maka kode sering di acu sebagai prosedur yang mengganti setiap plainteks dengan suatu kata kode, misalnya

Kereta api datang dikodekan menjadi hutan bakau hancur

Kode juga dapat berupa deretan angka dan huruf yang tidak bermakna, seperti

Kereta api datang dikodekan menjadi 23450 6543 78923

Tranformasi dari plainteks menjadi kode sering disebut *encoding*, sedangkan transformasi kebalikannya sering disebut *decoding*. Untuk melakukan *endcoding* dan *decoding* digunakan dokumen yang disebut buku kode (*codebook*) (Munir R. , 2006).

2.4.2 Tujuan Kriptografi

Aspek – aspek keamanan dalam kriptografi yaitu sebagai berikut:

a. **Kerahasiaan** (*confidentiality*), adalah layanan yang ditunjukkan untuk menjaga agar pesan tidak dibaca oleh pihak-pihak yang tidak berhak. Didalam kriptografi layanan ini direalisasikan dengan menyandikan pesan menjadi cipherteks. Misalnya pesan "Harap datang pukul 8" disandikan

menjadi “TrxC#45motyptre!%”. Istilah lain yang senada dengan *confidentiality* adalah *secrecy* dan *privacy*. Lebih jauh mengenai metode penyandian akan dibahas didalam bab-bab selanjutnya.

b. **Integritas Data** (*data integrity*), adalah layanan yang menjamin bahwa pesan masih asli/utuh atau belum pernah di manipulasi selama pengiriman. Dengan kata lain, aspek keamanan ini dapat di ungkapkan sebagai pertanyaan: “apakah pesan yang diterima masih asli atau tidak mengalami perubahan (modifikasi)?”. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi pesan oleh pihak-pihak yang tidak berhak, antara lain kedalam pesan yang sebenarnya. Dalam kriptografi layanan ini direalisasikan dengan menggunakan tanda-tanda digital (*digital signature*). Pesan yang telah ditandatangani menyiratkan bahwa pesan yang dikirim adalah asli.

c. **Otentikasi** (*authentication*) adalah layanan yang berhubungan dengan identifikasi, baik mengidentifikasi kebenaran pihak-pihak yang berkomunikasi (*user authentication* atau *entity authentication*) maupun mengidentifikasi kebenaran sumber pesan (*data origin authentication*). Dua pihak saling berkomunikasi harus dapat mengontetikasi satu sama lain sehingga ia dapat memastikan sumber pesan. Pesan yang dikirim melalui saluran komunikasi juga harus di otentikasi asalnya. Dengan kata lain aspek keamanan ini dapat di ungkapkan sebagai pertanyaan : “apakah pesan yang diterima benar-benar berasal dari pengirim yang benar?” otentikasi sumber pesan secara implisit juga berarti sumber pesan sudah tidak benar. Oleh

karena itu layanan integritas data selalu dikombinasikan dengan layanan otentikasi sumber pesan.

d. **Nirpenyangkalan** (*non-repudiation*) adalah layanan untuk mencegah entitas yang berkomunikasi melakukan penyangkalan yaitu pengiriman pesan menyangkal melakukan pengiriman atau penerimaan pesan menyangkal telah menerima pesan. Sebagai contoh misalkan pengirim pesan memberi otoritas kepada penerima pesan untuk melakukan pembelian, namun kemudian ia menyangkal telah memberikan otoritas tersebut. Contoh lainnya, misalkan seorang pemilik emas mengajukan tawaran kepada toko emas bahwa ia akan menjual emasnya. Tetapi, tiba-tiba harga emas turun drastis, lalu ia membantah telah mengajukan tawaran menjual emas. Dalam hal ini pihak toko emas perlu prosedur nirpenyangkalan untuk membuktikan bahwa pemilik emas telah melakukan kebohongan (Munir R. , 2006).

2.4 Algoritma kriptografi

Berdasarkan waktu kemunculannya, kriptografi dibedakan menjadi yaitu kriptografi klasik dan kriptografi modern. Kriptografi klasik terbagi menjadi 2 yaitu substitusi dan transposisi, sedangkan kriptografi modern terbagi dari sifat kuncinya menjadi dua bagian yaitu kriptografi simetris dan kriptografi asimetris (Hasugian, 2017)

2.5.1 Kriptografi klasik

Kriptografi ini ditemukan saat sebelum ditemukannya komputer atau masih menggunakan pensil dan kertas sebagai penulisan berbasis karakter yaitu enkripsi

dan dekripsi dilakukan pada setiap karakter pesan. Pada dasarnya algoritma kriptografi klasik dapat dikelompokkan ke dalam dua macam cipher

a. *Cipher* Substitusi

   Penggantian pesan atau teks berupa plainteks dengan huruf, angka atau simbol (Devi & Harshini, 2019). Pola bit ini biasanya disebut dengan Cipherteks. *Cipher* yang termasuk kedalam *Cipher* substitusi ini adalah *Caesar Cipher, Playfair Cipher, Hill Cipher.*

b. *Cipher* transposisi

   Cipher Transposisi adalah mengubah urutan huruf *plaintext* atau melakukan *transpose* terhadap rangkaian karakter (Setyaningsih, 2015) . *Cipher* transposisi mempunyai berbagai macam algoritma yang berbeda-beda seperti *Rail Fence Cipher, Myszkowski Transposition, Route Cipher, Columnar Transposition*.

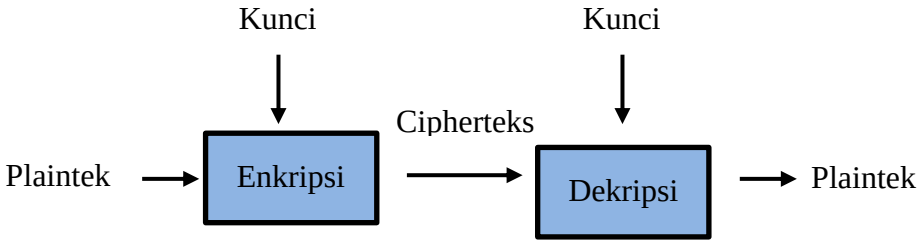
2.5.2 Kriptografi Modern

Berdasarkan kuncinya Kriptografi modern terbagi dua yaitu algoritma simetris dan algoritma asimetris

a. Algoritma Simetris

   Algoritma simetris atau disebut juga dengan algoritma kriptografi konvensional adalah algoritma yang menggunakan kunci untuk proses enkripsi sama dengan kunci untuk proses dekripsi. Algoritma simetris dibagi menjadi dua kategori yaitu algoritma aliran (*Stream Cipher*) dan algoritma blok (*Block Cipher*). Pada algortima Aliran, proses penyandian berorientasi pada sekumpulan bit atau byte data. Sedangkan

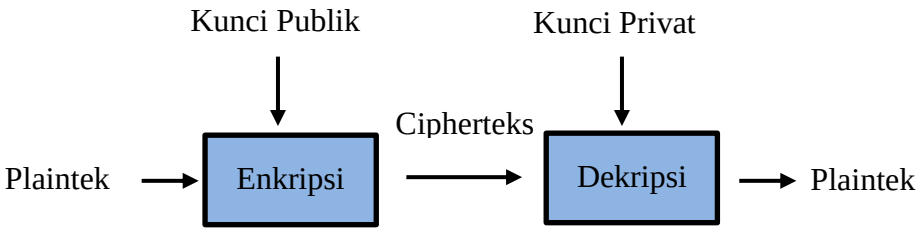
pada algoritma blok, proses peenyandian berorientasi pada sekumpulan bit atau byte data (per blok) adapun contoh algoritma kriptografi ini yaitu DES, RC4, AES (Basri, 2016).



Gambar 2. 1 Skema Kriptografi Simetris

b. Algoritma Asimetris

Pada pertengahan tahun 70-an Whitfield Diffie dan Martin Hellman menemukan teknik enkripsi asimetris yang merevolusi dunia kriptografi. Algoritma asimetris adalah pasangan kunci-kunci kriptografi yang salah satunya dipergunakan untuk proses enkripsi dan yang satu lagi untuk dekripsi. Semua orang yang mendapat kunci *public* dapat menggunakan untuk mengenkripsi suatu pesan, sedangkan hanya satu orang saja yang memiliki rahasia tertentu dalam hal ini kunci *privacy* untuk melakukan pembongkaran terhadap sandi yang dikirimkan untuknya. Contoh dari penggunaam kunci asimetris ini adalah RSA, DSA, Elgamal (Basri, 2016).



Gambar 2. 2 Skema Kriptografi Asimetris

2.5 *Playfair Cipher*

Playfair cipher termasuk ke dalam *polygram cipher* yang melakukan substitusi secara bigram (kelompok yang terdiri dari dua huruf). Cipher ini ditemukan oleh Sir Charles Wheatstone pada tahun 1854. Namun dipromosikan oleh Baron Lyon Playfair sehingga nama yang di abadikan adalah nama yang terakhir ini. *Playfair Cipher* digunakan oleh tentara inggris pada Perang *Boer* (Perang Dunia II). *Cipher* ini mengenkripsi pasangan karakter/huruf (bigram atau digraf) bukan huruf tunggal seperti pada *Cipher* klasik lainnya. Tujuannya adalah untuk membuat analisis frekuensi menjadi sangat sulit sebab frekuensi kemunculan huruf-huruf di dalam cipherteks menjadi datar(*flat*) (Munir R. , 2006).

Kunci kriptografinya adalah 25 huruf yang disusun didalam matriks persegi 5 x 5 dengan menghilangkan huruf j dari alfabet (dalam beberpa versi, yang dihilangkan adalah huruf Q, sedangkan dalam versi lain huruf J dan I ditulis dalam satu tempat sebagai I/J) setiap elemen persegi berisi huruf yang berbeda satu sama lain (Munir R. , 2006).

