MODIFIKASI VIGENERE CIPHER MENGGUNAKAN GRUP SIMETRI UNTUK MENGAMANKAN PESAN TEKS

SKRIPSI

OLEH
NIKEN DWI CAHYANTI
NIM. 17610087

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2022

MODIFIKASI VIGENERE CIPHER MENGGUNAKAN GRUP SIMETRI UNTUK MENGAMANKAN PESAN 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
Niken Dwi Cahyanti
NIM. 17610087

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2022

MODIFIKASI VIGENERE CIPHER MENGGUNAKAN GRUP SIMETRI UNTUK MENGAMANKAN PESAN TEKS

SKRIPSI

Oleh
Niken Dwi Cahyanti
NIM. 17610087

Telah Diperiksa dan Disetujui untuk Diuji
Malang, 15 Juni 2022

Dosen Pembimbing I	Dosen Pembimbing II
Prof. Dr. H. Turmudi, M.Si., Ph.D	Muhammad Khudzaifah, M.Si
NIP. 19571005 198203 1 006	NIDT. 19900511 20160801 1 057

Mengetahui,
Ketua Program Studi Matematika

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

MODIFIKASI VIGENERE CIPHER MENGGUNAKAN GRUP SIMETRI UNTUK MENGAMANKAN PESAN TEKS

SKRIPSI

Oleh
Niken Dwi Cahyanti
NIM. 17610087

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S.Mat)
Tanggal, 20 Juni 2022

Ketua Penguji	: Intan Nisfulaila, M.Si	.....
Anggota Penguji 1	: Dewi Ismiarti, M.Si	.....
Anggota Penguji 2	: Prof. Dr. H. Turmudi, M.Si., Ph.D	.....
Anggota Penguji 3	: Muhammad Khudzaifah, M.Si	.....

Mengetahui,
Ketua Program Studi Matematika

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

PERNYATAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Niken Dwi Cahyanti

NIM : 17610087

Jurusan : Matematika

Fakultas : Sains dan Teknologi

Judul Skripsi : Modifikasi Vigenere Cipher Menggunakan Grup Simetri untuk Mengamankan Pesan Teks

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya 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 dikemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 23 Juni 2022

Yang membuat pernyataan,

Niken Dwi Cahyanti

NIM. 17610087

MOTO DAN PERSEMBAHAN

“Sesungguhnya bersama kesulitan ada kemudahan. Maka apabila engkau telah selesai (dari sesuatu urusan), tetaplah bekerja keras (untuk urusan yang lain)” (QS. Al-Insyirah: 6-7).

Skripsi ini penulis persembahkan untuk keluarga penulis terutama kepada kedua orang tua yaitu Bapak Supardi dan Ibu Sulastri yang senantian memberikan do’a, dukungan, serta semangat kepada penulis.

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Segala Puji dan syukur kehadirat Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, dan shalawat serta salam selalu tercurahkan kepada junjungan kita Nabi Muhammad SAW, sehingga penulis mampu menyelesaikan penulisan skripsi ini dengan judul "Modifikasi Vigenere Cipher Menggunakan Grup Simetri untuk Mengamankan Pesan Teks" sebagai salah satu syarat untuk memperoleh gelar sarjana dalam bidang matematika di Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Selesainya penyusunan skripsi ini tidak lepas dari bimbingan serta bantuan dari berbagai pihak, karena itu penulis mengucapkan terima kasih yang sebesar-besarnya dan penghargaan setinggi-tingginya kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim.
2. Dr. Sri Harini, M.Si, selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim.
3. Dr. Elly Susanti, S.Pd., M.Sc, selaku ketua Program Studi Matematika, Universitas Islam Negeri Maulana Malik Ibrahim.
4. Prof. Dr. H. Turmudi, M.Si., Ph.D. selaku Dosen Pembimbing I yang telah memberikan bimbingan serta arahan kepada penulis.
5. Muhammad Khudzaifah, M.Si, selaku Dosen Pembimbing II yang telah memberikan bimbingan serta arahan kepada penulis.
6. Dewi Ismiarti, M.Si, selaku Penguji Utama dalam Ujian Skripsi.
7. Intan Nisfulaila, M.Si, selaku Ketua Penguji dalam Ujian Skripsi.
8. Seluruh dosen Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim yang telah memberikan ilmu pengetahuan kepada penulis.
9. Orang tua dan seluruh keluarga penulis yang telah memberikan dukungan serta do'a kepada penulis.

10. Seluruh mahasiswa Program Studi Matematika angkatan 2017 serta semua pihak yang tidak dapat penulis sebutkan satu persatu, penulis ucapkan terima kasih atas bantuannya dalam menyelesaikan penyusunan skripsi ini.

Akhirnya penulis berharap semoga skripsi ini dapat bermanfaat tidak hanya bagi penulis tetapi juga bermanfaat bagi semua pihak yang membaca skripsi ini.

Wassalamu'alaikum Wr.Wb.

Malang, 23 Juni 2022

Penulis

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR SIMBOL

C_i	*ciphertext* ke- i
P_i	*plaintext* ke- i
K_r	kunci ke – r
$\mathbb{Z}$	bilangan bulat
$a \mid b$	a habis membagi b
$a \nmid b$	a tidak habis membagi b
$(\boldsymbol{G}, *)$	grup dengan operasi biner $*$
S_{n}	grup simetri, $n =$ banyaknya anggota himpunan yang dipermutasikan
$\circ$	operasi komposisi
$\sigma \in S_n$	suatu permutasi σ yang merupakan elemen dari S_n
f^{-1}	invers dari fungsi f
I	fungsi identitas atau permutasi identitas
$\lvert\sigma\rvert$	orde dari suatu elemen σ

DAFTAR LAMPIRAN

ABSTRAK

Cahyanti, Niken Dwi. 2022. **Modifikasi Vigenere Cipher Menggunakan Grup Simetri untuk Mengamankan Pesan Teks.** Skripsi. Jurusan Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. H.Turmudi, M.Si., Ph.D., (II) Muhammad Khudzaifah, M.Si.

Kata Kunci: Modifikasi, Vigenere Cipher, Grup Simetri.

Kriptografi banyak digunakan untuk mengatasi masalah keamanan informasi yang mengalami pertukaran pada jaringan internet. Salah satu algoritma dalam kriptografi adalah Vigenere Cipher, algoritma tersebut terkenal sebagai metode yang tangguh dan tidak mudah dipecahkan. Namun, algoritma Vigenere Cipher memiliki kelemahan yaitu kuncinya yang pendek dan digunakan berulang-ulang. Oleh karena itu, perlu dilakukan modifikasi terhadap Vigenere Cipher. Tujuan penelitian ini yaitu untuk mengetahui modifikasi Vigenere Cipher menggunakan grup simetri dan juga untuk mengetahui tingkat keamanan hasil enkripsi modifikasi Vigenere Cipher jika dibandingkan dengan Vigenere Cipher. Hasil yang diperoleh dari penelitian ini adalah suatu algortima baru dari proses modifikasi Vigenere Cipher menggunakan grup simetri. Algoritma modifikasi Vigenere Cipher terbukti lebih kuat jika dibandingkan dengan Vigenere Cipher. Hal tersebut dikarenakan *plaintext* diacak terlebih dahulu menggunakan penyandian grup simetri, sehingga *plaintext* yang sebenarnya aman terhadap serangan metode kasiski dan *exhaustive key search*. Selain itu, algoritma baru yang dihasilkan juga mendukung penggunaan huruf kapital, huruf kecil, angka, dan simbol.

ABSTRACT

Cahyanti, Niken Dwi. 2022. **Modify Vigenere Cipher Using Symmetric Group for Message Text Security.** Thesis Department of Mathematics, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University. Advisor: (I) Prof. Dr. H.Turmudi, M.Si., Ph.D., (II) Muhammad Khudzaifah, M.Si.

Keyword: Modification, Vigenere Cipher, Symmetric Group

Cryptography is widely used to overcome information security problems that are exchanged on the internet network. One of the algorithms in cryptography is the Vigenere Cipher, the algorithm is known as a robust method and is not easily solved. However, the Vigenere Cipher algorithm has a weakness, namely the key is short and is used repeatedly. Therefore, it is necessary to modify the Vigenere Cipher. The purpose of this study is to determine the modification of the Vigenere Cipher using a symmetric group and also to determine the level of security of the encryption results of Vigenere Cipher modification when compared to the Vigenere Cipher. The results obtained from this research is a new algorithm of the modification process of the Vigenere Cipher using symmetric groups. The modified Vigenere Cipher algorithm is proven to be stronger than the Vigenere Cipher. This is because the plaintext is scrambled first using a symmetric group encoding, so that the actual plaintext is safe against attacks by the Kasiski method and exhaustive key search. In addition, the resulting new algorithm also supports the use of capital letters, lowercase letters, numbers, and symbols.

مستخلص البحث

جهيانتي، نيكين دوي. ٢٠٢٢. **تعديل فيجينير شفرات (*Vigenere Cipher*) باستخدام مجموعات التماثل (*Symmetric Group*) لتأمين الرسائل النصية**. البحث العلمي. قسم الرياضيات، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية بمالانج. المشرف : (١) البروفيسور، الدكتور، تورمودي، الماجستير، الحاج، (٢) محمد خذيفة، الماجستير.

الكلمات الأساسية: تعديل، فيجينير شفرات (*Vigenere Cipher*) ، مجموعات التماثل

يستخدم التشفير على نطاق واسع للتغلب على مشاكل أمن المعلومات التي يتم تبادلها على شبكة الإنترنت. واحدة من الخوارزميات في التشفير هي فيجينير شفرات (*Vigenere Cipher*)، وتعرف الخوارزمية بأنها طريقة قوية ولا يمكن حلها بسهولة. ولكن، خوارزمية فيجينير شفرات (*Vigenere Cipher*) بها نقطة ضعف، وهي أن المفتاح قصير ويستخدم بشكل متكرر. لذلك، من الضروري تعديل فيجينير شفرات (*Vigenere Cipher*). الغرض من هذا البحث هو لمعرفة تعديل فيجينير شفرات (*Vigenere Cipher*) باستخدام مجموعة متماثلة وأيضًا لمعرفة مستوى أمان تشفير فيجينير شفرات المعدل عند مقارنته بفيجينير شفرات (*Vigenere Cipher*). النتائج التي تم الحصول عليها من هذا البحث عبارة عن خوارزمية جديدة لعملية تعديل فيجينير شفرات (*Vigenere Cipher*) باستخدام المجموعات المتماثلة. ثبت أن خوارزمية فيجينير شفرات المعدلة أقوى من فيجينير شفرات (*Vigenere Cipher*). هذا لأن النص العادي يتم خلطه أولاً باستخدام ترميز مجموعة متماثل (*Symmetric Group*)، بحيث يكون النص العادي الفعلي آمنًا ضد هجمات طريقة كاسكي والبحث الشامل عن المفتاح. بالإضافة إلى ذلك، تدعم الخوارزمية الجديدة الناتجة أيضًا استخدام الأحرف الكبيرة والأحرف الصغيرة والأرقام والرموز.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Semakin pesatnya perkembangan teknologi informasi pada zaman sekarang, semakin banyak pula tindakan kejahatan penyalahgunaan informasi melalui jaringan internet. Hal tersebut dapat terjadi karena jaringan internet sering kali digunakan untuk proses pertukaran informasi. Oleh sebab itu, diperlukan peningkatan keamanan terhadap kerahasiaan suatu informasi yang mengalami pertukaran pada jaringan internet. Salah satu informasi yang sering dipertukarkan yaitu data atau pesan dalam bentuk teks. Menjaga keamanan data dapat dilakukan menggunakan metode penyandian. Teknik menyandikan pesan teks digunakan dengan tujuan supaya isi pesan tidak diketahui oleh orang yang tidak berwenang. Pesan rahasia tersebut sama halnya dengan amanat yang harus disampaikan kepada yang berhak menerimanya. Sebagaimana firman Allah SWT dalam Al-Qur'an surat An-Nisaa' ayat 58 dijelaskan bahwa:

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

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

Menjaga keamanan informasi menjadi hal yang harus diperhatikan. Kriptografi sebagai salah satu metode pengamanan data untuk meminimalisir banyaknya tindak kejahatan penyalahgunaan informasi yang bersifat penting dan rahasia (Hariati, Hardiyanti, & Putri, 2018). Kriptografi adalah seni dan ilmu yang

digunakan untuk mengamankan pesan yang dikirim oleh pengirim dari suatu tempat ke tempat penerima, agar tidak jatuh ke tangan pihak yang tidak berwenang (Mukhtar, 2018). Menjaga keamanan menggunakan kriptografi, terdapat proses penyandian pesan yang disebut enkripsi yaitu dengan melakukan perubahan teks asli (*plaintext*) ke dalam bentuk teks sandi (*ciphertext*) dengan menggunakan suatu algoritma. Sedangkan proses untuk mengembalikan *ciphertext* ke dalam bentuk *plaintext* disebut dekripsi (Rachmawati, Budiman, & Aulia, 2018).

Proses enkripsi dan dekripsi membutuhkan suatu protokol perjanjian kunci yaitu kesepakatan mengenai kunci rahasia yang dilakukan oleh pihak pengirim dan pihak penerima pesan sehingga kedua belah pihak dapat menyepakati suatu kunci rahasia yang sama. Penelitian Stickel (2005) memperkenalkan algoritma protokol perjanjian kunci yang didasarkan pada grup non-komutatif. Salah satu contoh grup non-komutatif yang dapat digunakan pada protokol perjanjian kunci Stickel yaitu grup simetri-n. Penelitian dengan menerapkan grup simetri-n pada algoritma pembentukan kunci telah dilakukan sebelumnya oleh Wasiatun Rizkiyah (2016). Penelitian tersebut melakukan proses enkripsi dan dekripsi menggunakan teknik transposisi dengan kunci yang telah terbentuk melalui proses pembentukan kunci atas grup simetri-n.

Vigenere Cipher termasuk di dalam jenis algoritma kriptografi klasik yang juga algoritma simetris artinya pada proses penyandiannya digunakan kunci yang sama. Vigenere Cipher dulunya terkenal sebagai metode yang tangguh dan sulit dipecahkan, hingga akhirnya dapat dipecahkan menggunakan metode Friedman dan Kasiski (Aliyu & Olaniyan, 2016). Kedua metode tersebut dapat digunakan oleh seorang kriptanalis untuk mengetahui panjangnya kunci yang dipakai pada

Vigenere Cipher. Setelah panjang kunci diketahui, maka kunci bisa ditemukan dengan metode *exhaustive key search* atau analisis frekuensi (Munir, 2019). Selain itu, Vigenere Cipher memiliki kelemahan yaitu kuncinya yang pendek dan digunakan berulang-ulang (Ariyus, 2008). Oleh sebab itu, sangat perlu melakukan modifikasi pada Vigenere Cipher untuk membuat keamanan lebih kuat dan tidak mudah dipecahkan.