Contoh sebuah persegi Playfair:

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

Gambar 2. 3 Huruf dalam matriks persegi *Playfair*

Huruf-huruf didalam matriks persegi biasanya hasil permutasi huruf-huruf. Jumlah kemungkinan matriks persegi 5x5 yang dapat dibuat adalah sebanyak dari huruf 25 huruf alfabet, yaitu

25! = 25!=15.511.210.043.330.985.984.000.000

Susunan karakter/huruf didalam matriks persegi juga bisa menggunakan dari kalimat yang mudah di ingat, misalkan

JALAN GANESHA SEPULUH

Plaintext yang dihasilkan yaitu

JA LA NG AN ES HA SE PU LU HZ = *Plaintext*

STAND = *Key*

Kunci (*key*) jika terdapat huruf yang sama atau ganda maka harus dibuang atau di ambil hanya satu dalam urutan karakter kunci (*key*). Kemudian tambahkan karakter/huruf alfabet lain (kecuali J). Masukkan karakter/huruf ke dalam matriks persegi 5x5 dari kiri atas ke kanak dan terus menerus pada kolom dibawahnya membentuk matriks persegi 5x5 *playfair* dan yang ditulis dalam persegi playfair *5x5* adalah kuncinya. Dimana plainteks adalah huruf bigram yang akan di enkripsi dalam bujur sangkar *playfair* :

S	T	A	N	D
B	C	E	F	G
H	I	K	L	M
O	P	Q	R	U

V	W	X	Y	Z

Gambar 2. 4 Huruf atau karakter dalam matriks Persegi 5x5

Untuk melakukan enkripsi, matriks pesegi 5x5 *playfair* diperluas dengan menambahkan kolom ke-enam dan baris keenam, misalnya kita gunakan matriks pesegi 5x5 *playfair* yang pertama untuk diperluar menjadi:

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

Gambar 2. 5 Matriks Persegi *Playfair* 5x5

Baris ke-6 = baris ke-1, kolom ke-6 = kolom ke-1

Pesan yang akan di enkripsi di atur terlebih dahulu sebagai berikut:

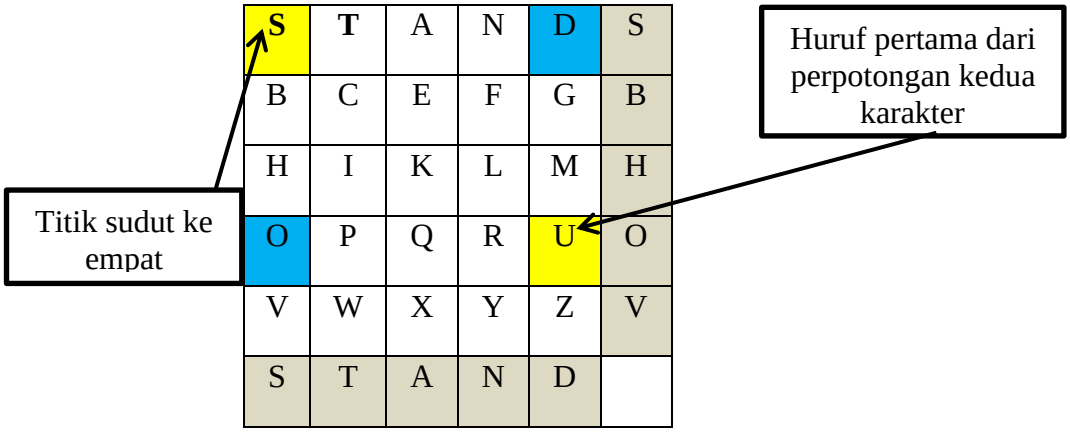
1. Apabila terdapat karakter/huruf (J) gantilah dengan karakter/huruf (I)
2. Tulis karakter/huruf dalam bentuk berpasangan (bigram)
3. Jangan sampai ada pasangan huruf yang sama, jika terdapat huruf yang sama atau berdampingan, tambahkan karakter/huruf X ditengah-tengah karakter/huruf (atau karakter/huruf lain, misalnya Z.
4. Jika terdapat jumlah huruf ganjil, maka tambahkan huruf X di akhir

Misalnya pada *plaintex* cc tidak ada huruf J, maka pesan langsung ditulis dalam pasangan huruf bigram:

GO OD BR OX OM SX SW EX EP CL EA NZ

teknik enkripsi dengan algortima *Playfair Cipher* sebagai berikut:

1. Apabila terdapat dua karakter/huruf terdapat pada baris persegi yang sama maka tiap karakter/huruf diganti dengan karakter/huruf di kanannya.
2. Apabila terdapat dua karakter/huruf pada kolom matriks persegi 5 x 5 yang sama maka tiap karakter/huruf diganti dengan karakter/huruf dibawahnya.
3. Apabila terdapat dua karakter/huruf tidak pada baris yang sama atau kolom yang sama, maka karakter/huruf pertama diganti dengan karakter/huruf pada perpotongan baris karakter/huruf pertama dengan kolom karakter/huruf kedua. karakter/huruf kedua diganti dengan karakter/huruf pada titik sudut keempat dari persegi panjang yang dibentuk dari 3 karakter/huruf yang digunakan sampai sejauh ini. Misalnya *OD* =



Gambar 2. 6 Posisi Ciperteks huruf atau Karakter

Contoh

Pesan yang akan dikirimkan berupa GOOD BROOMS SWEEP CLEAN telah ditulis dalam bigram sebagai berikut:

GO OD BR OX OM SX SW EX EP CL EA NZ : *Plaintext*

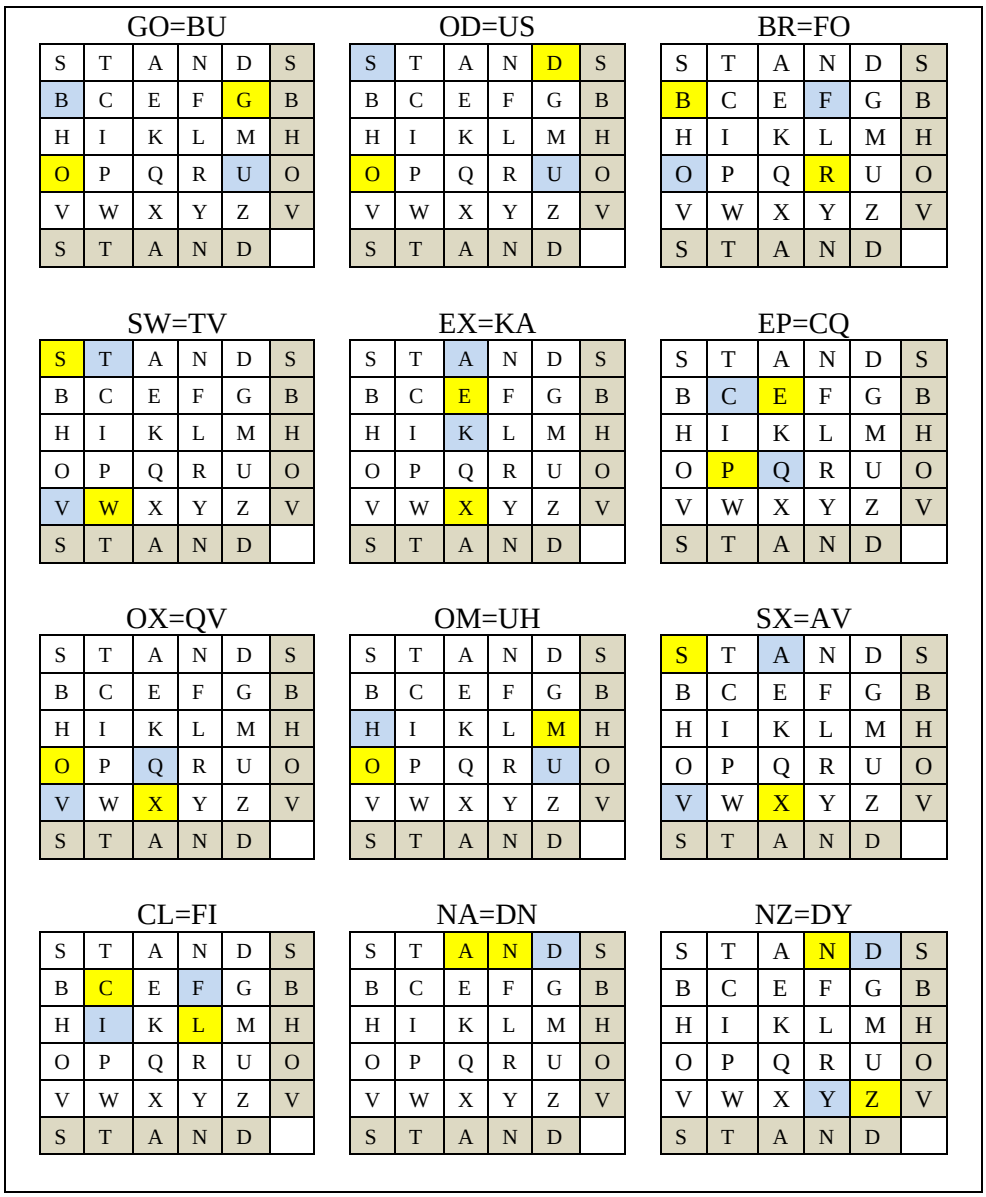
STAND : *Key*

dan persegi *playfair* yang digunakan setelah diperluaskan adalah

S	T	A	N	D	S

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

Gambar 2. 7 Huruf Bigram dalam Matriks Persegi yang diperluas



Gambar 2. 8 Proses enkripsi pasangan (bigram) huruf atau karakter

Cipertext yang dihasilkan adalah

BU US FO TV KA CQ QV UH AV FI DN DY

Teknik dekripsi adalah pengembalian dari teknik enkripsi, Langkah-langkahnya sebagai berikut:

1. Apabila terdapat dua huruf/karakter (bigram) terdapat pada baris matriks persegi 5X5 yang sama maka tiap huruf diganti dengan huruf dikirinya.
2. Apabila terdapat dua huruf/karakter pada kolom matriks persegi 5X5 yang sama maka tiap huruf diganti dengan huruf di atasnya.
3. Apabila terdapat dua huruf/karakter tidak pada baris yang sama atau kolom yang sama, maka huruf/karakter pertama diganti dengan huruf/karakter pada perpotongan baris huruf/karakter pertama dengan kolom huruf/karakter kedua. huruf/karakter kedua diganti dengan huruf/karakter pada titik sudut keempat dari persegi yang dibentuk dari tiga huruf/karakter yang digunakan sampai sejauh ini.
4. Buanglah huruf X yang tidak mengandung makna (Munir R. , 2019)

Karakter atau huruf berpasangan (bigram) *plaintext* tersebut, di ururtkan dengan menaruh karakter/huruf pada matriks persegi pasangan karakter/huruf *ciphertext*, misalnya:

BU US FO TV KA CQ QV UH AV FI DN DY **:** *Ciphertext*

STAND **:** *Key*

Ambil satu bigram yang sudah terenkripsi misalnya *BU*

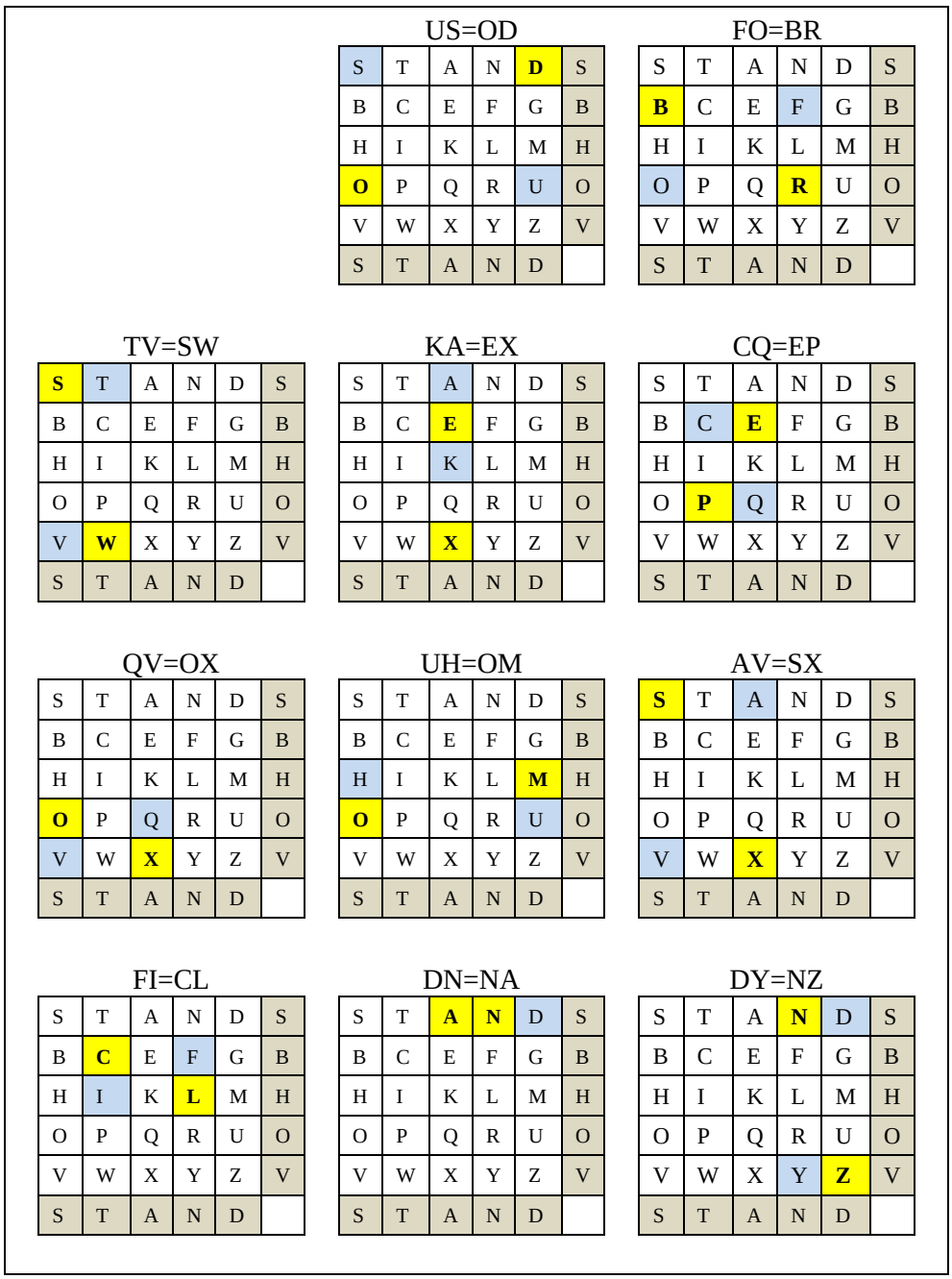
S	T	A	N	D	S
B	C	E	F	G	B
H	I	K	L	M	H
O	P	Q	R	U	O

V	W	X	Y	Z	V
S	T	A	N	D	

Gambar 2. 9 Proses dekripsi huruf Bigram persegi *Playfair Cipher*

Menghasilkan *plaintext* pasangan huruf (bigram) *GO*

Begitu juga dengan bigram yang lain



Gambar 2. 10 Proses dekripsi algoritma *playfair Cipher*

Menghasilkan *Plaintext* :

GO OD BR OX OM SX SW EX EP CL EA NZ

Jika semua digabungkan menjadi sebuah kata **GOOD BROOMS SWEEP CLEAN**

Didapatkan pesan awal yang dikirim oleh *sender* pengirim pesan.

Sayangnya ukuran poligram didalam *playfair cipher* tidak cukup besar, hanya dua huruf sehingga *playfair cipher* tidak aman, meskipun playfair cipher sulit dipecahkan dengan analisis frekuensi relatif huruf-huruf, namun ia dapat dipecahkan dengan analisis frekuensi pasangan huruf. Dalam bahasa Inggris kita bisa mempunyai frekuensi kemunculan pasangan huruf, misalnya pasangan huruf *TH* dan *HE* paling sering muncul. Dengan menggunakan tabel frekuensi kemunculan pasangan huruf didalam bahasa inggris dan cipherteks yang cukup banyak, *playfair cipher* dapat dipecahkan (Munir R. , 2006).

2.6 *Rail Fence Cipher*

Rail Fence Cipher adalah salah satu teknik enkripsi untuk menyamarkan tulisan dengan mengubah posisi karakter dengan bentuk diagonal ke bawah dan ke atas. Cipher ini menggunakan perubahan posisi atau susunan dan tidak memiliki kunci tertentu. Untuk memecahkannya kita harus memperhatikan tingkatan dari tulisan tersebut, karena cipher ini biasanya sistematis (Yusuf, 2018).

Algoritma ini berasal dari sebuah *cipher* transposisi. Oleh karena itu *cipher* dapat disebut sebagai *cipher* transposisi karena sebenarnya metode *cipher* transposisi ini mempermutasikan karakter-karakter plainteks, yaitu dengan menyusun ulang urutan karakter dalam pesan teks. Contoh paling sederhana dalam penggunaan *cipher* transposisi adalah dengan membalikkan karakter-karakter dalam suatu kata. Misalkan kata PASURUAN di enkripsi menjadi NAURUSAP,

ini adalah contoh paling sederhana. Sedangkan contoh *cipher* transposisi yang lebih rumit sebagai berikut:

Misalkan kita mempunyai plainteks

AKU KAN LULUS TAHUN INI

Untuk mengenkripsi pesan, plainteks ditulis secara horizontal dengan lebar kolom tetap, misal sebesar 5 karakter (kunci k=5)

A K U A K

A N L U L

U S T A H

U N I N I

Maka chiperteksnya dibaca secara vertikal menjadi

AAUUKNSNULTIAUANKLHI

Pada zaman Yunani dahulu, tentara sparta menggunakan sebuah alat yang dinamakan *scrytale.* Alat ini terdiri dari sebuah silinder dan pita panjang dari daun *papyrus*. Pesan dituliskan horizontal dan bila pita dilepaskan, maka huruf-huruf didalamnya telah tersusun membentuk sebuah pesan rahasia. *Scrytale* merupakan sebuah penerapan *cipher*r transposisi pada zaman dahulu.

Cipher transposisi mempunyai berbagai macam algoritma yang berbeda-beda seperti *Rail Fence Cipher, Route Cipher, Columnar Transposition,* dan *Myzkowski Transposition*. Setiap algoritma itu mempunyai sebuah kelebihan masing-masing. Oleh karena itu penulis akan mencoba membahas Algoritma *Rail Fence Cipher*.

Algoritma ini melibatkan penulisan plainteks sehingga mempunyai baris atas dan baris bawah yang terpisah. Urutan karakter pada baris atas akan di ikuti

oleh karakter berikutnya pada baris bawahnya, dan seterusnya sehingga *n-rail*. Apabila penulisan ke bawah sudah mencapai *n*, maka penulisan dilakukan ke baris atasnya dan seterusnya. Bila penulisan ke atas juga sudah mencapai *n-rail*, maka penulisan dilakukan seperti awal. *Ciphertext* dibaca secara horizontal. Untuk lebih jelasnya, berikut adalah contohnya:

Misalkan kita mempunyai *plainteks*

AKUAKANLULUSTAHUNINI

enkripsi dilakukan dengan kunci k = 4, offset = 0

A . . . . . N . . . . . T . . . . . N .
. K . . . A . L . . . S . A . . . I . I
. . U . K . . . U . U . . . H . N . . .
. . . A . . . . . L . . . . . U . . . .

maka *ciphertext*-nya menjadi

ANTNKALSAIIUKUUHNALU

Namun enkripsi juga dapat dilakukan dengan memulainya bukan dari baris paling atas (*offset* = 0), namun bisa juga dari baris lainnya dengan *offset* = 3. Dengan menggunakan contoh plainteks di atas:

AKUAKANLULUSTAHUNINI

Enkripsi dilakukan dengan kunci k = 4, *offset* = 3

. . . A . . . . . L . . . . . U . . . .
. . U . K . . . U . U . . . H . N . . .
. K . . . A . L . . . S . A . . . I . I
A . . . . . N . . . . . T . . . . . N .

maka cipherteksnya menjadi **ALUUKUUHNKALSAIIANTN**

Biasanya penulisan cipherteks dilakukan menjadi blok-blok standar biasanya sepanjang 5 karakter. Bila hasil cipherteks tidak habis dibagi dengan panjang karakter, maka penambahan karakter *dummy* dilakukan pada saat enkripsi.

Rail Fence Cipher mempunyai kelebihan dibandingkan algoritma lainnya dalam proses penulisan plainteks menjadi cipherteks karena penulisan dapat dilakukan dibaris mana saja. Hal ini akan menambah kerumitan dalam proses enkripsi maupun dekripsi.