Penelitian memodifikasi Vigenere Cipher sudah pernah dilakukan sebelumnya oleh Octavianingrum, Siambaton, dan Dewi (2018). Penelitian tersebut memodifikasi Vigenere Cipher dengan pembuatan kunci geser dan teknik enkripsi blok pada proses enkripsinya. Penelitian lain juga dilakukan oleh Widarma, Siregar, dan Irawan (2019). Penelitian tersebut menggabungkan Vigenere Cipher dan *Electronic Code Book* (ECB) untuk meningkatkan keamanan. Kedua penelitian tersebut menghasilkan algoritma baru yang lebih aman jika dibandingkan dengan Vigenere Cipher biasa. Akan tetapi, kedua penelitian tersebut hanya dapat mengenkripsi *plaintext* berupa huruf kapital saja. Karakter huruf kecil, angka, dan simbol belum dibahas pada kedua penelitian tersebut.

Berbeda dengan penelitian Latifah, Ambo, dan Kurnia (2017) yang memodifikasi Caesar Cipher dan Reilfence Cipher. Penelitian tersebut memodikasi algoritma Caesar Cipher dengan menggunakan ASCII *printable characters* dan modulus 95. Penelitian tersebut juga melakukan perubahan pada rumus enkripsi dan dekripsi pada Caesar Cipher serta menggabungkan Caesar Cipher dengan algoritma lain. Penelitian tersebut menghasilkan algoritma baru yang lebih kuat dibandingkan Caesar Cipher biasa dan mampu mengenkripsi huruf kapital, huruf kecil, simbol, dan angka. Caesar Cipher dan Vigenere Cipher memiliki rumus

enkripsi dan dekripsi yang sama, walaupun memiliki perbedaan pada kuncinya baik Caesar Cipher maupun Vigenere Cipher kunci yang digunakan selalu dalam lingkup yang sama. Oleh karena itu, pada penelitian ini akan digunakan ASCII *printable characters* untuk menutupi kekurangan dari kedua penelitian sebelumnya.

Penelitian ini memodifikasi Vigenere Cipher menggunakan grup simetri untuk mengamankan pesan teks. Penyandian grup simetri digunakan untuk mengacak *plaintext* sebelum dioperasikan dengan kunci Vigenere Cipher. Hal tersebut bertujuan untuk menyembunyikan *plaintext* yang sebenarnya. Penelitian ini diharapkan mampu meningkatkan keamanan pada algoritma Vigenere Cipher, sehingga menghasilkan algoritma baru yang lebih aman jika dibandingkan dengan Vigenere Cipher biasa.

1.2 Rumusan Masalah

Rumusan masalah dalam penelitian yang dilakukan yaitu sebagai berikut:

1. Bagaimana modifikasi algoritma Vigenere Cipher menggunakan grup simetri untuk mengamankan pesan teks?
2. Bagaimana tingkat keamanan hasil enkripsi modifikasi Vigenere Cipher jika dibandingkan dengan Vigenere Cipher?

1.3 Tujuan Penelitian

Tujuan penelitian berdasarkan rumusan masalah yaitu sebagai berikut:

1. Untuk mengetahui modifikasi algoritma Vigenere Cipher menggunakan grup simetri untuk mengamankan pesan teks.

2. Untuk mengetahui tingkat keamanan hasil enkripsi modifikasi Vigenere Cipher jika dibandingkan dengan Vigenere Cipher.

1.4 Manfaat Penelitian

Hasil dari penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Bagi Penulis

   Memperdalam wawasan penulis tentang Vigenere Cipher dan grup simetri serta mendapat pengetahuan tentang cara mengamankan pesan teks menggunakan modifikasi Vigenere Cipher dengan grup simetri.
2. Bagi Pembaca

   Sebagai referensi atau informasi mengenai modifikasi Vigenere Cipher menggunakan grup simetri.

1.5 Batasan Masalah

Batasan masalah pada penelitian ini yaitu:

1. Karakter yang digunakan berupa huruf, angka, dan karakter khusus berdasarkan ASCII *printable characters*.
2. Pembentukan kunci untuk proses penyandian grup simetri yaitu dengan menggunakan algoritma protokol perjanjian kunci Stickel atas grup simetri S_n.
3. Grup simetri yang digunakan pada penerapan algoritma baru yang dihasilkan yaitu grup simetri S_5.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penelitian ini terbagi menjadi lima bab dan masing-masing bab terbagi ke dalam subbab sebagai berikut:

Bab I Pendahuluan

Bagian bab pendahuluan ini berisi latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika penulisan.

Bab II Kajian Pustaka

Bab kajian pustaka ini berisi penjelasan tentang aritmatika modulo, keterbagian, kongruensi, fungsi, definisi grup, permutasi, grup simetri, orde dari suatu elemen, pengertian dan sejarah kriptografi, algoritma kriptografi, teknik transposisi (permutasi), protokol perjanjian kunci, Vigenere Cipher, kriptanalisis Vigenere Cipher, dan kajian keislaman.

Bab III Metode Penelitian

Bab metode penelitian ini berisi tentang jenis penelitian, pra penelitian, dan tahapan penelitian.

Bab IV Pembahasan

Bagian pembahasan ini menjelaskan secara keseluruhan langkah-langkah yang disebutkan dalam metode penelitian dan menjawab rumusan masalah.

Bab V Penutup

Bagian punutup ini berisi tentang kesimpulan yang diperoleh dari pembahasan dan saran yang ingin disampaikan.

BAB II

KAJIAN PUSTAKA

2.1 Aritmatika Modulo

Aritmatika modulo memainkan peran penting dalam perhitungan bilangan bulat, karena aritmatika modulo merupakan dasar dari komputasi bilangan bulat. Adapun tujuan digunakannya aritmatika modulo dalam operasi aritmatika yaitu supaya menghasilkan bilangan bulat di dalam ruang lingkup yang sama. Contoh penerapan aritmatika modulo yaitu pada algoritma kriptografi klasik yang menggunakan huruf alfabet. Diketahui bahwa huruf alfabet terdiri dari huruf "A" sampai huruf "Z", kemudian langkah awal dilakukan dengan memetakan {A,….,Z} ke {0,….,25}. Penggunaan aritmatika modulo pada kriptografi klasik bertujuan agar transformasi penyandian selalu berada dalam lingkup yang sama yakni bernilai {0,….,25}, sehingga hasil dari proses penyandian selalu memiliki pasangan karakter (Sadikin, 2012).

Operator yang digunakan pada aritmatika modulo adalah mod. Operasi modulo membutuhkan dua input yaitu bilangan bulat z dan modulo n dimana $n > 0$. Operasi modulo menghasilkan r, dimana r adalah sisa bagi atas z dibagi n (Sadikin, 2012). Operator modulo dinotasikan dalam persamaan berikut.

$$z \; mod \; n = r \qquad (2.1)$$

Contoh 2.1

(i) Temukan hasil operasi dari $31 \; mod \; 3$!

Jawab: 31 dibagi 3 adalah 10 sisa 1, $31 = (3 \times 10) + 1$ jadi $31 \; mod \; 3 = 1$

(ii) Temukan hasil operasi dari $44\ mod\ 4$!

Jawab: 44 dibagi 4 adalah 11 sisa 0, $44 = (4 \times 11) + 0$ jadi $44\ mod\ 4 = 0$

2.2 Keterbagian

Sifat-sifat yang berhubungan dengan keterbagian (*divisibility*) dapat dikatakan sebagai dasar dari pengembangan teori bilangan. Berikut adalah definisi keterbagian:

Definisi 2.1

Misalkan $a, b \in \mathbb{Z}$ dengan $a \neq 0$. Bilangan bulat a dikatakan habis membagi b, yang dinotasikan dengan $a|b$, jika dan hanya jika $b = ax$, untuk suatu $x \in \mathbb{Z}$. Bilangan bulat a disebut pembagi dari b dan b disebut kelipatan dari a (Abdussakir, 2009).

Contoh 2.2

(i) Misal $a = 2$ dan $b = 14$

Maka $2|14$, karena $14 \div 2 = 7 \in \mathbb{Z}$, sedemikian hingga $14 = 2 \cdot 7$

(ii) Misal $a = 2$ dan $b = 5$

Maka $2 \nmid 5$, karena $5 \div 2 = 2.5$ (bukan bilangan bulat)

2.3 Kongruensi

Salah satu konsep yang penting dalam teori bilangan adalah kongruensi. Definisi kongruensi bilangan bulat adalah sebagai berikut.

Definisi 2.2

Misalkan $a, b,$ dan m bilangan bulat dengan $m > 0$. Bilangan bulat a disebut kongruen dengan b modulo m jika $m|(a - b)$ dan ditulis $a \equiv b \ (mod\ m)$. Jika a tidak kongruen dengan b modulo m, maka ditulis $a \not\equiv b \ (mod\ m)$ (Munir, 2019).

Contoh 2.3

(i) $10 \equiv 7 \ (mod\ 3)$ karena $3|(10 - 7)$

(ii) $-5 \not\equiv 24 \ (mod\ 4)$ karena $4 \nmid (-5 - 24)$

2.4 Fungsi

Definisi 2.3

Misalkan S dan T merupakan himpunan tak kosong. Fungsi dari S ke T adalah suatu aturan yang memasangkan setiap anggota di himpunan S tepat satu ke anggota di himpunan T (Bartle & Sherbert, 2000).

Definisi 2.4

Misalkan $f: S \to T$ merupakan suatu fungsi dari S ke T.

(i) Fungsi f dapat dikatakan fungsi injektif (satu-satu) jika untuk setiap $x_1 \neq x_2$ akan berlaku $f(x_1) \neq f(x_2)$. Untuk membuktikan bahwa suatu fungsi f adalah fungsi injektif, kita harus menunjukkan bahwa untuk semua $x_1, x_2 \in S$, jika $f(x_1) = f(x_2)$, maka $x_1 = x_2$.

(ii) Fungsi f dapat dikatakan fungsi surjektif (pada) jika $f(S) = T$, yaitu jika daerah hasil $R(f) = B$. Artinya, untuk setiap $t \in T$ terdapat $x \in S$ sedemikian sehingga $f(x) = t$.

(iii) Fungsi f dapat dikatakan fungsi bijektif (satu-satu dan pada) jika f merupakan fungsi injektif (satu-satu) dan surjektif (pada) (Bartle & Sherbert, 2000).

Definisi 2.5 Fungsi Invers

Jika $f: S \to T$ adalah suatu fungsi bijektif (satu-satu dan pada) dari S ke T, maka dapat ditemukan balikan atau inversi (invers) dari f, dinotasikan dengan f^{-1}, yang memetakan $T\ ke\ S$ sebagai berikut: Misalkan x merupakan anggota dari himpunan S dan t merupakan anggota dari himpunan T, maka $f^{-1}(t) = x$ jika $f(x) = t$ (Munir, 2019).

2.5 Grup

Misalkan grup $(\boldsymbol{G}, *)$ adalah pasangan berurutan dengan $\boldsymbol{G}$ adalah himpunan tak kosong dan $*$ adalah operasi biner di $\boldsymbol{G}$ yang memenuhi syarat berikut:

1. Operasi $*$ bersifat asosiatif di G, yakni
   $(p * q) * r = p * (q * r), \forall\ p, q, r \in G$
2. Mempunyai elemen $e \in G$ yang merupakan elemen identitas dari G, sehingga
   $$p * e = e * p = p, \forall\ p \in G$$
3. Memiliki elemen $p^{-1} \in G$ yang merupakan elemen invers dari p, untuk semua $p \in G$ sehingga $p * p^{-1} = p^{-1} * p = e$ (Dummit & Foote, 2004).

Teorema 2.1

Misalkan p merupakan elemen dari grup G, misal m dan n adalah bilangan bulat. Maka,

$$p^m \cdot p^n = p^{m+n}$$

(Gilbert & Gilbert, 2015)**.**

Bukti:

Misalkan m adalah *arbitrary*, tetapi ditetapkan bahwa $m \in \mathbb{Z}$. Terdapat tiga kasus untuk n:

i. $n = 0$

ii. n adalah bilangan bulat positif

iii. n adalah bilangan bulat negatif

Pertama, misal $n = 0$ untuk kasus i. Maka,

$p^m \cdot p^n = p^m \cdot p^0 = p^m \cdot e = p^m$ dan $p^{m+n} = p^{m+0} = p^m$

Jadi $p^m \cdot p^n = p^{m+n}$ benar untuk $n = 0$

Kedua, kita gunakan induksi untuk $n > 0$ untuk kasus ii. Jika $n = 1$, kita punya

$p^m \cdot p^n = p^m \cdot p^1 = p^{m+1} = p^{m+n}$

Jadi benar untuk $n > 0$ ketika $n = 1$. Asumsikan bahwa $p^m \cdot p^n = p^{m+n}$ benar untuk $n = k$, sehingga $p^m \cdot p^k = p^{m+k}$. Kemudian, untuk $n = k + 1$, kita punya

$p^m \cdot p^n = p^m \cdot p^{k+1}$

$= p^m \cdot (p^k \cdot p)$ berdasarkan definisi dari x^{k+1}

$= (p^m \cdot p^k) \cdot p$ sifat asosiatif

$= p^{m+k} \cdot p^1$ berdasarkan hipotesis induksi

$= p^{m+k+1}$ berdasarkan definisi dari $x^{(m+k)+1}$

$= p^{m+n}$ karena $n = k + 1$

Jadi benar untuk $n = k + 1$, sehingga benar untuk n adalah bilangan bulat positif.

Ketiga, mmepertimbangkan kasus iii untuk n adalah bilangan bulat negatif.

Artinya $n = -a$, dimana a adalah bilangan bulat positif. Terdapat tiga kemungkinan untuk a, yaitu:

$a = m, a < m$, dan $m < a$.

Jika $a = m$, maka $n = -a = -m$, dan kita punya $p^m \cdot p^n = p^m \cdot p^{-m} = e$ berdasarkan pernyataan **a** pada **Teorema 3.9** dalam buku Linda Gilbert dan Jimmie Gilbert (2015), dan

$$p^{m+n} = p^{m-m} = p^0 = e$$

Kita punya $p^m \cdot p^n = p^{m+n}$ ketika $a = m$

Jika $a < m$, misal $m - a = b$, sehingga $m = a + b$ dimana a dan b bilangan bulat positif. Karena telah dibuktikan $p^m \cdot p^n = p^{m+n}$ ketika m dan n bilangan bulat positif, sehingga kita gunakan $p^a \cdot p^b = p^{a+b}$.

$$\begin{aligned} p^m \cdot p^n &= p^{a+b} \cdot p^{-a} \\ &= p^a \cdot p^b \cdot p^{-a} \\ &= p^b \cdot e \\ &= p^b = p^{b+a-a} \\ &= p^{m+n} \end{aligned}$$

Jadi benar $p^m \cdot p^n = p^{m+n}$ untuk kasus dimana $a < m$.

Anggap bahwa $m < a$. Misalkan $c = a - m$, sehingga c adalah bilangan positif dan $a = m + c$. Berdasarkan definisi dari p^{-a}, maka

$$\begin{aligned} p^{-a} = (p^{-1})^a &= (p^{-1})^{m+c} \\ &= (p^{-1})^m \cdot (p^{-1})^c \qquad \text{karena } m \text{ dan } c \text{ adalah bilangan positif} \\ &= p^{-m} \cdot p^{-c} \end{aligned}$$

Subtitusikan nilai p^{-a} ke dalam $p^m \cdot p^n = p^m \cdot p^{-a}$, kita punya

$$\begin{aligned} p^m \cdot p^n &= p^m \cdot (p^{-m} \cdot p^{-c}) \\ &= (p^m \cdot p^{-m}) \cdot p^{-c} \\ &= e \cdot p^{-c} = p^{-c} \end{aligned}$$

Kita juga punya

$$p^{m+n} = p^{m-a} = p^{m-(m+c)} = p^{-c}$$

Sehingga, benar $p^m \cdot p^n = p^{m+n}$ untuk kasus dimana $m < a$.

Telah dibuktikan bahwa $p^m \cdot p^n = p^{m+n}$ benar untuk kasus dimana m adalah bilangan bulat positif dan n adalah bilangat bulat ($n = 0, n < 0, n > 0$).

Pembuktian ini belum lengkap untuk kasus dimana $m = 0$ dan m adalah bilangan bulat negatif. Pembuktian untuk kasus tersebut sama seperti pembuktian pada kasus $n = 0$ dan n adalah bilangan bulat negatif.

Jadi **Teorema 2.1** terbukti.

2.6 Grup Permutasi

2.6.1 Permutasi

Berikut ini diberikan definisi permutasi dari suatu himpunan dan definisi grup permutasi dari suatu himpunan.

Definisi 2.6 Permutasi dari Ω, Grup Permutasi dari Ω

Permutasi dari suatu himpunan Ω adalah suatu fungsi dari Ω ke Ω yang satu-satu dan onto (bijektif). Sedangkan grup permutasi dari suatu himpunan Ω adalah himpunan permutasi-permutasi dari Ω yang membentuk grup dengan fungsi komposisi (Gallian, 2017).

Misalkan $\Omega = \{a_1, a_2, \dots, a_n\}$ dan σ adalah fungsi bijektif dari Ω ke Ω, maka σ adalah suatu permutasi berderajat n.

Misalnya $\sigma(a_1) = b_1, \sigma(a_2) = b_2, \dots, \sigma(a_n) = b_n$, dimana $\{b_1, b_2, \dots, b_n\} = \{a_1, a_2, \dots, a_n\}$, kedua himpunan tersebut sama tetapi memiliki urutan elemen yang

berbeda. Selanjutnya permutasi dapat dinotasikan dengan matriks dua baris, peta dari setiap elemennya ditulis tepat di bawahnya seperti berikut ini.

$$\sigma = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ \sigma(a_1) & \sigma(a_2) & \dots & \sigma(a_n) \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ b_1 & b_2 & \dots & b_n \end{pmatrix}$$

Permutasi Identitas. Misalkan himpunan Ω memuat n elemen-elemen yang berbeda, fungsi identitas I dari himpunan Ω pada dirinya sendiri disebut permutasi identitas berderajat n. Jika $\Omega = \{a_1, a_2, \dots, a_n\}$, maka permutasi identitas berderajat n yaitu (Raisinghania & Aggarwal, 1980):

$$I = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ a_1 & a_2 & \dots & a_n \end{pmatrix}$$

Invers Permutasi. Misalkan σ adalah permutasi berderajat n, didefinisikan atas himpunan berhingga Ω yang memuat n elemen-elemen yang berbeda. Maka berdasarkan definisi permutasi, σ adalah fungsi bijektif dari Ω ke dirinya sendiri. Sekarang, fungsi σ bijektif dan dapat diinverskan. Akibatnya, invers fungsi σ ada dan berdasarkan definisi, fungsi tersebut juga fungsi bijektif dari Ω ke dirinya sendiri dan dinotasikan dengan σ^{-1}. Sehingga, σ^{-1} juga permutasi berderajat n didefinisikan atas Ω dan dikenal sebagai invers permutasi σ.

Jika σ adalah suatu permutasi dari elemen-elemen himpunan Ω

$$\sigma = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ b_1 & b_2 & \dots & b_n \end{pmatrix}$$

Maka dengan menukar baris dari σ kita peroleh σ^{-1}, yaitu (Raisinghania & Aggarwal, 1980):

$$\sigma^{-1} = \begin{pmatrix} b_1 & b_2 & \dots & b_n \\ a_1 & a_2 & \dots & a_n \end{pmatrix}$$

2.6.2 Grup Simetri

Misalkan Ω merupakan himpunan tak kosong. Misalkan S_Ω himpunan yang memuat semua fungsi-fungsi bijektif dari Ω ke dirinya sendiri. Dengan kata lain, himpunan S_Ω yang memuat semua permutasi pada Ω. Himpunan S_Ω dengan komposisi fungsi ∘ atau (S_Ω,∘) adalah suatu grup. Operasi komposisi ∘ adalah operasi biner di S_Ω, karena $\delta: \Omega \to \Omega$ dan $\gamma: \Omega \to \Omega$ keduanya merupakan fungsi bijektif, maka $\delta \circ \gamma$ juga fungsi bijektif dari Ω ke dirinya sendiri. Secara umum komposisi fungsi bersifat asosiatif, jadi operasi ∘ juga memenuhi sifat asosiatif. Identitas dari S_Ω mempunyai elemen identitas yaitu permutasi 1 yang didefinisikan dengan $1(a) = a, \forall a \in \Omega$. Terdapat fungsi invers $\delta^{-1}: \Omega \to \Omega$ untuk setiap permutasi $\delta: \Omega \to \Omega$ yang memenuhi $\delta \circ \delta^{-1} = \delta^{-1} \circ \delta = 1$. Dengan demikian, ($S_\Omega$,∘) memenuhi semua syarat grup. Grup (S_Ω,∘) disebut grup simetri pada himpunan Ω. Elemen-elemen dari S_Ω merupakan permutasi dari Ω, bukan elemen dari Ω itu sendiri. Kasus khusus ketika $\Omega = \{1, 2, 3, \dots, \mathrm{n}\}$, grup simetri pada Ω dinotasikan dengan S_n, grup simetri berderajat n. Orde dari S_n adalah n! (Dummit & Foote, 2004).

Karena fungsi dari Ω ke Ω adalah fungsi bijektif, maka fungsi bijektif tersebut dapat dinyatakan dalam bentuk permutasi. Himpunan S_Ω merupakan himpunan semua permutasi dari Ω ke Ω dengan komposisi fungsi disebut grup permutasi. Oleh karena itu, grup simetri (S_Ω,∘) dapat dikatakan sebagai grup permutasi. Suatu sikel $\sigma = (a_1 a_2 \dots a_\mathrm{m})$ adalah permutasi yang memetakan a_i ke $a_{\mathrm{i}+1}, 1 \leq i \leq m-1$ dan memetakan a_m ke a_1. Sebagai contoh, (2 1 3) adalah suatu permutasi yang memetakan 2 ke 1, 1 ke 3 dan 3 ke 2. Panjang sikel adalah banyaknya elemen yang terdapat dalam sikel tersebut. Suatu sikel dengan panjang

p disebut sikel-p dan dua sikel dikatakan saling lepas jika elemen dari suatu sikel tidak berada pada sikel yang lain dan sebaliknya (Dummit & Foote, 2004).

Contoh 2.4

Grup simetri berderajat lima S_5 dengan operasi komposisi $(S_5,\circ)$ yang didefinisikan sebagai himpunan semua fungsi bijektif dari {1, 2, 3, 4, 5} ke dirinya sendiri.

Misalkan permutasi a dari himpunan {1, 2, 3, 4, 5} dengan menetapkan $a(1)=3, a(2)=5, a(3)=1, a(4)=2, a(5)=4$ atau dapat dituliskan dalam bentuk sebagai berikut:

$$a=\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ a(1) & a(2) & a(3) & a(4) & a(5) \end{pmatrix}$$

$$a=\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 5 & 1 & 2 & 4 \end{pmatrix}$$

Adapun cara mengoperasi permutasi dengan fungsi komposisi yaitu sebagai berikut:

Misalkan diberikan $a=\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 5 & 1 & 2 & 4 \end{pmatrix}$ dan $b=\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \end{pmatrix}$, maka

$$a\circ b=\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 5 & 1 & 2 & 4 \end{pmatrix}\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \end{pmatrix}$$

$$a\circ b=\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 1 & 2 & 4 & 3 \end{pmatrix}$$

Hasil operasi $a\circ b$ dapat dituliskan dalam notasi sikel menjadi (1532) (4)

Sebagai contoh tambahan, misalkan $a, b \in S_n$ adalah sebagai berikut:

$$a=\begin{pmatrix} 1 & 2 & 3 & 4 & \dots & n \\ 3 & 2 & 1 & 4 & \dots & n \end{pmatrix} \qquad b=\begin{pmatrix} 1 & 2 & 3 & 4 & \dots & n \\ 2 & 1 & 3 & 4 & \dots & n \end{pmatrix}$$

Maka

$$b \circ a = \begin{pmatrix} 1 & 2 & 3 & 4 & \dots & n \\ 3 & 1 & 2 & 4 & \dots & n \end{pmatrix}$$

$$a \circ b = \begin{pmatrix} 1 & 2 & 3 & 4 & \dots & n \\ 2 & 3 & 1 & 4 & \dots & n \end{pmatrix}$$

Sehingga, $a \circ b \neq b \circ a$.

Hal tersebut menunjukkan bahwa grup simetri S_{n} merupakan grup tidak komutatif untuk $n \geq 3$ (Dummit & Foote, 2004). Sedangkan $S_1 = \{(1)\}$ merupakan grup komutatif, karena S_1 hanya memuat elemen identitas. Grup simetri $S_2 = \{(1), (12)\}$ juga merupakan grup komutatif, karena S_2 hanya memiliki dua elemen dan $(1)(12) = (12)(1)$.

2.7 Orde dari Suatu Elemen

Misal $\boldsymbol{G}$ adalah sebarang grup dan σ merupakan sebarang elemen dari $\boldsymbol{G}$. Orde dari suatu elemen σ yaitu suatu bilangan bulat positif terkecil k sedemikian hingga $\sigma^k = \boldsymbol{e}$ ($\boldsymbol{e}$ adalah elemen identitas di G), dan dapat dinotasikan dengan $|\sigma| = k$ (Raisinghania dan Aggarwal, 1980).

Contoh 2.5

Grup simetri berderajat tiga S_3 dengan operasi komposisi (S_3,$\circ$) yang didefinisikan sebagai himpunan semua fungsi bijektif dari {1,2,3} ke dirinya sendiri. Elemen-elemen dari grup simetri S_3 ditetapkan sebagai berikut:

$$\sigma_1 = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{pmatrix} = (13)(2) \qquad \sigma_2 = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix} = (132)$$

$$\sigma_3 = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \end{pmatrix} = (123) \qquad \sigma_4 = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix} = (12)(3)$$

$$\sigma_5 = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \end{pmatrix} = (1)(2)(3) \qquad \sigma_6 = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 3 & 2 \end{pmatrix} = (1)(2,3)$$

Elemen σ_5 merupakan elemen identitas, maka

$\sigma_1 \circ \sigma_1 = (1)(2)(3) = \sigma_5$, maka orde dari elemen σ_1 adalah $|\sigma_1| = 2$

$\sigma_2 \circ \sigma_2 \circ \sigma_2 = \sigma_3 \circ \sigma_2 = (1)(2)(3) = \sigma_5$, maka orde dari elemen σ_2 adalah $|\sigma_2| = 3$

$\sigma_3 \circ \sigma_3 \circ \sigma_3 = \sigma_2 \circ \sigma_3 = (1)(2)(3) = \sigma_5$, maka orde dari elemen σ_3 adalah $|\sigma_3| = 3$

$\sigma_4 \circ \sigma_4 = (1)(2)(3) = \sigma_5$, maka orde dari elemen σ_4 adalah $|\sigma_4| = 2$

$\sigma_5 = (1)(2)(3)$, maka orde dari elemen σ_5 adalah $|\sigma_5| = 1$

$\sigma_6 \circ \sigma_6 = (1)(2)(3) = \sigma_5$, maka orde dari elemen σ_6 adalah $|\sigma_6| = 2$

Jadi dapat disimpulkan

$|\sigma_5| = 1,\ |\sigma_1| = |\sigma_4| = |\sigma_6| = 2$, dan $|\sigma_2| = |\sigma_3| = 3$

Orde dari suatu elemen di grup simetri S_{n} dapat lebih mudah ditentukan menggunakan Teorema berikut ini (lebih jelasnya lihat **Teorema 5.3** dan buktinya pada buku *Contemporary Abstract Algebra*).

Teorema 2.2

Orde dari suatu permutasi pada grup simetri, yang ditulis dalam bentuk perkalian sikel-sikel saling lepas, adalah kelipatan persekutuan terkecil (KPK) dari panjang sikel-sikel tersebut (Ruffini dalam Gallian, 2017).

Bukti:

Pertama, amati suatu sikel dengan panjang n yang mempunyai orde n.

Asumsikan α dan β adalah sikel saling lepas yang panjangnya m dan n, dan misalkan k adalah KPK dari m dan n. Berdasarkan **Teorema 4.1** (teorema dan bukti dapat dilihat di buku *Contemporary Abstract Algebra*) maka α^k dan β^k adalah permutasi identitas ε, dan karena α dan β *commute*, sehingga $(\alpha\beta)^k = \alpha^k\beta^k = \varepsilon\varepsilon = \varepsilon$. Berdasarkan **Corollary 2** (beserta bukti dapat dilihat di buku *Contemporary Abstract Algebra*) maka $|\alpha\beta|$ atau mari kita sebut dengan t harus membagi k.

Tetapi kemudian $(\alpha\beta)^t = \alpha^t\beta^t = \varepsilon$, sedemikian hingga $\alpha^t = \beta^{-t}$. Namun, jika elemen dari suatu sikel α tidak berada pada sikel β dan sebaliknya (saling lepas), maka α^t dan β^{-t} juga saling lepas, karena suatu sikel yang dipangkatkan tidak memperkenalkan elemen baru. Tetapi, jika α^t dan β^{-t} adalah sama dan saling lepas, maka $\alpha^t = \beta^{-t} = \varepsilon$, sebab tiap elemen di dalam α^t ditetapkan oleh β^{-t} dan sebaliknya. Karena $|\alpha|$ dan $|\beta|$ harus membagi t, maka m dan n harus membagi t. Ini berarti k harus membagi t juga. Ini menunjukkan bahwa $k = t$.