Secara keseluruhan algoritma *cipher* Transposisi ini mempunyai kelemahan karena serumit apapun algoritma yang kita pakai untuk mengubah posisi atau permutasi suatu teks, kita hanya akan mengubah urutan teks (plainteks) tersebut tidak mengubahnya menjadi karakter lain. Kemunculan karakter plainteks akan sama dengan cipherteksnya, hal ini dapat memberikan pertunjuk bahwa proses enkripsi menggunakan salah satu algoritma *cipher* transposisi. Sehingga, usaha untuk memecahkan suatu *cipher* transposisi tidaklah sulit bila kita mencoba semua algoritma *cipher* transposisi.

2.7 Super Enkripsi

Pada dasarnya algoritma kriptografi klasik dibagi menjadi dua yaitu, substitusi dan Transposisi yang telah dibahas sebelumnya. Kedua teknik ini termasuk algoritma yang mudah dipecahkan melalui *Brute force. Brute force* adalah memecahkan sandi dengan mencoba seluruh sandi yang mungkin kedalam *ciphertext* sampai dapat dirubah kedalam *plaintext.* Rata-rata, setengah dari seluruh kemungkinan kunci harus dicoba agar dapat berhasil (Stallings, 2003).

Untuk mempersulit algoritma sehingga kunci tidak mudah ditemukan oleh pihak yang tidak bersangkutan, maka dikembangkan algoritma baru dengan

menggabungkan kedua teknik algoritma klasik tersebut. Super enkripsi merupakan sebuah konsep dengan menggunakan dua atau lebih dari teknik substitusi dan transposisi *cipher* untuk mendapatkan suatu algoritma yang lebih andal atau susah dipecahkan (Ariyus, 2008). Untuk menjalankan teknik super enkripsi ini, harus memahami teknik substitusi dan transposisi. Super enkripsi dijalankan dengan melakukan enkripsi pesan dengan teknik substitusi, selanjutnya *ciphertext* yang telah didapatkan di enkripsi lagi dengan teknik transposisi

2.8 Enkripsi dan Dekripsi

Enkripsi merupakan hal yang sangat penting dalam kriptografi, merupakan pengamanan data yang dikirim supaya terjaga kerahasiaannya. Enkripsi adalah proses pengacakan pesan asli (*Plaintext*) menjadi acak (*Ciphertext*) yang sulit dibaca oleh orang yang tidak mempunyai kunci Dekripsi. Dekripsi adalah kebalikan dari enkripsi yaitu mengembalikan *ciphertext* menjadi *plaintext*. Algoritma yang digunakan untuk enkripsi (Ariyus, 2008).

Proses enkripsi dikatakan aman apabila menghasilkan *ciphertext* yang membutuhkan waktu lama (misalnya seribu tahun) untuk didekripsikan oleh orang yang tidak mempunyai kunci dekripsi (Kromodimoeljo, 2010).

2.9 Kriptanalisis

Sejarah kriptografi pararel dengan sejarah kriptanalisis (*Crytanalysis*), yaitu bidang ilmu dan seni untuk memecahkan cipherteks, kata “ kriptanalisis “ sendiri relatif masih baru (pertama kali di ungkapkan oleh Wiliam Friedman pada tahun 1920), namun sebenarnya teknik kriptanalisis sudah ada sejak abad ke-9. Adalah seorang ilmuan arab pada abad IX bernama Abu Yusuf Yaqub Ibnu Ishaq Ibnu

As-Sabbah Ibnu ‘Omran Ibnu Ismail AL-Kindi, atau yang lebih dikenal sebagai Al-Kindi yang menulis buku tentang seni memecahkan kode. Dalam bukunya yang berjudul ‘*Risalah Fi Istikhraj al-Mu’amma* ‘ (*Manuscript for the Deciphering Cryptographic Messages*). Ia menuliskan naskah untuk menguraikan kode-kode rahasia.

Yang dilakukan Al-Kindi adalah dalam kriptanalisis dikenal dengan nama teknik analsis frekuensi, yaitu teknik untuk memecahkan cipherteks berdasarkan frekuensi kemunculan karakter didalam pesan dan kaitannya dengan frekuensi kemunculan karakter didalam alfabet. Istilah lain dari kriptanalisis adalah pembajakan. Analsis frekuensi dilatar belakangi oleh fakta bahwa dalam *cipher* gagal menyembunyikan statistikk kemunculan karakter didalam cipherteksnya. Misalnya, didalam bahasa inggris huruf “E” adalah huruf paling sering muncul didalam kalimat-kalimat berbahasa inggris. Jika didalam cipherteks terdapat huruf yang paling sering muncul, maka kemungkinan besar huruf tersebut didalam plainteksnya adalah huruf “E”. Teknik analisis frekuensi masih digunakan dalam kriptanalisis modern, tetapi karena cipher semakin rumit, maka pendekatan matematika masih tetap dominan dalam melakukan kriptanalisis. Sejarah kriptanalisis mencatat hasil gemilang seperti pemecahan telegram Zimmermann yang membawa amerika serikat ke kancah Perang Dunia I dan pemecahan cipherteks dari mesin enigma ikut andil mengakhiri Perang Dunia II (Munir R. , 2019).

Kriptanalisis bertujuan untuk memecahkan cipherteks menjadi plainteks semula tanpa memiliki akses ke kunci yang digunakan. Kriptanalisis berusaha menemukan kunci dan mengungkap plainteks. Dalam membahas serangan

kriptografi, kita selalu mengasumsikan kriptanalis mengetahui algoritma kriptografi yang digunakan, sehingga satu-satunya keamanan sistem kriptografi terletak sepenuhnya pada kunci.

Hal ini didasarkan pada prinsip *Kerckhoff* (1883) " Semua algoritma Kriptografi Harus Publik ; Hanya Kunci Rahasia".

Dengan kata lain, kriptanalisis mengetahui algoritma enkripsi dan dekripsi secara detail. Merahasiakan algoritma kriptografi bukan solusi yang praktis, sebab setiap kali algortima berhasil diketahui lawan, maka kriptografer harus membuat algoritma baru. Dengan membuat algoritma menjadi publik, maka cukup kunci yang dirahasiakan. Jika kunci berhasil dicuri, maka kunci baru dibangkitkan tanpa harus mengganti algoritmanya. Jadi tidak mengherankan kalau semua algoritma kriptografi telah dipublikasikan didalam berbagai jurnal dan buku-buku sehingga siapapun dapat mempelajarinya.

BAB III
PEMBAHASAN

3.1 Proses Enkripsi Pesan Teks Menggunakan Metode Super Enkripsi

Pada pembahasan ini teknik algoritma yang dipakai algoritma *Playfair Cipher* menggunakan matriks persegi 5x5. Bentuk umum penyandian algoritma *Playfair Cipher* adalah menggunakan karakter pembentuk *plaintext* dan *ciphertext* yang menggunakan seluruh karakter huruf (A – Z). Berikut adalah matriks persegi 5x5 yang umum atau biasanya digunakan dalam algoritma *Playfair Cipher*

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

Gambar 3. 1 Matriks Persegi 5x5

Sebelum dilakukan enkripsi *Plaintext* langkah-langkah melakukannya yaitu :

1. Ganti huruf (Q) pada *Plaintext* dengan huruf (I), yang dipakai adalah huruf Q. Bebapa versi lain (menggunakan huruf J atau X).
2. Tulis karakter/huruf berbentuk (bigram) berpasangan

3. Apabila ada dua karakter/huruf yang sama atau berdampingan, maka sisipkan karakter/huruf X ditengahnya (atau karakter/huruf lain, misalnya Z.
4. Apabila terdapat jumlah karakter atau huruf ganjil , maka tambahkan huruf X pada bigram terakhir.

Berikut ini algoritma dalam melakukan enkripsi dengan menggunakan algoritma *Playfair* dengan matriks persegi 5X5

1. Apabila terdapat dua karakter atau huruf pada baris matriks persegi 5 X 5 yang sama maka setiap karakter/huruf di ambil huruf dikanannya
2. Apabila terdapat dua karakter atau huruf pada kolom matriks persegi 5 X 5 yang sama maka setiap karakter atau huruf di ambil huruf dibawahnya.
3. Apabila terdapat dua karakter atau huruf tidak pada kolom dan baris yang sama maka karakter atau huruf pertama diambil karakter atau huruf pada perpotongan baris karakter atau huruf pertama dengan kolom karakter atau huruf kedua. Karakter atau huruf kedua diganti dengan karakter atau huruf pada titik sudut keempat dari matriks persegi 5 X 5 yang dibentuk dari tiga karakter atau huruf yang digunakan.

Tahap pertama

Doni sebelum mengirim pesan yaitu dengan mengenkripsi pesan , Pesan yang akan dikirimkan adalah

Pesan = SETELAH PANDEMI AKAN WISUDA

menggunakan algoritma *Playfair Cipher* dengan Kunci

Kunci = INGAT ORANG TUA

- Proses enkripsi sebagai berikut :

  Disusun menjadi bentuk bigram (dua huruf)