Contoh 2.6

$a_1 = (13)(2)$ → KPK dari 2 dan 1 adalah 2. Jadi orde dari a_1 adalah $|a_1| = 2$

$a_2 = (132)$ → KPK dari 3 adalah 3. Jadi orde dari a_2 adalah $|a_2| = 3$

$a_3 = (123)$ → KPK dari 3 adalah 3. Jadi orde dari a_3 adalah $|a_3| = 3$

$a_4 = (12)(3)$ → KPK dari 2 dan 1 adalah 2. Jadi orde dari a_4 adalah $|a_4| = 2$

$a_5 = (1)(2)(3)$ → KPK dari 1, 1, 1 adalah 1. Jadi orde dari a_5 adalah $|a_5| = 1$

$a_6 = (1)(23)$ → KPK dari 2 dan 1 adalah 2. Jadi orde dari a_6 adalah $|a_6| = 2$

Jadi dapat disimpulkan

$|a_5| = 1,\ |a_1| = |a_4| = |a_6| = 2$, dan $|a_2| = |a_3| = 3$

2.8 Kriptografi

2.8.1 Pengertian dan Sejarah Kriptografi

Kriptografi berasal dari Bahasa Yunani, yakni kripto dan grafik, kripto berarti rahasia (secret) dan grafia berarti tulisan (writing). Sedangkan menurut termologi, kriptografi adalah seni dan ilmu yang mempelajari tentang teknik-teknik matematis yang berkaitan dengan aspek keamanan informasi seperti integrasi data, autentifikasi kerahasiaan, dan autentikasi keaslian data. Tingkat keamanan dalam

kriptografi adalah ukuran kekuatan yang dapat dicapai oleh suatu algoritma kriptografi dalam menjaga informasi dari serangan kriptanalisis. Kriptografi digunakan untuk menjaga keamanan pesan yang dikirim dari satu tempat ke tempat yang lain agar tidak jatuh ketangan pihak yang tidak berwenang. Perkembangan kriptografi sangat pesat, hal tersebut ditandai dengan banyaknya penyelesaian masalah keamanan data yang dihadapi dengan kriptografi (Mukhtar, 2018).

Sejarah kriptografi dimulai pada zaman Romawi Kuno, ketika Julian Caesar ingin mengirimkan suatu pesan rahasia untuk seorang Jendral di medan perang. Julian Caesar memikirkan suatu cara agar pesan rahasia tersebut tidak terbuka di tengah jalan, karena Julian Caesar ingin hanya Jendral saja yang paham isi pesan rahasia tersebut. Pastinya Jendral tersebut diberi tahu terlebih dahalu cara membaca pesan yang teracak tersebut, beserta dengan kuncinya. Julian Caesar mengacak pesan tersebut dengan cara mengganti semua susunan alfabet dari a, b, c, kemudian diubah a menjadi d, b menjadi e, c menjadi f, dan seterusnya (Ariyus, 2006).

Teknik kriptografi yang sudah dikembangkan sejak zaman dahulu bahkan sebelum ditemukannya komputer disebut kriptografi klasik. Walaupun demikian, sampai zaman ini teknik tersebut masih sering dipakai untuk mengamankan informasi. Kriptografi klasik dapat dibedakan menjadi dua macam yaitu *cipher* substitusi dan *cipher* transposisi. *Cipher* substitusi adalah algoritma kriptografi yang cara kerjanya yaitu dengan mengganti setiap karakter *plaintext* dengan salah satu karakter *ciphertext*. Sedangkan *cipher* transposisi itu sendiri adalah metode kriptografi dengan mengubah urutan karakter dalam *plaintext* atau melakukan *transpose* terhadap rangkaian karakter (Latifah, Ambo, & Kurnia, 2017).

Menjaga keamanan pesan dengan kriptografi, teks pesan (*plaintext*) diubah menjadi teks sandi menggunakan suatu algoritma dan kunci rahasia. Pengirim pesan menentukan kunci rahasia atau kata sandi kemudian dengan menggunakan fungsi enkripsi (yang biasanya menyertakan beberapa rumus matematika) mengubah angka-angka tersebut ke dalam bentuk teks sandi (*ciphertext*). Teks sandi dan kunci rahasia dikirim ke penerima. Penerima dapat menggunakan fungsi dekripsi untuk mengubah teks sandi ke dalam bentuk teks asli (Rachmawati, Budiman, & Aulia, 2018). Proses menyandikan pesan disebut enkripsi (*encryption*) sedangkan proses mengembalikan teks sandi ke teks asli disebut dengan dekripsi (*decyption*). Pesan awal yang belum diacak ataupun yang telah didekripsi disebut dengan teks asli (*plaintext*), sedangkan pesan yang telah diacak disebut dengan teks sandi (*ciphertext*) (Mukhtar, 2018).

2.8.2 Algoritma Kriptografi

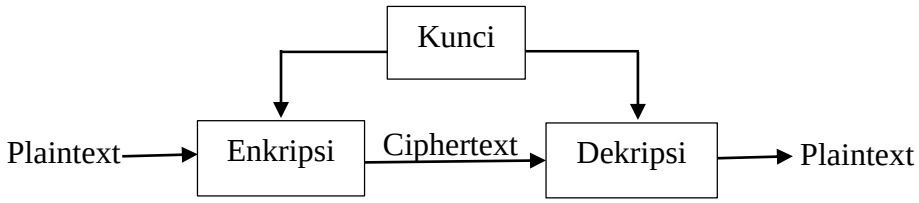
Algortima adalah tahap-tahap atau urutan yang tersusun secara matematis untuk menyelesaikan suatu permasalahan. Sedangkan algoritma kriptografi adalah tahapan-tahapan bagaimana cara menyembunyikan isi pesan rahasia dari orang-orang yang tidak berwenang akan pesan tersebut (Ariyus, 2008).

Berdasarkan kunci yang dipakainya, algoritma kriptografi terbagi menjadi tiga bagian yaitu:

1. Algoritma Simetri

   Algoritma simetri atau kunci rahasia merupakan algoritma kriptografi yang memakai satu kunci yang sama untuk melakukan enkripsi dan dekripsi. Oleh karena itu pengirim pesan dan penerima pesan harus menjaga benar-benar

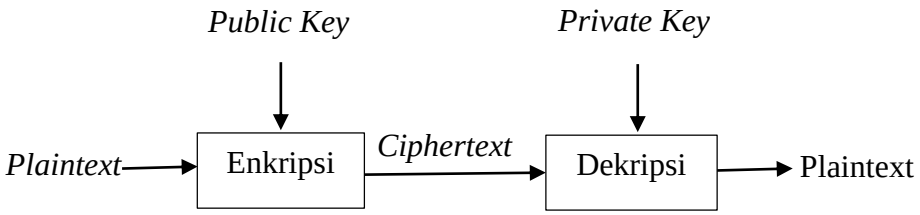
kerahasiaan kunci supaya tidak jatuh ke tangan orang yang tidak berwenang, jika kunci diketahui maka orang lain tersebut dapat melakukan proses enkripsi dan dekripsi pesan (Ariyus, 2008). Adapun beberapa algoritma yang telah dikembangkan oleh para ahli kriptografi memakai algoritma simetri diantaranya adalah *One Time Pad* (OTP), *Data Encryptions Standart* (DES), *Advance Encryptions Standart* (AES), dan Vigenere Cipher.



Gambar 2.1 Algoritma Simetri

2. Algoritma Asimetri

Algoritma asimetri atau algoritma kunci publik mempunyai dua kunci yang berbeda pada proses enkripsi dan dekripsinya. Terdapat dua macam kunci pada algortima asimetri, yaitu: kunci yang boleh diketahui oleh semua orang (kunci umum atau *public key*) dan kunci yang disembunyikan (kunci rahasia atau *private key*). Kedua kunci tersebut memiliki hubungan satu dengan lainnya. Algoritma kunci publik lebih aman daripada algoritma simetri, karena dengan menggunakan algortima kunci publik seorang kriptanalis dapat mengenkripsi pesan dengan kunci umum namun tidak dapat mendekripsi pesan karena kunci rahasia hanya dimiliki oleh penerima pesan saja. Terdapat beberapa algoritma yang dikembangkan oleh para ahli kriptografi memakai algoritma asimetri, diantaranya RSA, Diffle-Hellman (DH), *Digital Signature Algorithm* (DSA), kriptografi Quantum, *Elliptic Curve Cryptography* (ECC), dan lain-lain.



Gambar 2.2 Algoritma Asimetri

3. Fungsi Hash

Fungsi hash adalah suatu fungsi matematika yang mengambil masukan panjang variabel lalu mengubahnya ke dalam bentuk biner dengan panjang yang tepat. Nama lain dari fungsi hash yaitu fungsi satu arah *(one-way function), fingerprint*, fungsi kompresi, *message digest*, dan *message authentication code* (MAC). Fungsi hash digunakan pada saat seseorang ingin membuat sidik jari dari suatu pesan. Sidik jari dari pesan itu sendiri bertujuan untuk memberi tanda bahwasanya pesan yang diterima memang benar-benar dari pengirim (Ariyus, 2008).

2.8.3 Teknik Transposisi (Permutasi)

Teknik transposisi (permutasi) ini memakai permutasi karakter, dengan memakai teknik ini pesan asli (*plaintext*) tidak dapat diketahui isinya kecuali seseorang tersebut mempunyai kunci untuk mendekripsi pesan kembali ke bentuk asal (Ariyus, 2006). Penyandian pesan menggunakan teknik transposisi (permutasi) yaitu dengan cara melakukan permutasi k terhadap pesan yang asli, sedangkan untuk mengembalikan pesan ke bentuk asal dilakukan dengan invers permutasi k^{-1} pada teks sandi (Sadikin, 2012).

1. Proses Penyandian Menggunakan Teknik Transposisi (Permutasi)

Langkah pertama dari proses penyandian menggunakan teknik transposisi (permutasi) yaitu dengan membagi teks asli (*plaintext*) menjadi blok-blok yang

memuat beberapa karakter. Kunci yang digunakan pada teknik ini adalah bentuk permutasi- k, yang dapat menjadikan pesan asli tidak dapat diketahui kecuali oleh seseorang yang mempunyai kunci untuk mendekripsi *ciphertext* ke bentuk asal (Ariyus, 2006).

2. Proses Dekripsi Menggunakan Teknik Transposisi (Permutasi)

Proses dekripsi pada teknik transposisi ini pada dasarnya sama saja dengan proses enkripsinya, namun pada proses dekripsi penerima pesan terlebih dahulu menginverskan kunci yang telah disepakati sebelumnya.

Berikut ini contoh proses penyandian mengunakan teknik permutasi.

Terdapat kunci untuk melakukan permutasi:

x	a_1	a_2	a_3	a_4	a_5
$k(x)$	a_4	a_5	a_1	a_2	a_3

dan juga terdapat kunci invers permutasi:

x	a_1	a_2	a_3	a_4	a_5
$k^{-1}(x)$	a_3	a_4	a_5	a_1	a_2

Misalkan akan dilakukan permutasi atau enkripsi terhadap teks di bawah ini:

SAYA SEDANG MENCOBA ENKRIPSI

Selanjutnya membagi teks tersebut menjadi lima blok dengan ketentuan apabila terdapat kekurangan pada blok dapat ditambahkan dengan karakter seperti ∅, sedangkan untuk spasi dapat diganti dengan karakter seperti #.

SAYA# SEDAN G#MEN COBA#

ENKRI PSI∅∅

Setelah itu setiap blok diubah menjadi seperti di bawah ini dengan menggunakan kunci pertama yang telah ditentukan.

Blok 1: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_5 & a_1 & a_2 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ S & A & Y & A & \# \\ a_4 & a_5 & a_1 & a_2 & a_3 \\ A & \# & S & A & Y \end{pmatrix}$

Blok: 2 $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_5 & a_1 & a_2 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ S & E & D & A & N \\ a_4 & a_5 & a_1 & a_2 & a_3 \\ A & N & S & E & D \end{pmatrix}$

Blok 3: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_5 & a_1 & a_2 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ G & \# & M & E & N \\ a_4 & a_5 & a_1 & a_2 & a_3 \\ E & N & G & \# & M \end{pmatrix}$

Blok 4: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_5 & a_1 & a_2 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ C & O & B & A & \# \\ a_4 & a_5 & a_1 & a_2 & a_3 \\ A & \# & C & O & B \end{pmatrix}$

Blok 5: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_5 & a_1 & a_2 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & N & K & R & I \\ a_4 & a_5 & a_1 & a_2 & a_3 \\ R & I & E & N & K \end{pmatrix}$

Blok 6: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_5 & a_1 & a_2 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ P & S & I & \% & \% \\ a_4 & a_5 & a_1 & a_2 & a_3 \\ \emptyset & \emptyset & P & S & I \end{pmatrix}$

Sehingga diperoleh *ciphertext* yaitu

A#SAYANSEDENG#MA#COBRIENKØØPSI

Sedangkan untuk proses dekripsi *ciphertext* dilakukan dengan cara yang sama seperti proses enkripsi namun dengan menggunakan kunci invers yang telah ditentukan sebelumnya.

2.8.4 Protokol Perjanjian Kunci

Protokol perjanjian kunci merupakan salah satu teknik dalam kriptografi yang digunakan untuk mengatasi masalah keamanan pengiriman informasi rahasia melalui jalur komunikasi yaitu pihak pengirim dan pihak penerima pesan harus menyepakati kunci yang sama. Pada teknik ini, antara pihak pengirim dan penerima pesan akan saling menukarkan parameter. Parameter tersebut dapat diketahui khalayak umum, namun dari parameter tersebut akan dihasilkan suatu kunci rahasia yang sama dan tidak dapat diketahui oleh khalayak umum (Musthofa & Lestari, 2014). Salah satu algortima protokol perjanjian kunci yang terkenal adalah protokol perjanjian kunci Stickel yang menggunakan grup non-komutatif. Grup $(G,\circ)$ dikatakan grup non-komutatif apabila fungsi komposisi "$\circ$" tidak memenuhi sifat komutatif (Myasnikov, Shpilrain, & Ushakov, 2008).

Misalkan G adalah suatu grup tidak komutatif dan $\sigma_1, \sigma_2 \in G$ dengan R orde dari σ_1 dan S merupakan orde dari σ_2, sehingga $\sigma_1 \circ \sigma_2 \neq \sigma_2 \circ \sigma_1$. Proses perjanjian kunci yang dilakukan oleh pengirim dan penerima adalah (Myasnikov, Shpilrain, & Ushakov, 2008):

1. Pengirim memilih sebarang bilangan asli dengan syarat $M < R$, $N < S$ dan mengirim $r = {\sigma_1}^M {\sigma_2}^N$.
2. Penerima memilih sebarang bilangan asli dengan syarat $P < R$, $Q < S$ dan mengirim $s = {\sigma_1}^P {\sigma_2}^Q$
3. Pengirim menerima s dari penerima selanjutnya menghitung $K_1 = {\sigma_1}^M s {\sigma_2}^N$
4. Penerima menerima r dari pengirim selanjutnya menghitung $K_2 = {\sigma_1}^P r {\sigma_2}^Q$

Sehingga dapat diperoleh kunci rahasia yang telah disepakati oleh kedua pihak yaitu

$$K_1 = {\sigma_1}^M s{\sigma_2}^N = {\sigma_1}^M {\sigma_1}^P {\sigma_2}^Q {\sigma_2}^N = {\sigma_1}^{M+P} {\sigma_2}^{Q+N} = {\sigma_1}^P {\sigma_1}^M {\sigma_2}^N {\sigma_2}^Q = {\sigma_1}^P r {\sigma_2}^Q = K_2$$

Jadi, $K = K_1 = K_2$

2.8.5 Vigenere Cipher

Vigenere Cipher adalah salah satu jenis algoritma kriptografi klasik yang berbasis karakter dan juga termasuk di dalam algoritma simetris yaitu algoritma yang menggunakan kunci yang sama dalam proses enkripsi dan dekripsinya. Nama Vigenere Cipher diambil sesuai dengan penemunya, Blaise de Vigenere. Algoritma ini dulunya merupakan metode yang tangguh dan sulit dipecahkan, hingga akhirnya dapat dipecahkan menggunakan metode Friedman dan Kasiski (Aliyu & Olaniyan, 2016). Selain itu, Vigenere Cipher juga memiliki kelemahan yaitu kuncinya yang pendek dan digunakan berulang-ulang. Metode dari subtitusi Vigenere untuk menghasilkan *ciphertext* dapat dilakukan dengan dua cara yaitu angka dan huruf. Metode subtitusi menggunakan angka dengan cara menukar huruf dengan angka. Sedangkan metode subtitusi mengunakan huruf, bisa digunakan *tabula recta* (disebut juga bujursangkar) seperti **Tabel 2.1** Tabula Recta Vigenere (Ariyus, 2008). Baris paling atas dari **Tabel 2.1** menyatakan karakter-karakter *plaintext*, sedangkan kolom paling kiri menyatakan karakter-karakter kunci. Setiap baris di dalam bujursangkar merupakan karakter-karakter *ciphertext*. Penggunaan tabula recta untuk melakukan enkripsi yaitu dengan cara menarik garis vertikal dari karakter *plaintext* ke bawah, lalu tarik garis horizontal dari karakter kunci ke kanan. Perpotongan kedua garis tersebut merupakan karakter *ciphertext*. Sedangkan untuk melakukan dekripsi pesan, dilakukan dengan menarik garis secara horizontal dari

karakter kunci ke karakter *ciphertext*, kemudian dari karakter *ciphertext* ditarik garis vertikal ke atas sampai karakter *plaintext* yang dituju (Munir, 2019).

Tabel 2.1 Tabula recta Vigenere (Ariyus, 2008)

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

Secara matematis proses enkripsi dan dekripsi menggunakan Vigenere Cipher. Misalkan kunci dengan panjang n merupakan rangkaian $K_1, \dots, K_n$, *plaintext* merupakan rangkaian $P_1, \dots, P_m$, dan *ciphertext* merupakan rangkaian $C_1, \dots, C_m$, maka enkripsi Vigenere Cipher dapat dituliskan persamaan sebagai berikut ini:

$$C_i = (P_i + K_r)\ mod\ 26 \qquad (1 \leq i \leq m) \qquad (2.2)$$

Sedangkan untuk proses dekripsi dapat menggunakan persamaan berikut (Munir, 2019):

$$P_i = (C_i - K_r)\ mod\ 26 \qquad (1 \leq i \leq m) \qquad (2.3)$$

dan

$$i \equiv r(mod\ n) \qquad (1 \leq r \leq m) \qquad (2.4)$$

Keterangan:

P_i = Karakter ke-i pada *plaintext*

C_i = Karakter ke-i pada *ciphertext*

K_r = Karakter ke-r pada kunci

(Catatan: apabila kunci kurang dari *plaintext* maka kunci akan diulang hingga seluruh *plaintext* tersebut terpenuhi. Apabila kunci melebihi *plaintext*, maka kunci akan ditulis sesuai dengan panjang *plaintext* yang diperlukan).

2.8.6 Kriptanalisis Vigenere Cipher

Kriptanalisis merupakan bidang ilmu dan seni untuk memecahkan *ciphertext*. Kriptanalisis pertama kali dikemukakan oleh Al-Kindi pada abad ke-9. Kriptanalisis yang dilakukan oleh Al-Kindi dikenal dengan metode analisis frekuensi. Vigenere Cipher tergolong cipher subtitusi, salah satu kelemahan cipher subtitusi yaitu tidak dapat menyembunyikan hubungan statistik antara karakter *plaintext* dengan kemunculan karakter pada *ciphertext*. Apabila suatu karakter sering muncul pada *plaintext* maka akan sering muncul pula pada *ciphertext*. Kelemahan tersebut yang dimanfaatkan untuk memecahkan *ciphertext* menggunakan metode analisis frekuensi. Kriptanalisis dengan metode analisis frekuensi dilakukan menggunakan tabel frekuensi kemunculan huruf-huruf di

dalam suatu bahasa untuk menerka *plaintext*. Misalkan dalam Bahasa Inggris, frekuensi kemunculan huruf dapat dilihat pada tabel berikut ini.

Tabel 2.2 Frekuensi Kemunculan Huruf dalam Bahasa Inggris

Huruf	%	Huruf	%	Huruf	%
A	8.2	J	0.1	S	6.3
B	1.5	K	0.8	T	9.0
C	2.8	L	4.0	U	2.8
D	4.2	M	2.4	V	1.0
E	12.7	N	6.7	W	2.4
F	2.2	O	7.5	X	2.0
G	2.0	P	1.9	Y	0.1
H	6.1	Q	0.1	Z	0.1
I	7.0	R	6.0		

Tabel 2.2 memperlihatkan bahwa huruf paling sering muncul dalam Bahasa Inggris adalah E, T, A, O, I, N, S, H, R, D, dan L. Selain frekuensi kemunculan huruf, juga terdapat bigram dan trigram yang sering muncul dalam Bahasa Inggris yaitu he, in, an, en, th, nt, es, er, re, ti, dan 10 trigram yaitu the, and, for, has, tha, ent, ing, ion, tio, nde.

Langkah-langkah melakukan kriptanalisis menggunkan metode analisis frekuensi untuk *plaintext* dalam bahasa Inggris yaitu sebagai berikut:

1. Misalkan panjang kunci telah berhasil diketahui adalah *m*. Setiap karakter kelipatan-*m* pasti disandikan dengan karakter kunci yang sama. Lakukan pengelompokkan setiap kelipatan *m* karakter sehingga menjadi *m* buah "pesan", masing-masing disandikan dengan subtitusi alphabet-tunggal (dalam hal ini Caesar Cipher).

2. Setiap pesan dari langkah 1 dapat dibobol menggunakan metode analisis frekuensi.
3. Hasil dari langkah 2, kriptanalis dapat menyusun karakter-karakter kunci atau dapat menerka suatu kata yang membantu memecahkan .

Apabila tidak diketahui panjang kuncinya, Vigenere Cipher sebenarnya aman dari analisis frekuensi. Namun, kelemahan utama dari Vigenere Cipher adalah kunci yang digunakan berulang. Hal tersebut dimanfaatkan oleh Friedrich Kasiski untuk menemukan panjang kunci Vigenere Cipher. Pada tahun 1863 Friedrich Kasiski berhasil memecahkan Vigenere Cipher, tetapi sebenarnya cara serupa sudah dikembangkan oleh Charles Babbage pada tahun 1854. Metode Kasiski menggunakan keuntungan dalam Bahasa Inggris yaitu selain mengandung perulangan huruf juga mengandung perulangan pasangan huruf atau tripel huruf, contohnya seperti TH, THE, dan lain sebagainya.

Langkah-langkah untuk menemukan panjang kunci menggunakan metode Kasisiki yaitu sebagai berikut:

1. Temukan semua kriptogram yang berulang pada *ciphertext* (kriptogram yang berulang biasanya terdapat pada pesan teks yang panjang). Setelah itu, hitunglah jarak antara kriptogram yang berulang.
2. Jarak yang diperoleh dari langkah 1 dihitung faktor pembaginya (faktor pembagi menyatakan panjang kunci yang mungkin). Tentukan irisan dari semua himpunan faktor pembagi, nilai dari irisan tersebut merupakan kemungkinan panjang kunci.

Setelah panjang kunci berhasil ditemukan, maka karakter kunci dapat dicari menggunakan metode analisis frekuensi atau secara *exhaustive key search,* yaitu

dengan cara mencoba semua kemungkinan kunci yang panjangnya sama kemudian periksa apakah hasil dekripsi merupakan pesan memiliki makna. Semua kemungkinan kunci yang harus dicoba yaitu sebanyak 26^p, dengan p menyatakan panjang kunci. Sehingga semakin panjang kunci yang digunakan, maka semakin banyak pula kemungkinan kunci yang harus dicoba (Munir, 2019).

2.9 Kajian Keislaman

Kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan dari satu tempat ke tempat yang lain. Menjaga keamanan dan kerahasiaan data menjadi sangat penting saat data memiliki nilai (Mukhtar, 2018). Pesan bersifat rahasia artinya pesan tersebut sengaja disembunyikan agar tidak diketahui oleh orang lain. Jika seseorang ingin menyampaikan pesan rahasia kepada seseorang yang dituju dan tidak ingin pesan diketahui orang lain maka diperlukan algoritma penyandian pesan. Pesan rahasia juga dapat disebut sebagai amanah yang mana harus disampaikan kepada yang berhak menerimanya saja. Menjaga amanah dapat dijadikan sebagai tujuan dari merahasiakan data.

Amanah dalam arti sempit yaitu memelihara titipan dan mengembalikan kepada pemiliknya dalam keadaan semula, sedangkan dalam arti yang luas amanah yaitu menyimpan rahasia orang, menjaga kehormatan orang lain, menjalankan tugas-tugas yang diberikan kepadanya dan lain sebagainya.

Amanah artinya dapat dipercaya, kata tersebut seakar dengan kata iman. Sifat amanah itu sendiri lahir dari kekuatan iman, semakin tipisnya kekuatan iman seseorang maka semakin pudar pula sifat amanah dalam dirinya. Sehingga terdapat kerterkaitan antara iman dan amanah. Sebagaimana sabda Rasulullah SAW “tidak

sempurna iman seseorang yang tidak amanah, dan tidak sempurna agama orang yang tidak menunaikan janji" (HR. Ahmad). Oleh sebab itu, menjaga amanah sangatlah penting.

Adapun ayat tentang amanah di dalam Al-Qur'an, salah satunya yaitu surat Al-Ma'arij ayat 32 yang artinya:

وَالَّذِيْنَ هُمْ لِاَمٰنٰتِهِمْ وَعَهْدِهِمْ رٰعُوْنَ (32)

"dan orang-orang yang memelihara amanah-amanah (yang dipikulnya) dan janjinya". (QS.Al-Ma'arij:32)

Ayat ini menjelaskan tentang orang-orang yang memelihara dan tidak mengkhianati perkara-perkara agama dan dunia yang diamanahkan kepada mereka dan apa yang dijanjikan orang lain kepadanya. Menurut Muhammad Nassib Ar-Rifa'I dalam buku Ringkasan Ibnu Katsir Jilid 4 menjelaskan orang-orang yang memelihara amanah-amanah dan janjinya, apabila mereka diberi amanah tidak mengkhianatinya dan bila berjanji tidak pernah melanggarnya. Inilah sifat orang-orang beriman, sedangkan yang sebaliknya adalah sifat-sifat orang munafik. Orang yang beriman apabila mereka diberi amanah, mereka tidak khianat dan apabila berjanji tidak ingkar. Orang-orang yang menjaga amanah yang mereka emban baik itu amanah Allah SWT maupun dari makhluk-Nya serta menepati janji-janji tanpa membatalkannya, apalagi melanggarnya (Abidin & Khairudin, 2017).

BAB III

METODE PENELITIAN

3.1 Jenis Penelitian

Metode penelitian yang digunakan dalam penelitian ini yaitu studi kepustakaan (*library research*). Studi kepustakaan adalah penelitian yang dilakukan dengan cara mengumpulkan dan mengkaji beberapa buku, *leaflet*, majalah, dan referensi teori lainnya yang relevan dengan masalah dan tujuan penelitian (Danial & Warsiah, 2009). Penelitian ini dilakukan dengan mengumpulkan dan mengkaji berbagai literatur yang berupa jurnal-jurnal, tugas akhir, dan beberapa buku yang ada kaitannya dengan materi tentang penyandian Vigenere Cipher dan grup simetri.

3.2 Pra Penelitian

Langkah awal yang dilakukan pada tahap pra penelitian yaitu memilih masalah dan menentukan judul. Setelah masalah dan judul disetujui oleh pembimbing, kemudian peneliti melakukan studi pendahuluan untuk mendapatkan gambaran awal tentang penelitian memodifikasi Vigenere Cipher menggunakan grup simetri untuk mengamankan pesan teks. Selanjutnya mengumpulkan berbagai referensi teori yang relevan dengan penyandian modifikasi Vigenere Cipher dan grup simetri. Literatur dari berbagai referensi teori tersebut diperlukan sebagai bahan pembahasan dari hasil penelitian.

3.3 Tahapan Penelitian

Secara umum tahapan penelitian yang dilakukan dalam penelitian ini yaitu sebagai berikut:

1. Memaparkan teknik penyandian Vigenere Cipher sebelum dimodifikasi dan memaparkan modifikasi Vigenere Cipher menggunakan grup simetri S_n
2. Menyusun algoritma penyandian modifikasi Vigenere Cipher mengunakan grup simetri S_n dan penerapannya menggunakan grup simetri S_5
   a. Proses enkripsi menggunakan modifikasi Vigenere Cipher yaitu sebagai berikut:
      1) Setelah menentukan *plaintext* dapat dilakukan pembentukan kunci dengan algoritma protokol perjanjian kunci Stickel atas grup simetri.
      2) Membagi *plaintext* menjadi blok-blok
      3) Mengenkripsi *plaintext* dengan kunci grup simetri menggunakan teknik transposisi (permutasi) dan menghasilkan *ciphertext1*.
      4) Menentukan kunci Vigenere Cipher.
      5) Mengkonversi *ciphertext1* dan kunci Vigenere Cipher ke dalam bentuk angka berdasarkan ASCII *printable characters*.
      6) *Ciphertext1* dan kunci Vigenere Cipher disubtitusikan ke dalam persamaan enkripsi Vigenere Cipher yang telah dimodifikasi dengan memanfaatkan ASCII *printable characters*.
      7) Hasil perhitungan pada *step* 6 dikembalikan ke bentuk karakter berdasarkan ASCII *printable characters*, sehingga memperoleh *ciphertext* akhir.

b. Proses dekripsi menggunakan modifikasi Vigenere Cipher yaitu sebagai berikut:
    1) Mengkonversi *ciphertext* akhir dan kunci Vigenere Cipher ke dalam bentuk angka berdasarkan ASCII *printable characters*.
    2) *Ciphertext* akhir dan kunci Vigenere Cipher disubtitusikan ke dalam persamaan dekripsi Vigenere Cipher yang telah dimodifikasi dengan memanfaatkan ASCII *printable characters*.
    3) Hasil perhitungan pada *step* 2 akan dikembalikan ke dalam bentuk karakter berdasarkan ASCII *printable characters*, sehingga diperoleh *ciphertext1*.
    4) Menginvers kunci grup simetri
    5) Membagi *ciphertext1* menjadi blok-blok
    6) Melakukan dekripsi *ciphertext1* dengan invers kunci grup simetri menggunakan teknik transposisi (permutasi) dan menghasilkan *plaintext*.