  SETELAH PANDEMI AKAN WISUDA menjadi

Plaintext = **SE TE LA HP AN DE MI AK AN WI SU DA**

Kunci (*Key*) = **INGATORANGTUA**

Menjadi

Kunci (*Key*) = **INGATORU**

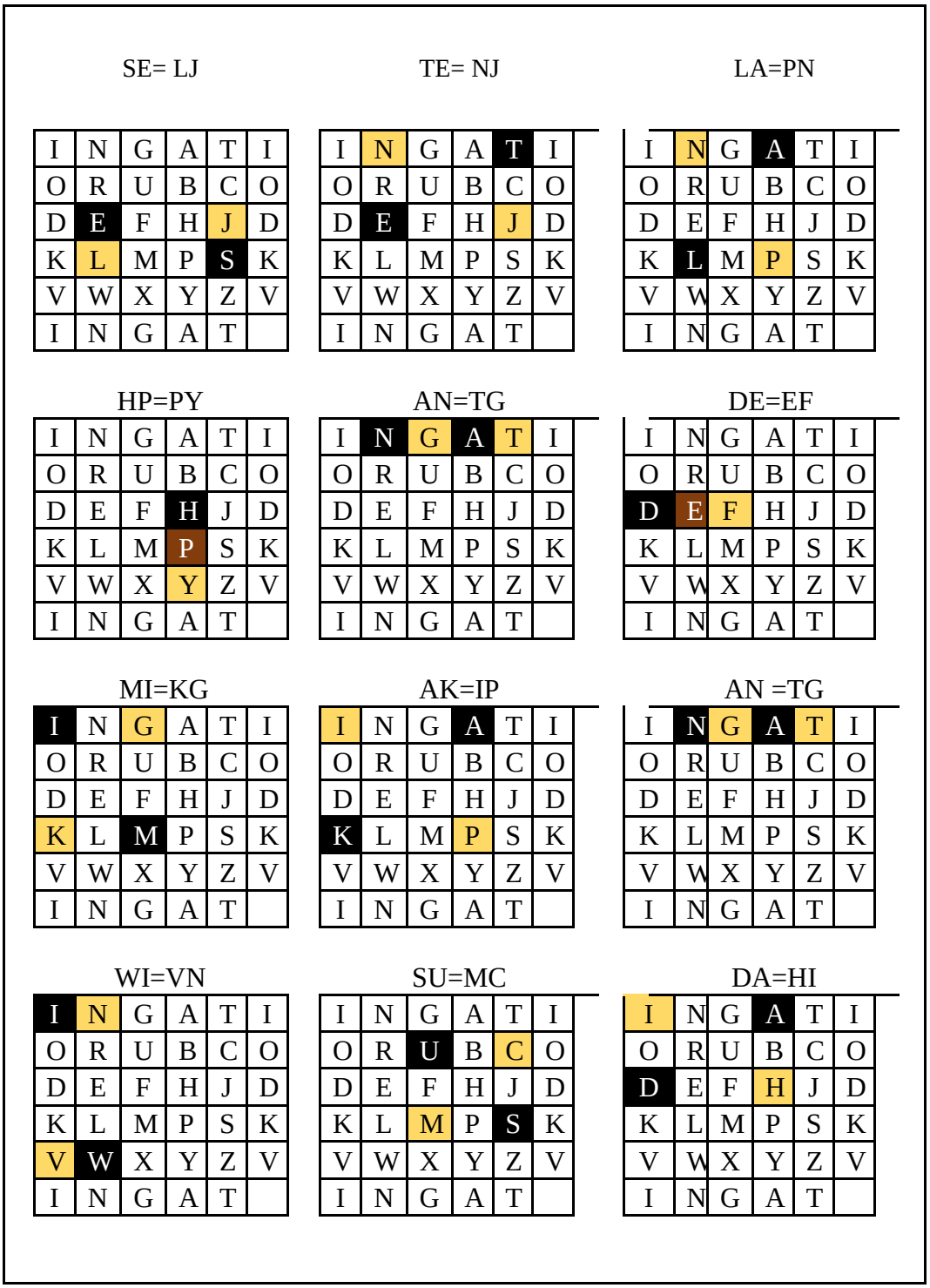
Kunci yang sudah dengan karakter yang sama tidak dimasukkan kedalam matriks 5x5

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

Gambar 3. 2 Matriks Persegi 5x5 dengan kunci INGAT ORU

Dari hasil memasukkan plainteks **SE TE LA HP AN DE MI AK AN WI SU DA**

didapat sebuah hasil cipherteks yaitu



Gambar 3. 3 Proses Enkripsi Setiap Bigram

Menghasilkan *ciphertext* yaitu " **LJ NJ PN PY TG EF KG IP TG VN MC HI"**

Tahap kedua

Setelah dari proses enkripsi menggunakan algoritma *Playfair Cipher* dilanjut enkripsi menggunakan algoritma *Rail Fence*

Pertama proses enkripsi *Rail Fence Cipher* yaitu memasukkan cipherteks dari proses enkripsi *Playfair Cipher.*

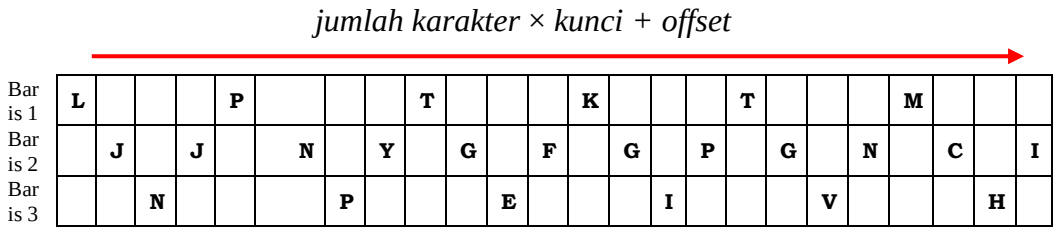
Cipherteks yaitu "**LJ NJ PN PY TG EF KG IP TG VN MC HI"**

Plaintext **= LJNJPNPYTGEFKGIPTGVNMCHI**

Key = 3

Offset = 0

Terlebih dahulu menyiapkan jumlah kunci yang yaitu 3, yang berarti jumlah matriks sama dengan jumlah baris 3 dan jumlah kolom sebanyak kali jumlah karaktek plainteks dengan baris. Ditambah jumlah *offset* (jika ada) kolom.



Gambar 3. 4 Proses Enkripsi Algoritma Rail Fence

Pada tahap ini cara menuliskan hasil enkripsi *Rail Fence cipher* adalah dengan menuliskan baris karakter atau huruf baris I, baris II, dan baris III berurutan

Baris I = "**LPTKTM"**

Baris II = "**JJNYGFGPGNCI"**

Baris III = "**NPEIVH**"

menghasilkan *ciphertext* = **LPTKTM JJNYGFGPGNCI NPEIVH**

3.2 Proses Dekripsi Pesan Teks Menggunakan Metode Super Enkripsi

Pada proses dekripsi adalah lawan dari pembentukan sandi atau proses dikembalikan seperti semula, memakai teknik *Rail Fence* setalah itu teknik *Playfair Cipher.*

Ciphertext = **LPTKTM JJNYGFGPGNCI NPEIVH**

Dengan jumlah karakter adalah 24 karakter

Dengan *key* (kunci) = 3 , dam *offset* = 0

Maka ,

Langkah Pertama

Ditentukan terlebih dahulu jumlah matriksnya dengan= *jumlah karakter* × *kunci* + *offset*= 24 × 3+0 =72

Baris 1	*				*				*				*				*				*			
Baris 2		*		*		*		*		*		*		*		*		*		*		*		*
Baris 3			*				*				*				*				*				*	

Gambar 3. 5 Proses Dekripsi Algoritma Rail Fence

Langkah Kedua

Hitung dan sesuaikan jumlah karakter pada setiap baris

Ciphertext = **LPTKTMJJNYGFGPGNCINPEIVH**

Baris I = 6 huruf = **LPTKTM**

Baris II =12 huruf = **JJNYGFGPGNCI**

Baris III =6 huruf = **NPEIVH**

Langkah Ketiga

Seseuai karakter dengan langkah pertama dengan menempatkan karakter pada posisi yang telah ditentukan dengan cara membacanya adalah diagonal naik turun sejumlah baris dan kolom (Langkah 1)

Baris 1	**L**				**P**				**T**				**K**				**T**				**M**			
Baris 2		**J**		**J**		**N**		**Y**		**G**		**F**		**G**		**P**		**G**		**N**		**C**		**I**
Baris 3			**N**				**P**				**E**				**I**				**V**				**H**	

Gambar 3. 6 Proses dekripsi dengan menyusun karakter

Maka akan didapatkan hasil *plaintext* = **"LJNJPNPYTGEFKGIPTGVNMCHI"**

Setelah didapatkan hasilnya *Plaintext* maka

Langkah kelima

Proses dekripsi selanjutnya adalah menggunakan mengoperasikan dengan algoritma *Playfair Cipher*, caranya adalah kebalikan dari proses enkripsi algoritma *Playfair Cipher*.

Dengan *cipertext* atau hasil dari proses dekripsi *Rail Fence* adalah

Cipertext = **LJNJPNPYTGEFKGIPTGVNMCHI**

Dengan menggunakan kunci (*key*) = INGAT ORANG TUA

Maka didapatkan hasil sebagai berikut

LJ=SE

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

NJ=TE

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

PN=LA

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

PY=HP

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

TG=AN

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

EF=DE

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

KG=MI

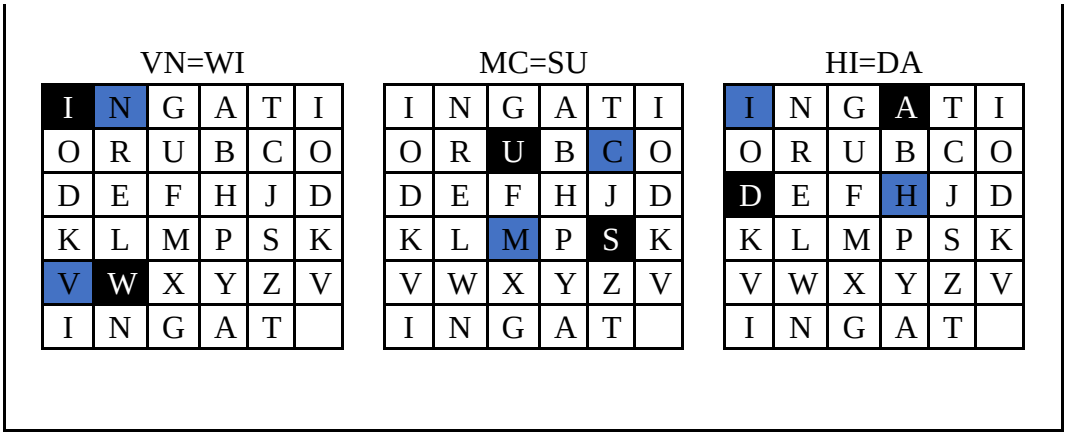
I	N	G	A	T	I
O	R	U	B	C	O
D	E	F	H	J	D
K	L	M	P	S	K
V	W	X	Y	Z	V
I	N	G	A	T	

IP=AK

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

TG=AN

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



Gambar 3. 7 Proses Dekripsi Menggunakan Algoritma *Playfair Cipher*

Didapatkan hasil *Plaintext* = **SE TE LA HP AN DE MI AK AN WI SU DA**

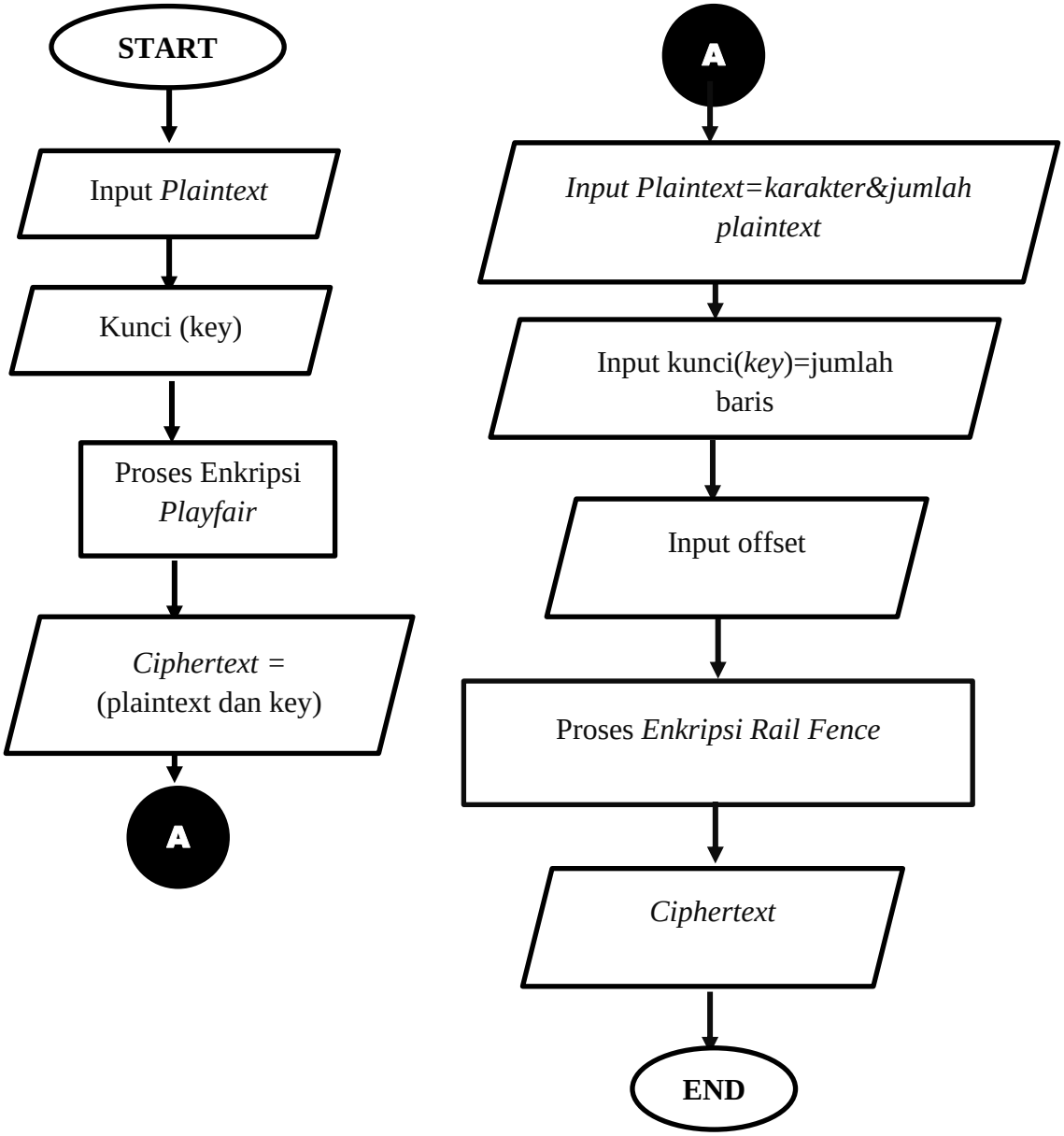
Dengan isi pesan asli kalau di urutkan menjadi " **SETELAH PANDEMI AKAN WISUDA** "

3.3 Implementasi Super Enkripsi Menggunakan Aplikasi Python

Setelah mengetahui proses manual dari pembuatan penyandian pesan berbentuk teks menggunakan algoritma *Playfair Cipher* dan *Rail Fence Cipher* maka langkah selanjutnya adalah pembuatan program aplikasi untuk mendukung ketepatan dalam penyandian pesan teks.

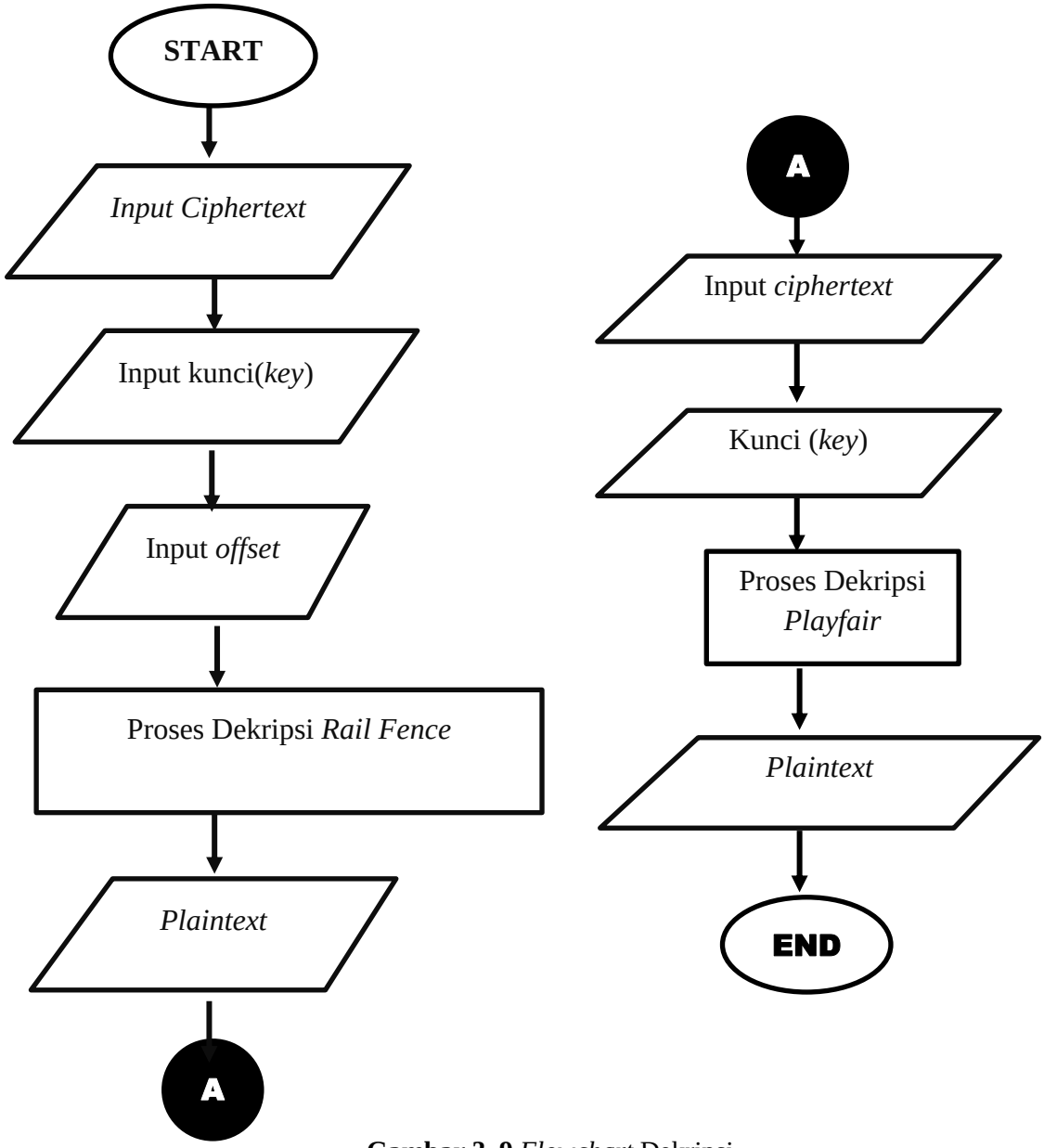
Proses super enkripsi dengan menggunakan algoritma *Playfair Cipher* dan *Rail Fence Cipher* dengan tahap pertama adalah di enkeripsi tahap kedua didekripsi agar hasil pesan teks menjadi seperti semula. Berikut adalah rancangan *Flowchart* penyandian super enkripsi *Playfair Cipher* dan *Rail fence Cipher* dengan langkah-langkah yang sudah dijelaskan sebelumnya.