3. Melakukan uji kriptanalisis pada hasil enkripsi Modifikasi Vigenere Cipher kemudian dibandingkan dengan Vigenere Cipher.

BAB IV

PEMBAHASAN

4.1 Modifikasi Algoritma Vigenere Cipher

4.1.1 Vigenere Cipher

Vigenere Cipher merupakan salah satu cipher subtitusi yang bekerja hampir mirip dengan Caesar Cipher, yang membedakannya adalah pergeseran yang dilakukan menggunakan Vigenere Cipher tidak selalu sama. Pergeseran karakter *plaintext* pada Algoritma Vigenere Cipher ditentukan berdasarkan huruf kata kunci. Kunci pada Vigenere Cipher digunakan berulang-ulang.

Algoritma dari proses enkripsi dan dekripsi menggunakan Vigenere Cipher sebelum dimodifikasi yaitu sebagai berikut:

1) Setelah menentukan *plaintext* dan kunci, selanjutnya mengkonversi karakter huruf alfabet pada *plaintext* dan kunci ke dalam bentuk angka.
2) Subtitusikan *plaintext* dan kunci ke dalam persamaan (2.2)

$$C_i = (P_i + K_r)\ mod\ 26$$

(Catatan: apabila kunci kurang dari *plaintext* maka kunci akan diulang hingga seluruh *plaintext* tersebut terpenuhi. Apabila kunci melebihi *plaintext*, maka kunci akan ditulis sesuai dengan panjang *plaintext* yang diperlukan)

3) Mengembalikan hasil perhitungan pada *step* 2 ke dalam bentuk huruf alfabet, sehingga diperoleh *ciphertext*.

Sedangkan tahapan dari proses dekripsi menggunakan Vigenere cipher yaitu sebagai berikut:

1) Mengkonversi karakter huruf alfabet pada *ciphertext* dan kunci ke dalam bentuk angka
2) Subtitusikan *ciphertext* dan kunci ke dalam persamaan (2.3)

$$P_i = (C_i - K_r)\ mod\ 26$$

3) Mengembalikan hasil perhitungan pada *step* 2 ke dalam bentuk huruf alfabet, sehingga diperoleh *plaintext* kembali.

Berikut ini adalah contoh penerapan algoritma Vigenere Cipher untuk mengamankan pesan teks:

A. Proses Enkripsi Vigenere Cipher

Misalkan *plaintext* adalah `AS THE FORECAST BECAME MORE CERTAIN ACROSS THE WEEKEND THE GRAPICHS WERE USED EXTENSIVELY AS THE EXTENT OF THE SNOW FALL` dan kuncinya adalah `EASY`. Langkah awal yang dilakukan yaitu mengkonversi karakter huruf alfabet pada *plaintext* dan kunci ke dalam bentuk angka dengan melihat Tabel Konversi Alfabet (Lampiran 1). Sehingga,

Tabel 4.1 Enkripsi Vigenere Cipher

Plaintext (Teks Pesan)

P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}	P_{14}	P_{15}	P_{16}	P_{17}
A	S	T	H	E	F	O	R	E	C	A	S	T	B	E	C	A
0	18	19	7	4	5	14	17	4	2	0	18	19	1	4	2	0

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E
4	0	18	24	4	0	18	24	4	0	18	24	4	0	18	24	4

Langkah kedua, subtitusikan *plaintext* dan kunci ke dalam persamaan (2.2)

$$C_i = (P_i + K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

4	18	11	5	8	5	6	15	8	2	18	16	23	1	22	0	4
E	S	L	F	I	F	G	P	I	C	S	Q	X	B	W	A	E
C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	C_{16}	C_{17}

Lanjutan **Tabel 4.1** Enkripsi Vigenere Cipher

Plaintext (Teks Pesan)

P_{18}	P_{19}	P_{20}	P_{21}	P_{22}	P_{23}	P_{24}	P_{25}	P_{26}	P_{27}	P_{28}	P_{29}	P_{30}	P_{31}	P_{32}	P_{33}	P_{34}
M	E	M	O	R	E	C	E	R	T	A	I	N	A	C	R	O
12	4	12	14	17	4	2	4	17	19	0	8	13	0	2	17	14

Kunci Vigenere Cipher

K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2
A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A
0	18	24	4	0	18	24	4	0	18	24	4	0	18	24	4	0

Langkah kedua, subtitusikan *plaintext* dan kunci ke dalam persamaan (2.2)

$$C_i = (P_i + K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

12	22	10	18	17	22	0	8	17	11	24	12	13	18	0	21	14
M	W	K	S	R	W	A	I	R	L	Y	M	N	S	A	V	O
C_{18}	C_{19}	C_{20}	C_{21}	C_{22}	C_{23}	C_{24}	C_{25}	C_{26}	C_{27}	C_{28}	C_{29}	C_{30}	C_{31}	C_{32}	C_{33}	C_{34}

Lanjutan **Tabel 4.1** Enkripsi Vigenere Cipher

Plaintext (Teks Pesan)

P_{35}	P_{36}	P_{37}	P_{38}	P_{39}	P_{40}	P_{41}	P_{42}	P_{43}	P_{44}	P_{45}	P_{46}	P_{47}	P_{48}	P_{49}	P_{50}	P_{51}
S	S	T	H	E	W	E	E	K	E	N	D	T	H	E	G	R
18	18	19	7	4	22	4	4	10	4	13	3	19	7	4	6	17

Kunci Vigenere Cipher

K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3
S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S
18	24	4	0	18	24	4	0	18	24	4	0	18	24	4	0	18

Langkah kedua, subtitusikan *plaintext* dan kunci ke dalam persamaan (2.2)

$$C_i = (P_i + K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

10	16	23	7	22	20	8	4	2	2	17	3	11	5	8	6	9
K	Q	X	H	W	U	I	E	C	C	R	D	L	F	I	G	J
C_{35}	C_{36}	C_{37}	C_{38}	C_{39}	C_{40}	C_{41}	C_{42}	C_{43}	C_{44}	C_{45}	C_{46}	C_{47}	C_{48}	C_{49}	C_{50}	C_{51}

Lanjutan **Tabel 4.1** Enkripsi Vigenere Cipher

Plaintext (Teks Pesan)

P_{52}	P_{53}	P_{54}	P_{55}	P_{56}	P_{57}	P_{58}	P_{59}	P_{60}	P_{61}	P_{62}	P_{63}	P_{64}	P_{65}	P_{66}	P_{67}	P_{68}
A	P	H	I	C	S	W	E	R	E	U	S	E	D	E	X	T
0	15	7	8	2	18	22	4	17	4	20	18	4	3	4	23	19

Kunci Vigenere Cipher

K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4
Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y
24	4	0	18	24	4	0	18	24	4	0	18	24	4	0	18	24

Langkah kedua, subtitusikan *plaintext* dan kunci ke dalam persamaan (2.2)

$$C_i = (P_i + K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

24	19	7	0	0	22	22	22	15	8	20	10	2	7	4	15	17
Y	T	H	A	A	W	W	W	P	I	U	K	C	H	E	P	R
C_{52}	C_{53}	C_{54}	C_{55}	C_{56}	C_{57}	C_{58}	C_{59}	C_{60}	C_{61}	C_{62}	C_{63}	C_{64}	C_{65}	C_{66}	C_{67}	C_{68}

Lanjutan **Tabel 4.1** Enkripsi Vigenere Cipher

Plaintext (Teks Pesan)

P_{69}	P_{70}	P_{71}	P_{72}	P_{73}	P_{74}	P_{75}	P_{76}	P_{77}	P_{78}	P_{79}	P_{80}	P_{81}	P_{82}	P_{83}	P_{84}
E	N	S	I	V	E	L	Y	A	S	T	H	E	E	X	T
4	13	18	8	21	4	11	24	0	18	19	7	4	4	23	19

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y
4	0	18	24	4	0	18	24	4	0	18	24	4	0	18	24

Langkah kedua, subtitusikan *plaintext* dan kunci ke dalam persamaan (2.2)

$$C_i = (P_i + K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

8	13	10	6	25	4	3	22	4	18	11	5	8	4	15	17
I	N	K	G	Z	E	D	W	E	S	L	F	I	E	P	R
C_{69}	C_{70}	C_{71}	C_{72}	C_{73}	C_{74}	C_{75}	C_{76}	C_{77}	C_{78}	C_{79}	C_{80}	C_{81}	C_{82}	C_{83}	C_{84}

Lanjutan **Tabel 4.1** Enkripsi Vigenere Cipher

Plaintext (Teks Pesan)

P_{85}	P_{86}	P_{87}	P_{88}	P_{89}	P_{90}	P_{91}	P_{92}	P_{93}	P_{94}	P_{95}	P_{96}	P_{97}	P_{98}	P_{99}	P_{100}
E	N	T	O	F	T	H	E	S	N	O	W	F	A	L	L
4	13	19	14	5	19	7	4	18	13	14	22	5	0	11	11

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y
4	0	18	24	4	0	18	24	4	0	18	24	4	0	18	24

Langkah kedua, subtitusikan *plaintext* dan kunci ke dalam persamaan (2.2)

$$C_i = (P_i + K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

8	13	11	12	9	19	25	2	22	13	6	20	9	0	3	9
I	N	I	M	J	T	Z	C	W	N	G	U	J	A	D	J
C_{85}	C_{86}	C_{87}	C_{88}	C_{89}	C_{90}	C_{91}	C_{92}	C_{93}	C_{94}	C_{95}	C_{96}	C_{97}	C_{98}	C_{99}	C_{100}

Maka *ciphertext* yang diperoleh adalah

ESLFIFGPICSQXBWAEMWKSRWAIRLYMNSAVOKQXHWUIECCRDLFIGJYTHAA

WWWPIUKCHEPRINKGZEDWESLFIEPRINIMJTZCWNGUJADJ

B. Proses Dekripsi Vigenere Cipher

Proses dekripsi algoritma Vigenere Cipher atau proses mengembalikan *ciphertext* menjadi *plaintext* kembali dapat dilakukan seperti contoh berikut ini:

Langkah pertama yaitu mengkonversi karakter huruf pada *ciphertext* dan kunci ke dalam bentuk angka dengan melihat Tabel Konversi Alfabet (Lampiran 1). Sehingga,

Tabel 4.2 Dekripsi Vigenere Cipher

Ciphertext (Teks Sandi)

C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	C_{16}	C_{17}
E	S	L	F	I	F	G	P	I	C	S	Q	X	B	W	A	E
4	18	11	5	8	5	6	15	8	2	18	16	23	1	22	0	4

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E
4	0	18	24	4	0	18	24	4	0	18	24	4	0	18	24	4

Langkah kedua, subtitusikan *ciphertext* dan kunci ke dalam persamaan (2.3)

$$P_i = (C_i - K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

0	18	19	7	4	5	14	17	4	2	0	18	19	1	4	2	0
A	S	T	H	E	F	O	R	E	C	A	S	T	B	E	C	A
P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}	P_{14}	P_{15}	P_{16}	P_{17}

Lanjutan **Tabel 4.2** Dekripsi Vigenere Cipher

Ciphertext (Teks Sandi)

C_{18}	C_{19}	C_{20}	C_{21}	C_{22}	C_{23}	C_{24}	C_{25}	C_{26}	C_{27}	C_{28}	C_{29}	C_{30}	C_{31}	C_{32}	C_{33}	C_{34}
M	W	K	S	R	W	A	I	R	L	Y	M	N	S	A	V	O
12	22	10	18	17	22	0	8	17	11	24	12	13	18	0	21	14

Kunci Vigenere Cipher

K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2
A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A
0	18	24	4	0	18	24	4	0	18	24	4	0	18	24	4	0

Langkah kedua, subtitusikan *ciphertext* dan kunci ke dalam persamaan (2.3)

$$P_i = (C_i - K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

12	4	12	14	17	4	2	4	17	19	0	8	13	0	2	17	14
M	E	M	O	R	E	C	E	R	T	A	I	N	A	C	R	O
P_{18}	P_{19}	P_{20}	P_{21}	P_{22}	P_{23}	P_{24}	P_{25}	P_{26}	P_{27}	P_{28}	P_{29}	P_{30}	P_{31}	P_{32}	P_{33}	P_{34}

Lanjutan **Tabel 4.2** Dekripsi Vigenere Cipher

Ciphertext (Teks Sandi)

C_{35}	C_{36}	C_{37}	C_{38}	C_{39}	C_{40}	C_{41}	C_{42}	C_{43}	C_{44}	C_{45}	C_{46}	C_{47}	C_{48}	C_{49}	C_{50}	C_{51}
K	Q	X	H	W	U	I	E	C	C	R	D	L	F	I	G	J
10	16	23	7	22	20	8	4	2	2	17	3	11	5	8	6	9

Kunci Vigenere Cipher

K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3
S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S
18	24	4	0	18	24	4	0	18	24	4	0	18	24	4	0	18

Langkah kedua, subtitusikan *ciphertext* dan kunci ke dalam persamaan (2.3)

$$P_i = (C_i - K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

18	18	19	7	4	22	4	4	10	4	13	3	19	7	4	6	17
S	S	T	H	E	W	E	E	K	E	N	D	T	H	E	G	R
P_{35}	P_{36}	P_{37}	P_{38}	P_{39}	P_{40}	P_{41}	P_{42}	P_{43}	P_{44}	P_{45}	P_{46}	P_{47}	P_{48}	P_{49}	P_{50}	P_{51}

Lanjutan **Tabel 4.2** Dekripsi Vigenere Cipher

Ciphertext (Teks Sandi)

C_{52}	C_{53}	C_{54}	C_{55}	C_{56}	C_{57}	C_{58}	C_{59}	C_{60}	C_{61}	C_{62}	C_{63}	C_{64}	C_{65}	C_{66}	C_{67}	C_{68}
Y	T	H	A	A	W	W	W	P	I	U	K	C	H	E	P	R
24	19	7	0	0	22	22	22	15	8	20	10	2	7	4	15	17

Kunci Vigenere Cipher

K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4
Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y
24	4	0	18	24	4	0	18	24	4	0	18	24	4	0	18	24

Langkah kedua, subtitusikan *ciphertext* dan kunci ke dalam persamaan (2.3)

$$P_i = (C_i - K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

0	15	7	8	2	18	22	4	17	4	20	18	4	3	4	23	19
A	P	H	I	C	S	W	E	R	E	U	S	E	D	E	X	T
P_{52}	P_{53}	P_{54}	P_{55}	P_{56}	P_{57}	P_{58}	P_{59}	P_{60}	P_{61}	P_{62}	P_{63}	P_{64}	P_{65}	P_{66}	P_{67}	P_{68}

Lanjutan **Tabel 4.2** Dekripsi Vigenere Cipher

Ciphertext (Teks Sandi)

C_{69}	C_{70}	C_{71}	C_{72}	C_{73}	C_{74}	C_{75}	C_{76}	C_{77}	C_{78}	C_{79}	C_{80}	C_{81}	C_{82}	C_{83}	C_{84}
I	N	K	G	Z	E	D	W	E	S	L	F	I	E	P	R
8	13	10	6	25	4	3	22	4	18	11	5	8	4	15	17

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y
4	0	18	24	4	0	18	24	4	0	18	24	4	0	18	24

Langkah kedua, subtitusikan *ciphertext* dan kunci ke dalam persamaan (2.3)

$$P_i = (C_i - K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

4	13	18	8	21	4	11	24	0	18	19	7	4	4	23	19
E	N	S	I	V	E	L	Y	A	S	T	H	E	E	X	T
P_{69}	P_{70}	P_{71}	P_{72}	P_{73}	P_{74}	P_{75}	P_{76}	P_{77}	P_{78}	P_{79}	P_{80}	P_{81}	P_{82}	P_{83}	P_{84}

Lanjutan **Tabel 4.2** Dekripsi Vigenere Cipher

Ciphertext (Teks Sandi)

C_{85}	C_{86}	C_{87}	C_{88}	C_{89}	C_{90}	C_{91}	C_{92}	C_{93}	C_{94}	C_{95}	C_{96}	C_{97}	C_{98}	C_{99}	C_{100}
I	N	I	M	J	T	Z	C	W	N	G	U	J	A	D	J
8	13	11	12	9	19	25	2	22	13	6	20	9	0	3	9

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y
4	0	18	24	4	0	18	24	4	0	18	24	4	0	18	24

Langkah kedua, subtitusikan *ciphertext* dan kunci ke dalam persamaan (2.3)

$$P_i = (C_i - K_r)\ mod\ 26$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

4	13	19	14	5	19	7	4	18	13	14	22	5	0	11	11
E	N	T	O	F	T	H	E	S	N	O	W	F	A	L	L
P_{85}	P_{86}	P_{87}	P_{88}	P_{89}	P_{90}	P_{91}	P_{92}	P_{93}	P_{94}	P_{95}	P_{96}	P_{97}	P_{98}	P_{99}	P_{100}

Maka hasil dekripsi Vigenere Cipher mengembalikan *ciphertext* ke *plaintext* yaitu

AS THE FORECAST BECAME MORE CERTAIN ACROSS THE WEEKEND THE GRAPICHS WERE USED EXTENSIVELY AS THE EXTENT OF THE SNOW FALL

4.1.2 Kelemahan Vigenere Cipher

Vigenere Cipher dulunya terkenal dengan metode yang tangguh dan tidak mudah dipecahkan. Namun seiring berkembangnya teknologi, Vigenere Cipher menjadi tidak aman karena terdapat kelemahan Vigenere Cipher yaitu kuncinya yang pendek dan digunakan secara berulang-ulang. Kelemahan ini dapat

dimanfaatkan untuk memecahkan Vigenere Cipher. Adapun metode yang digunakan yaitu metode kasiski untuk menentukan panjang kunci yang digunakan. Selanjutnya kriptanalisis dapat dilakukan menggunakan metode analisis frekuensi atau secara *exhaustive key search*.

Perhatikan contoh proses penyandian Vigenere Cipher di atas, untuk pesan AS THE FORECAST BECAME MORE CERTAIN ACROSS THE WEEKEND THE GRAPICHS WERE USED EXTENSIVELY AS THE EXTENT OF THE SNOW FALL dengan kunci EASY diperoleh *Ciphertext* yaitu:

ES**LFI**FGPICSQXBWAEMWKSRWAIRLYMNSAVOKQXHWUIECCRD**LFI**GJYTHA AWWWPIUKCH**EPRIN**KGZEDWES**LFIEPRIN**IMJTZCWNGUJADJ

Terdapat kriptogram yang berulang yaitu **LFI** dan **EPRIN**. Jarak antara dua buah **LFI** yang berurutan adalah sebagai berikut:

Jarak **LFI** ke-1 dengan **LFI** ke-2 = 44

Jarak **LFI** ke-2 dengan **LFI** ke-3 = 32

Jarak antara dua buah perulangan **EPRIN** adalah 16

Faktor pembagi 44 adalah { 2, 4, 11, 44 }, faktor pembagi 32 adalah {2, 4, 8, 16, 32}, dan faktor pembagi 16 adalah { 2, 4, 8,16 }, irisan dari ketiga himpunan ini adalah 2 dan 4. Jadi, dapat diperkirakan panjang kunci yang digunakan adalah 2 atau 4. Kita asumsikan terlebih dahulu panjang kunci adalah 4 huruf (tanpa mengabaikan kemungkinan panjang kuncinya 2). Kemudian untuk menentukan kemungkinan karakter kunci digunakan analisis frekuensi. Langkah awal yang dilakukan yaitu mengelompokan *ciphertext* menjadi empat bagian. Kelompok 1 terdiri dari karakter-karakter hasil enkripsi dengan karakter kunci ke-1, kelompok 2 terdiri dari karakter-karakter hasil enkripsi dengan karakter kunci ke-2, kelompok

3 terdiri dari karakter-karakter hasil enkripsi dengan karakter kunci ke-3, kelompok 4 terdiri dari karakter-karakter hasil enkripsi dengan karakter kunci ke-4.

ES**LF** **I**FGP ICSQ XBWA EMWK SRWA IRLY MNSA VOKQ
1234 1234 1234 1234 1234 1234 1234 1234 1234
XHWU IECC RD**LF** **I**GJY THAA WWWP IUKC H**EPR** **IN**KG
1234 1234 1234 1234 1234 1234 1234 1234 1234
ZEDW ES**LF** **IEPR** **IN**IM JTZC WNGU JADJ
1234 1234 1234 1234 1234 1234 1234

Tabel 4.3 Pengelompokkan *Ciphertext*

Kelompok	*Ciphertext*	Huruf Paling Sering Muncul
1	EIIXESIMVXIRITWIHIZEIIJWJ	I
2	SFCBMRRNOHEDGHWUENESENTNA	E
3	LGSWWWLSKWCLJAWKPKDLPIZGD	W, L
4	FPQAKAYAQUCFYAPCRGWFRMCUJ	A

Diketahui bahwa 10 huruf yang paling sering muncul dalam Bahasa Inggris adalah E, T, A, O, I, N, S, H, R, D, L, dan trigram yang paling sering muncul adalah THE. Karena kriptogram LFI paling sering muncul di dalam *ciphertext*, maka dari 10 huruf tersebut kemungkinan kata 3-huruf dibentuk dan kita dapat menerka LFI adalah THE. Berdasarkan hal tersebut dapat dibuat tabel yang memetakan karakter *plaintext* dengan *ciphertext*, kemudian menentukan karakter kuncinya.

Tabel 4.4 Penerkaan Karakter Kunci

Kelompok	Huruf *Plaintext*	Huruf *Ciphertext*	Kunci
1	E	I	E
2	*	*	*
3	T	L	S
4	H	F	Y

Selanjutnya dilakukan penerkaan untuk karakter kunci lainnya yang belum diketahui (*). **Tabel 4.3** memperlihatkan bahwa pada kelompok 2 huruf paling sering muncul adalah huruf E. Sedangkan pada **Tabel 2.2** memperlihatkan bahwa huruf yang memiliki frekuensi kemunculan paling tinggi adalah huruf E. Asumsikan bahwa setiap karakter huruf E pada *plaintext* dienkripsi menjadi huruf E. Sehingga kemungkinan besar kunci yang belum diketahui adalah huruf A. Jadi, kuncinya mungkin EASY. Lakukan dekripsi menggunakan kunci tersebut untuk memastikan apakah kunci EASY merupakan kunci yang benar, dan periksa apakah hasil dekripsi merupakan pesan bermakna. Setelah dilakukan dekripsi diperoleh pesan sebagai berikut: AS THE FORECAST BECAME MORE CERTAIN ACROSS THE WEEKEND THE GRAPICHS WERE USED EXTENSIVELY AS THE EXTENT OF THE SNOW FALL

Ternyata benar bahwa EASY adalah kunci yang digunakan.

4.1.3 Modifikasi Vigenere Cipher Menggunakan Grup Simetri

Telah dibahas sebelumnya mengenai Vigenere Cipher, diketahui bahwa Vigenere memiliki kelemahan sehingga Vigenere Cipher dapat dipecahkan. Sehingga perlu dilakukan modifikasi terhadap Vigenere Cipher, supaya pesan rahasia dapat tersembunyi dengan aman. Modifikasi Vigenere Cipher yang dilakukan yaitu dengan tidak membuang spasi pada *plaintext* dan mengacak *plaintext* terlebih dahulu menggunakan penyandian grup simetri S_n sebelum dioperasikan dengan kunci Vigenere Cipher. Pengacakan *plaintext* terlebih dahulu menggunakan penyandian grup simetri bertujuan untuk menyembunyikan *plaintext* yang asli, walaupun seseorang mencoba membobol *ciphertext* menggunakan

metode Kasiski dan *exhaustive key search* tidak akan menemukan *plaintext* yang sebenarnya. Selain itu, digunakan ASCII *printable characters* yang bertujuan agar algoritma Vigenere Cipher mendukung penggunaan huruf kapital, huruf kecil, angka dan simbol. Semakin banyak karakter yang digunakan maka *ciphertext* yang dihasilkan juga semakin teracak, sehingga mempersulit seorang kriptanalis untuk membobol *ciphertext*.

Secara umum, teknik penyandian modifikasi Vigenere Cipher dimulai dengan mengacak *plaintext* menggunakan penyandian grup simetri. Setelah *plaintext* teracak, langkah selanjutnya menentukan kunci Vigenere Cipher. Proses enkripsi dilakukan dengan persamaan enkripsi Vigenere Cipher yang memanfaatkan ASCII *printable characters* dan menghasilkan *ciphertext*. Sedangkan proses dekripsinya dilakukan dengan persamaan dekripsi Vigenere Cipher yang memanfaatkan ASCII *printable characters.* Namun, untuk mendapatkan *plaintext* yang sebenarnya harus dilakukan dekripsi lagi menggunakan teknik penyandian grup simetri.

Teknik penyandian grup simetri yaitu penyandian pesan yang perhitungannya menggunakan grup simetri dalam pembentukan kunci, sedangkan untuk proses enkripsi dan dekripsinya digunakan teknik transposisi (permutasi). Teknik transposisi yaitu dengan melakukan permutasi k terhadap *plaintext*, sedangkan untuk dekripsinya dilakukan dengan invers permutasi k^{-1} terhadap *ciphertext*. Maka akan dibuktikan bahwa *plaintext* yang dienkripsi menggunakan teknik transposisi (permutasi) akan menghasilkan *ciphertext* yang apabila didekripsi akan kembali ke *plaintext*.

Misalkan permutasi k: $\Omega \to \Omega$ dituliskan sebagai berikut

$$\begin{pmatrix} 1 & 2 & 3 & \dots & n \\ a(1) & a(2) & a(3) & \dots & a(n) \end{pmatrix}$$

Baris pertama pada permutasi k berisi semua anggota domain Ω sementara baris kedua berisi $a(i)$ anggota kodomain Ω, untuk semua $i \in \Omega$. Permutasi k memetakan $1 \mapsto a(1), 2 \mapsto a(2), 3 \mapsto a(3), \dots, n \mapsto a(n)$.

Berdasarkan **Definisi 2.6** Permutasi dari suatu himpunan Ω adalah suatu fungsi dari Ω ke Ω yang satu-satu dan pada (bijektif). Sehingga setiap anggota himpunan Ω selalu memiliki pasangan tepat satu di himpunan Ω. Misalkan masing-masing karakter pada *plaintext* merupakan anggota dari domain Ω, maka karakter *ciphertext* merupakan anggota kodomain himpunan Ω. Dapat disimpulkan bahwa *ciphertext* yang dihasilkan dari proses enkripsi merupakan perubahan posisi dari *plaintext* berdasarkan pemetaan dari Ω ke Ω.

Fungsi invers adalah sebuah fungsi yang berkebalikan dari fungsi aslinya. Berdasarkan **Definisi 2.5** jika suatu fungsi bijektif maka fungsi tersebut memiliki invers. Karena permutasi merupakan fungsi bijektif maka permutasi memiliki invers. Perhatikan kembali permutasi k, permutasi yang memetakan $1 \mapsfrom a(1), 2 \mapsfrom a(2), 3 \mapsfrom a(3), \dots, n \mapsfrom a(n)$ disebut invers permutasi k dan dapat dinotasikan dengan permutasi k^{-1}. Dengan kata lain, enkripsi dilakukan dengan permutasi k terhadap *plaintext*, sedangkan dekripsi dilakukan dengan invers permutasi k^{-1} pada *ciphertext*.

Perhatikan bahwa,

Permutasi $k =$ kunci (K)

Invers permutasi $k^{-1} =$ invers kunci (K^{-1})

Misalkan kunci K dari himpunan $\{P_1, P_2, P_3, P_4\}$ dengan menetapkan

$K(P_1) = P_2, K(P_2) = P_4, K(P_3) = P_1, K(P_4) = P_3$ atau dapat dituliskan dalam bentuk sebagai berikut:

$$K = \begin{pmatrix} P_1 & P_2 & P_3 & P_4 \\ K(P_1) & K(P_2) & K(P_3) & K(P_4) \end{pmatrix} = \begin{pmatrix} P_1 & P_2 & P_3 & P_4 \\ P_2 & P_4 & P_1 & P_3 \end{pmatrix}$$

dan kunci invers K^{-1} dari himpunan $\{C_1, C_2, C_3, C_4\}$, dimana $\{C_1, C_2, C_3, C_4\} = \{P_1, P_2, P_3, P_4\}$, kedua himpunan tersebut sama tetapi memiliki urutan elemen yang berbeda. Sehingga,

$K^{-1}(C_1) = C_3,\ K^{-1}(C_2) = C_1,\ K^{-1}(C_3) = C_4,\ K^{-1}(C_4) = C_2$, atau bisa ditulis menjadi

$$K^{-1} = \begin{pmatrix} C_1 & C_2 & C_3 & C_4 \\ C_3 & C_1 & C_4 & C_2 \end{pmatrix}$$

keterangan $P_i =$ karakter *plaintext* ke – i, , untuk $i = 1,2,3,4$

$C_i =$ karakter *ciphertext* ke – i, untuk $i = 1,2,3,4$

Misalkan diberikan *plaintext* = COBA, maka

$$K = \begin{pmatrix} P_1 & P_2 & P_3 & P_4 \\ P_2 & P_4 & P_1 & P_3 \end{pmatrix} = \begin{pmatrix} P_1 & P_2 & P_3 & P_4 \\ C & O & B & A \\ P_2 & P_4 & P_1 & P_3 \\ O & A & C & B \\ \downarrow & \downarrow & \downarrow & \downarrow \\ C_1 & C_2 & C_3 & C_4 \end{pmatrix}$$

Sehingga diperoleh *ciphertext* = OACB

$$K^{-1} = \begin{pmatrix} C_1 & C_2 & C_3 & C_4 \\ C_3 & C_1 & C_4 & C_2 \end{pmatrix} = \begin{pmatrix} C_1 & C_2 & C_3 & C_4 \\ O & A & C & B \\ C_3 & C_1 & C_4 & C_2 \\ C & O & B & A \\ \downarrow & \downarrow & \downarrow & \downarrow \\ P_1 & P_2 & P_3 & P_4 \end{pmatrix}$$

Sehingga teks sandi (*ciphertext*) kembali ke bentuk *plaintext* = COBA

Jadi terbukti bahwa *plaintext* yang dienkripsi menggunakan teknik transposisi (permutasi) akan menghasilkan *ciphertext* yang apabila didekripsi akan kembali ke *plaintext*.