Flowchart enkripsi *Playfair Cipher – Rail Fence Cipher*

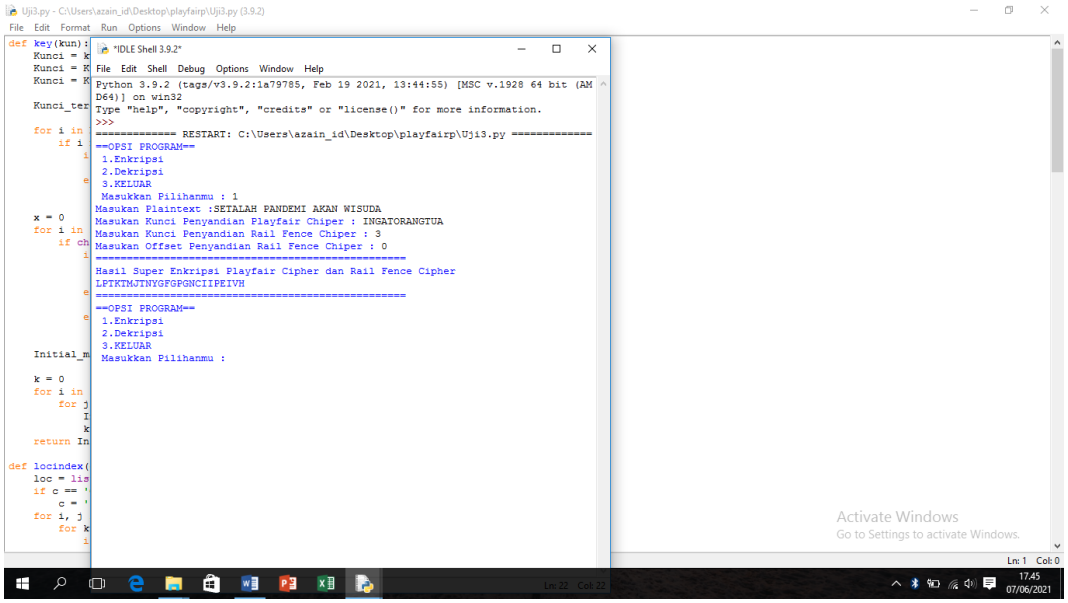


Gambar 3. 8 *Flowchart* Enkripsi

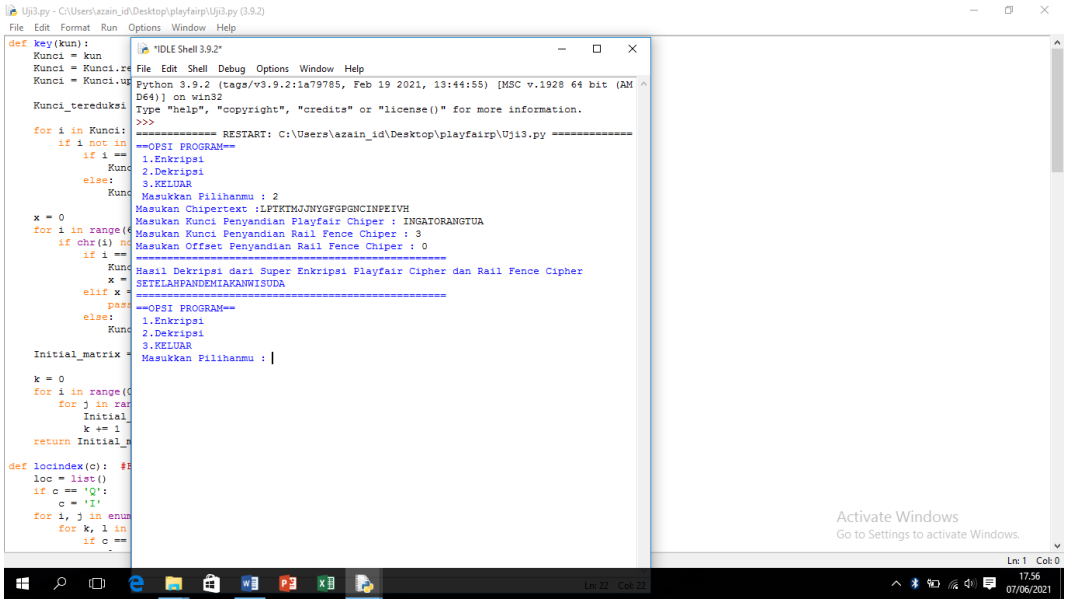
Flowchart dekripsi *Playfair Cipher– Rail Fence Cipher*



Gambar 3. 9 *Flowchart* Dekripsi



Gambar 3. 10 Hasil enkripsi menggunakan algoritma *Playfair Cipher* dan *Rail Fence Cipher*



Gambar 3. 11 Hasil dekripsi menggunakan algoritma *Playfair Cipher* dan *Rail Fence Cipher*

Hasil implementasi menggunakan aplikasi *Python* mendapat hasil plainteks yang sama secara aplikasi maupun dengan secara manual

3.4 Kajian Integrasi Keislaman Terhadap Pentingnya Amanah

Salah satu hal yang tidak lepas dari kehidupan sehari-hari komunikasi, komunikasi memungkinkan kita untuk saling bertukar pendapat, bertukar fikiran, informasi dan lain sebagainya. Kerahasiaan dalam berkomunikasi sangat dibutuhkan, agar orang lain yang tidak bertanggung jawab tidak bisa semenah-menah dalam mendengarkan, membaca dan menyebarkan informasi yang didapatkan dengan cara menyadap pesan, komunikasi, dan data dengan cara yang tidak diperkenankan secara negara maupun secara hukum agama. Dalam agama islam pun Allah SWT memerintahkan kepada setiap muslim berkewajiban untuk mengajak dan menyerukan perilaku kebaikan, dan menghindarkan dari segalah hal yang bersifat keburukan. Pesan, informasi dan data harus disampaikan kepada yang berhak menerima. seperti halnya dalam surah An-Nisa' ayat: 58

۞ اِنَّ اللّٰهَ يَأْمُرُكُمْ اَنْ تُؤَدُّوا الْاَمٰنٰتِ اِلٰٓى اَهْلِهَاۙ وَاِذَا حَكَمْتُمْ بَيْنَ النَّاسِ اَنْ تَحْكُمُوْا بِالْعَدْلِۗ اِنَّ اللّٰهَ نِعِمَّا يَعِظُكُمْ بِهٖۗ اِنَّ اللّٰهَ كَانَ سَمِيْعًاۢ بَصِيْرًا ٥٨

Artinya :

58. Sesungguhnya Allah menyuruh kamu menyampaikan amanah kepada pemiliknya. Apabila kamu menetapkan hukum di antara manusia, hendaklah kamu tetapkan secara adil. Sesungguhnya Allah memberi pengajaran yang paling baik kepadamu. Sesungguhnya Allah Maha Mendengar lagi Maha Melihat. (QS.An-Nisa 4/58).

Ayat ini memerintahkan agar menyampaikan "amanat" kepada yang berhak. Pengertian "amanat" dalam ayat ini, ialah sesuatu yang dipercayakan kepada seseorang untuk dilaksanakan dengan sebaik-baiknya. Kata "amanat" dengan pengertian ini sangat luas, meliputi "amanat" Allah kepada hamba-Nya, amanat

seseorang kepada sesamanya dan terhadap dirinya sendiri. Amanat Allah terhadap hamba-Nya yang harus dilaksanakan antara lain: melaksanakan apa yang diperintahkan-Nya dan menjauhi larangan-Nya. Semua nikmat Allah berupa apa saja hendaklah kita manfaatkan untuk taqarrub (mendekatkan diri) kepada-Nya.

Amanat seseorang terhadap sesamanya yang harus dilaksanakan antara lain: mengembalikan titipan kepada yang punya dengan tidak kurang suatu apa pun, tidak menipunya, memelihara rahasia dan lain sebagainya. Sifat adil seorang suami terhadap istrinya, begitu pun sebaliknya, seperti melaksanakan kewajiban masing-masing terhadap yang lain, tidak membeberkan rahasia pihak yang lain, terutama rahasia khusus antara keduanya yang tidak baik diketahui orang lain.

يٰٓاَيُّهَا الَّذِيْنَ اٰمَنُوْا لَا تَخُوْنُوا اللّٰهَ وَالرَّسُوْلَ وَتَخُوْنُوْٓا اَمٰنٰتِكُمْ وَاَنْتُمْ تَعْلَمُوْنَ

Artinya:

Wahai orang-orang yang beriman, janganlah kamu mengkhianati Allah dan Rasul serta janganlah kamu mengkhianati amanat yang dipercayakan kepadamu, sedangkan kamu mengetahui. (QS.Al-Anfal/8:27)

Al-Qur'an surah Al-Anfal ayat 27 menerangkan bahwa Allah menyerukan kepada kaum muslimin yang beriman untuk tidak mengkhianati Allah dan Rosul-Nya dengan cara tidak mengabaikan kewajiban-kewajiban yang harus mereka laksanakan, tidak melanggar larangan-larangan-Nya. Tidak mengkhanati amanat yang yang telah dipercayakan kepada mereka. Sabda Nabi Muhammad SAW:

"Rasulullah saw pada setiap khutbahnya selalu bersabda: "Tidak beriman orang yang tak dapat dipercaya, dan tidak beragama orang yang tak dapat dipercaya." (Riwayat Ahmad dan Ibnu Hibban dari Anas bin Malik).

dan hadist riwayat muslim dari Abu Hurairah menerangkan juga yang ***Artinya:***

"Tanda-tanda orang munafik itu ada tiga. Apabila menuturkan kata-kata ia berdusta, dan apabila berjanji ia menyalahi, dan apabila diberi kepercayaan ia berkhianat." (H.R Muslim - Abu Hurairah).

BAB IV
PENUTUP

4.1 Kesimpulan

Berdasarkan hasil Pembahasan dan menggunakan aplikasi *Python* , maka dapat disimpulkan bahwa:

Penggunaan sistem keamanan kriptografi super enkripsi memakai teknik algoritma *Playfair Cipher* dan *Rail Fence Cipher,* sistem keamanan menjadi lebih sukar untuk dipecahkan. Karena pada proses penyandian terjadi dua kali proses enkripsi menggunakan algoritma yang berbeda dan teknik pengembalian atau dekripsi pesan terjadi dua kali.

Keamanan super enkripsi terletak pada jumlah karakter plainteks yang digunakan saat di enkripsi menggunakan *Playfair Cipher* menghasilkan banyaknya jumlah percobaan agar menemukan isi pesan dan menggunakan *Rail fence cipher* dengan jumlah *offset* dan kunci yang padu agar teknik ini jadi tidak mudah lagi untuk ditebak atau diserang oleh pembajakan.

4.2 Saran

Pada penelitian ini terdapat beberapa saran yang bisa dipertimbangkan untuk pengembangan pada penelitian yang berikutnya, yaitu:

1. Penelitian berikutnya disarankan bisa dapat membangun sebuah aplikasi yang bisa diterapkan pada perangkat lunak android, ios, windows dan perangkat lunak lainnya pada masanya.

2. Pada pengembangan selanjutnya media yang digunakan adalah selain menggunakan teks , dapat juga menggunakan media atau format lain seperti .jpeg, .png, .rar, .mpeg/mp4, .mp3 dan lain sebagainya .
3. Penelitian selanjutnya diharapkan kengguanan lebih dari dua algortima dalam mengenkripsi dan dekripsi pesan sehingga pesan akan lebih sulit dipecahkan oleh serangan pembajakan.

DAFTAR RUJUKAN

Basri. (2016). Kriptografi Simetris Dan Asimetris Dalam Perspektif. *Jurnal Ilmiah Ilmu Komputer, 2*(2), 16-23.

Devi, K. R., & Harshini, G. N. (2019, April-June). Analysis and Comparison of Substitution and Transposition Cipher. *IJRAR- International Journal of Research and Analytical Reviews, 6*(2), 549-555.

Effendy, O. U. (1989). *Kamus Komunikasi.* Bandung: PT. Mandar Maju.

Hasugian, B. S. (2017). Peranan Kriptografi Sebagai Keamanan Sistem Informasi Pada Usaha Kecil Dan Menengah. *Jurnal Warta Universitas Dharmawangsa* , 53.

Meyer, C., M. (1982). Cryptography, A New Dimension in Computer Data Security. *Cryptography*.

Munir, R. (2004). Sistem Kriptografi Kunci-Publik Diktat Kuliah. Bandung: Depatemen Teknik informatika Institute Teknologi Bandung.

Munir, R. (2006). Kriptografi. Bandung: Informatika.

Munir, R. (2019). *Kriptografi Edisi kedua.* Bandung: PENERBIT INFORMATIKA.

Schneier, B. (1996). *Aplied Cryptography 2nd.* John Wiley & Son.

Setyaningsih , E. (2009). Jurnal Teknologi. *Penyandian Citra Menggunakan Metode Playfair Cipher*, 212-217.

Setyaningsih, E. (2015). *Kriptografi & Implementasi menggunakan matlab.* Bandung: Penerbit Andi.

Siambaton, M. Z. (2018). Modifikasi Algoritma Playfair Cipher dengan pengurutan Array pada matrik ISSN 2598-6341. 66-71.

Soyomukti, N. (2012). *Pengantar Ilmu Komunikasi.* Yogyakarta: AR-RUZ MEDIA.

Yusuf, A. (2018). JURNAL USU. *Implementasi algoritma Rail fence dengan LSB.*

LAMPIRAN

Enkripsi dan dekripsi *Playfair Cipher- Rail Fence Cipher*

```
def key(kun):
    Kunci = kun
    Kunci = Kunci.replace(" ", "")
    Kunci = Kunci.upper()

    Kunci_tereduksi = list()

    for i in Kunci:  # mereduksi karakter yang sama pada kunci
        if i not in Kunci_tereduksi:
            if i == "Q":
                Kunci_tereduksi.append("I")
            else:
                Kunci_tereduksi.append(i)

    x = 0
    for i in range(65, 91):  # menambah karakter lain di kunci tereduksi
        if chr(i) not in Kunci_tereduksi:
            if i == 73 and chr(81) not in Kunci_tereduksi:
                Kunci_tereduksi.append("I")
                x = 1
            elif x == 0 and i == 73 or i == 81:
                pass
            else:
```

```
                Kunci_tereduksi.append(chr(i))

    Initial_matrix = [[0 for i in range(5)] for j in range(5)]

    k = 0
    for i in range(0, 5):  # Membuat matriks 5x5 dengan kunci tereduksi
        for j in range(0, 5):
            Initial_matrix[i][j] = Kunci_tereduksi[k]
            k += 1
    return Initial_matrix

def locindex(c):  #Fungsi posisi karakter
    loc = list()
    if c == 'Q':
        c = 'I'
    for i, j in enumerate(Initial_matrix):
        for k, l in enumerate(j):
            if c == l:
                loc.append(i)
                loc.append(k)
                return loc

def encryptPF(plain):  # Enkripsi Playfair chiper
    msg = plain
    msg = msg.upper()
    msg = msg.replace(" ", "")
```

```
    i = 0
    chiphertext = ''
    for s in range(0, len(msg) + 1, 2):
        if s < len(msg) - 1:
            if msg[s] == msg[s + 1]:
                msg = msg[:s + 1] + 'X' + msg[s + 1:]
    if len(msg) % 2 != 0:
        msg = msg[:] + 'X'
    while i < len(msg):
        loc = list()
        loc = locindex(msg[i])
        loc1 = list()
        loc1 = locindex(msg[i + 1])
        if loc[1] == loc1[1]:
            chiphertext = chiphertext + Initial_matrix[(loc[0] + 1) % 5][loc[1]] + Initial_matrix[(loc1[0] + 1) % 5][loc1[1]]
        elif loc[0] == loc1[0]:
            chiphertext = chiphertext + Initial_matrix[loc[0]][(loc[1] + 1) % 5] + Initial_matrix[loc1[0]][(loc1[1] + 1) % 5]
        else:
            chiphertext = chiphertext + Initial_matrix[loc[0]][loc1[1]] + Initial_matrix[loc1[0]][loc[1]]
        i = i + 2
    return chiphertext


def decryptPF(chiper):  # dekripsi Playfair chiper
```

```
    msg = chiper
    msg = msg.upper()
    msg = msg.replace(" ", "")
    plaintext = ''
    i = 0
    while i < len(msg):
        loc = list()
        loc = locindex(msg[i])
        loc1 = list()
        loc1 = locindex(msg[i + 1])
        if loc[1] == loc1[1]:
            plaintext = plaintext + Initial_matrix[(loc[0] - 1) % 5][loc[1]] + Initial_matrix[(loc1[0] - 1) % 5][loc1[1]]
        elif loc[0] == loc1[0]:
            plaintext = plaintext + Initial_matrix[loc[0]][(loc[1] - 1) % 5] + Initial_matrix[loc1[0]][(loc1[1] - 1) % 5]
        else:
            plaintext = plaintext + Initial_matrix[loc[0]][loc1[1]] + Initial_matrix[loc1[0]][loc[1]]
        i = i + 2
    return plaintext



def encryptRF(plain, key, offset=0): #Enkripsi RailFence
    fence = [[] for i in range(key)]
    rail = offset
    var = 1
```

```
    for char in plain:
        fence[rail].append(char)
        rail += var

        if rail == key - 1 or rail == 0:
            var = -var

    res = ''
    for i in fence:
        for j in i:
            res += j

    return res

def decryptRF(cipher, key, offset=0):  #dekripsi RailFence
    plain = ''

    if offset:
        t = encryptRF('o' * offset + 'x' * len(cipher), key)
        for i in range(len(t)):
            if (t[i] == 'o'):
                cipher = cipher[:i] + '#' + cipher[i:]

    length = len(cipher)
    fence = [['#'] * length for _ in range(key)]
```

```
    i = 0
    for rail in range(key):
        p = (rail != (key - 1))
        x = rail
        while (x < length and i < length):
            fence[rail][x] = cipher[i]
            if p:
                x += 2 * (key - rail - 1)
            else:
                x += 2 * rail
            if (rail != 0) and (rail != (key - 1)):
                p = not p
            i += 1



    for i in range(length):
        for rail in range(key):
            if fence[rail][i] != '#':
                plain += fence[rail][i]
    return plain


while (1): #program untuk memilih
    choice = int(input("==OPSI PROGRAM==\n 1.Enkripsi \n 2.Dekripsi \n 3.KELUAR \n Masukkan Pilihanmu : "))
    if choice == 1: #Pilihan Enkripsi
```

```
        a = input('Masukan Plaintext :')
        b = input('Masukan Kunci Penyandian Playfair Chiper : ')
        c = int(input('Masukan Kunci Penyandian Rail Fence Chiper : '))
        d = int(input('Masukan Offset Penyandian Rail Fence Chiper : '))
        Initial_matrix = key(b)
        chipertextPF = encryptPF(a)
        chipertextRF = encryptRF(chipertextPF, c, d)
        result = chipertextRF

print("================================================")
        print("Hasil Super Enkripsi Playfair Cipher dan Rail Fence Cipher")
        print(result)

print("================================================")


    elif choice == 2:
        a = input('Masukan Chipertext :')
        b = input('Masukan Kunci Penyandian Playfair Chiper : ')
        c = int(input('Masukan Kunci Penyandian Rail Fence Chiper : '))
        d = int(input('Masukan Offset Penyandian Rail Fence Chiper : '))
        Initial_matrix = key(b)
        plaintextRF = decryptRF(a,c,d)
        plaintextPF = decryptPF(plaintextRF)
        result = plaintextPF

print("================================================")
```

```
        print("Hasil Dekripsi dari Super Enkripsi Playfair Cipher dan Rail Fence Cipher")
        print(result)

print("================================================")
    elif choice == 3:
        exit()
    else:
        print("Choose correct choice")
```

RIWAYAT HIDUP

Ahmad Zaini, Lahir di Pasuruan 15 Maret 1995, Tinggal Desa Jatiarjo RT 035 RW 017 Kecamatan Prigen Kabupaten Pasuruan. Anak Sulung dari 2 bersaudara, Putra dari pasangan bapak Abd. Wahab dan Ibu Yama. Mulai menempuh Pendidikan dasar di MI Miftahul Ulum Tonggowa, menempuh pendidikan di Mts NU Mifathul Ulum Tonggowa, dan menempuh MA NU Sunan Giri Talang. Selanjutnya pada tahun 2014 melanjutkan Kuliah di Universitas Islam Negeri Maulana Malik Ibrahim Malang, Fakultas Sains dan Teknologi, mengambil Program Studi Matematika.

KEMENTRIAN 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 : Ahmad Zaini
NIM : 14610098
Fakultas/Jurusan : Sains dan Teknologi/ Matematika
Judul Skripsi : Implementasi Algoritma Super Enkripsi Pada Pengaman Pesan Berbentuk Teks
Pembimbing I : Muhammad Khudzaifah, M.Si
Pembimbing II : Juhari, M.Si

No.	Tanggal	Hal	Tanda Tangan	
1	14 September 2020	Konsultasi BAB I & II	1	
2	23 September 2020	Revisi BAB I & II		2
3	27 September 2020	ACC BAB I & II	3	
4	28 Januari 2021	Konsultasi BAB I, II & III		4
5	2 Februari 2021	Revisi BAB I, II & III	5	
6	15 Maret 2021	ACC BAB I, II & III		6
7	18 Maret 2021	Revisi BAB IV	7	
8	15 April 2021	Revisi BAB I, II , III & IV		8
9	22 April 2021	ACC BAB I, II , III & IV	9	
10	01 Mei 2021	Konsultasi Keagamaan		10
11	04 Mei 2021	ACC Keagamaan	11	
12	7 Mei 2021	ACC Keseluruhan		12

Malang, 07 Mei 2021
Mengetahui,
Ketua Program Studi Matematika

Dr. Usman Pagalay, M.Si
NIP. 19650414 200312 1 001