Berikut adalah persamaan untuk enkripsi pesan menggunakan algoritma Vigenere Cipher yang memanfaatkan *ASCII printable characters*:

$$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32 \quad (4.1)$$

Sedangkan untuk proses dekripsi dapat menggunakan persamaan berikut:

$$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32 \quad (4.2)$$

Keterangan:

C_i = *ciphertext* ke- i

P_i = *plaintext* ke- i

K_r = kunci ke – r

32 = karakter spasi pada tabel ASCII.

Karakter dalam ASCII *printable characters* yaitu dari karakter dengan angka desimal 32 sampai 126, yang seluruhnya berjumlah 95 karakter. Sehingga digunakan modulus 95. Sedangkan penambahan 32 dilakukan supaya nilai C_i kembali berada di interval 32-126.

Pembuktian untuk persamaan (4.1) dan (4.2) dapat dilakukan dengan mengembalikan rumus enkripsi ke rumus dekripsi dan sebaliknya, sehingga:

$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$

$(C_i - 32) = (P_i - 32 + K_r)\ mod\ 95$ ... ... 32 dipindah ruas

$95|(C_i - 32) - (P_i - 32 + K_r)$ ... ... sesuai definisi kongruensi

$(C_i - 32) - (P_i - 32 + K_r) = 95.k$ ... ... berdasarkan definisi keterbagian

$C_i - 32 - P_i + 32 - K_r = 95.k$ ... ... sifat distributif

$-C_i + 32 + P_i - 32 + K_r = 95.(-k)$ ... ...dikalikan (-1)

$(P_i - 32) - (C_i - 32 - K_r) = 95.(-k)$ ... ... Sifat distributif

$95|(P_i - 32) - (C_i - 32 - K_r)$ ... ... Berdasarkan definisi keterbagian

$(P_i - 32) = (C_i - 32 - K_r)\ mod\ 95$ …. … …. berdasarkan definisi kongruensi

$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$ … … … … 32 dipindah ruas

Sedangkan pembuktian rumus dekripsi, yaitu mengembalikan persamaan (4.2) ke persamaan (4.1) adalah sebagai berikut:

$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$

$(P_i - 32) = (C_i - 32 - K_r)\ mod\ 95$ … … … … …. 32 dipindah ruas

$95|(P_i - 32) - (C_i - 32 - K_r)$ … … … … … … ….. sesuai definisi kongruensi

$(P_i - 32) - (C_i - 32 - K_r) = 95.(-k)$ … … … .. sesuai definsisi keterbagian

$P_i - 32 - C_i + 32 + K_r = 95.(-k)$ … … … … …. Sifat distributif

$-P_i + 32 + C_i - 32 - K_r = 95.(k)$ … … … … … dikalikan (-1)

$(C_i - 32) - (P_i - 32 + K_r) = 95.(k)$ … … … … sifat distributif

$95|(C_i - 32) - (P_i - 32 + K_r)$ … … … … … … .. sesuai definisi keterbagian

$(C_i - 32) = (P_i - 32 + K_r)\ mod\ 95$ … … … … .. sesuai definisi kongruensi

$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$

Jadi terbukti bahwa persamaan (4.1) dan (4.2) dapat digunakan untuk proses enkripsi dan dekripsi Vigenere Cipher berdasarkan ASCII *printable characters*.

Berikut ini contoh proses enkripsi dan dekripsi Vigenere Cipher menggunakan persamaan (4.1) dan (4.2):

Misalkan nilai $P = 56$ dan $K = 32$, kemudian disubtitusikan ke dalam persamaan (4.1) sehingga:

$C = \big((P - 32 + K)\ mod\ 95\big) + 32$

$= \big((56 - 32 + 32)\ mod\ 95\big) + 32$

$= 88$

Diperoleh $C = 88$, dengan kunci yang sama yaitu $K = 32$, kemudian didekripsi menggunakan persamaan (4.2) :

$$P = \big((C - 32 - K)\ mod\ 95\big) + 32$$
$$= \big((88 - 32 - 32)\ mod\ 95\big) + 32$$
$$= 56$$

Diperoleh nilai $P = 56$, jadi terbukti bahwa *plaintext* yang dienkripsi menggunakan persamaan (4.1) akan menghasilkan *ciphertext* yang apabila didekripsi menggunakan persamaan (4.2) akan kembali ke *plaintext*.

4.2 Teknik Penyandian Modifikasi Algoritma Vigenere Cipher Menggunakan Grup Simetri

4.2.1 Proses Enkripsi dan Dekripsi Modifikasi Vigenere Cipher Menggunakan Grup Simetri S_n

Grup simetri S_n merupakan salah satu grup non-komutatif yang dapat digunakan pada protokol perjajian kunci Stickel (n adalah banyaknya anggota himpunan yang dipermutasikan). Penerapan grup simetri S_n pada proses pembentukan kunci akan menghasilkan suatu kunci rahasia. Kunci tersebut nantinya digunakan pada proses enkripsi penyandian grup simetri, sedangkan pada proses dekripsinya digunakan invers kunci. Cara menentukan S_n yang digunakan untuk pembentukan kunci grup simetri yaitu pilih sebarang n, dengan $n \geq 3$ (karena S_n yang digunakan haruslah tidak komutatif) dan n kurang dari atau sama dengan banyaknya *plaintext*. Sebenarnya pemilihan besar atau kecil nya nilai n pada S_n yang digunakan tidak terlalu mempengaruhi tingkat teracaknya *plaintext*.

Jadi, disarankan pastikan kunci grup simetri yang dihasilkan merupakan permutasi yang acak supaya *plaintext* teracak dengan maksimal.

A. Proses enkripsi:

Setelah menentukan *plaintext* dapat dilakukan proses enkripsi menggunakan algoritma modifikasi Vigenere Cipher sebagai berikut ini:

1. Pembentukan kunci grup simetri.

   Berikut proses pembentukan kunci yang dimulai dengan pihak pengirim dan penerima pesan menyepakati dua elemen dari grup simetri S_n yaitu:

   pilih σ_1 dan σ_2 adalah elemen dari S_n, dengan $\sigma_1 \circ \sigma_2 \neq I$ (I adalah permutasi identitas) dan $\sigma_1 \circ \sigma_2 \neq \sigma_2 \circ \sigma_1$, sedemikian hingga $K \neq I$ (K adalah kunci grup simetri).

$$\sigma_1 = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ \sigma_1(a_1) & \sigma_1(a_2) & \dots & \sigma_1(a_n) \end{pmatrix}$$

$$\sigma_2 = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ \sigma_2(a_1) & \sigma_2(a_2) & \dots & \sigma_2(a_n) \end{pmatrix}$$

   Kemudian tentukan R yang merupakan orde dari σ_1 dan S adalah orde dari σ_2.

   (i) Pengirim pesan memilih bilangan asli $M < R$ dan $N < S$ dan mengirim

   $$r = {\sigma_1}^M \circ {\sigma_2}^N$$

   (ii) Penerima pesan memilih bilangan asli $P < R$, $Q < S$ dan mengirim

   $$s = {\sigma_1}^P \circ {\sigma_2}^Q$$

   (iii) Pengirim pesan menerima s dari pihak penerima

   (iv) Penerima pesan menerima r dari pihak pengirim

   (v) Pengirim pesan menghitung

   $$K_1 = {\sigma_1}^M \circ s \circ {\sigma_2}^N$$

   (vi) Penerima pesan menghitung

$K_2 = {\sigma_1}^P \circ r \circ {\sigma_2}^Q$

Sehingga dapat diperoleh kunci rahasia yang telah disepakati oleh kedua pihak yaitu

$K_1 = {\sigma_1}^M \circ s \circ {\sigma_2}^N$

$= {\sigma_1}^M \circ {\sigma_1}^P \circ {\sigma_2}^Q \circ {\sigma_2}^N$

$= {\sigma_1}^{M+P} \circ {\sigma_2}^{Q+N}$

$= {\sigma_1}^P \circ {\sigma_1}^M \circ {\sigma_2}^N \circ {\sigma_2}^Q$

$= {\sigma_1}^P r {\sigma_2}^Q$

$= K_2$

Jadi, $K = K_1 = K_2$

2. Membagi *plaintext* menjadi blok-blok. Setiap bloknya berisi karakter sebanyak n, dan apabila terdapat karakter spasi pada *plaintext* maka spasinya tidak dihilangkan karena merupakan salah satu karakter dalam Tabel ASCII *printable characters*. Sehingga karakter spasi dapat dituliskan dengan *sp*. Sedangkan untuk kekurangan pada blok dapat ditambahkan dengan karakter yang disukai seperti ∅.
3. Setiap blok *plaintext* akan dipermutasikan dengan kunci grup simetri seperti berikut ini:

Blok: $K = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ k(a_1) & k(a_2) & \dots & k(a_n) \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ P_1 & P_2 & \dots & P_n \\ k(a_1) & k(a_2) & \dots & k(a_n) \\ k(P_1) & k(P_2) & \dots & k(P_n) \\ \downarrow & \downarrow & \downarrow & \downarrow \\ C_1 & C_2 & \dots & C_n \end{pmatrix}$

Keterangan: $P_i =$ karakter *plaintext* ke- i

$C_i =$ karakter *ciphertext* ke- i

Setelah semua blok selesai dipermutasikan, maka karakter-karakter pada *plaintext* akan teracak.

4. Menentukan Kunci Vigenere Cipher

   (Catatan: apabila banyak karakter pada kunci kurang dari *plaintext* maka kunci akan diulang hingga seluruh *plaintext* tersebut terpenuhi. Apabila kunci melebihi *plaintext*, maka kunci akan ditulis sesuai dengan panjang *plaintext* yang diperlukan)

5. *Plaintext* yang telah dipermutasikan menjadi *ciphertext1* dan kunci Vigenere Cipher dikonversi ke dalam bentuk angka desimal dengan melihat Tabel ASCII *printable characters* (Lampiran 2).

6. *Ciphertext1* dan kunci Vigenere Cipher akan disubtitusikan ke dalam persamaan (4.1):

   $$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$$

   (Catatan: apabila terdapat karakter ∅ pada *ciphertext* maka tidak ikut dienkripsi namun tetap ditulis di posisinya)

7. Hasil perhitungan dari *step* 6 dikembalikan ke dalam bentuk karakter berdasarkan ASCII *printable characters*, sehingga diperoleh *ciphertext* akhir.

B. Proses dekripsi

Selanjutnya dari proses enkripsi yang telah dilakukan sebelumnya, *ciphertext* akhir yang diperoleh akan didekripsi menggunakan modifikasi algoritma Vigenere Cipher dengan grup simetri S_n, sehingga proses dekripsinya yaitu sebagai berikut:

1. Mengkonversi karakter-karakter pada *ciphertext* akhir dan kunci Vigenere Cipher ke dalam bentuk angka dengan melihat tabel ASCII *printable characters* (Lampiran 2).
2. *Ciphertext* akhir dan kunci Vigenere Cipher akan disubtitusikan ke dalam persamaan (4.2):

$$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$$

(Catatan: apabila terdapat karakter ∅ pada *ciphertext* maka tidak ikut didekripsi namun tetap ditulis di posisinya)

3. Hasil perhitungan pada *step* 2 akan dikembalikan ke dalam bentuk karakter berdasarkan ASCII *printable characters*, sehingga diperoleh *ciphertext1*.
4. Menginvers kunci grup simetri

$$K^{-1} = \begin{pmatrix} k(a_1) & k(a_2) & \dots & k(a_n) \\ a_1 & a_2 & \dots & a_n \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ k^{-1}(a_1) & k^{-1}(a_2) & \dots & k^{-1}(a_n) \end{pmatrix}$$

5. Membagi *ciphertext1* menjadi blok-blok
6. Setiap blok *ciphertext1* dipermutasikan dengan kunci invers grup simetri seperti berikut ini:

Blok: $K^{-1} = \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ k^{-1}(a_1) & k^{-1}(a_2) & \dots & k^{-1}(a_n) \end{pmatrix}$

$$= \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ C_1 & C_2 & \dots & C_n \\ k^{-1}(a_1) & k^{-1}(a_2) & \dots & k^{-1}(a_n) \\ k^{-1}(C_1) & k^{-1}(C_2) & \dots & k^{-1}(C_n) \\ \downarrow & \downarrow & \downarrow & \downarrow \\ P_1 & P_2 & \dots & P_n \end{pmatrix}$$

Keterangan: $P_i =$ karakter *plaintext* ke- i

$C_i =$ karakter *ciphertext* ke- i

Setelah semua blok selesai dipermutasikan, maka diperoleh *plaintext* yang sebenarnya.

4.2.2 Proses Enkripsi dan Dekripsi Modifikasi Vigenere Cipher Menggunakan Grup Simetri S_5

A. Proses enkripsi:

Misalkan diberikan *plaintext* yaitu `AS THE FORECAST BECAME MORE CERTAIN ACROSS THE WEEKEND THE GRAPICHS WERE USED EXTENSIVELY AS THE EXTENT OF THE SNOW FALL`, maka proses enkripsinya adalah sebagai berikut:

1. Membentuk kunci grup simetri

   Berikut proses pembentukan kunci yang dimulai dengan pihak pengirim dan penerima pesan menyepakati dua elemen dari grup simetri S_5 yaitu:

   pilih σ_1 dan σ_2 adalah elemen dari S_5, dengan $\sigma_1 \circ \sigma_2 \neq I$ (I adalah permutasi identitas) dan $\sigma_1 \circ \sigma_2 \neq \sigma_2 \circ \sigma_1$, sedemikian hingga $K \neq I$ (K adalah kunci grup simetri).

$$\sigma_1 = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_4 & a_5 & a_1 & a_2 \end{pmatrix} = (13524)$$

$$\sigma_2 = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix} = (1)(2534)$$

   Kemudian tentukan R yang merupakan orde dari σ_1 dan S adalah orde dari σ_2

   $R = |(13524)| =$ KPK dari 5 adalah 5

   $S = |(1)(2534)| =$ KPK dari 1 dan 4 adalah 4

   (i) Pengirim memilih dua bilangan asli $M < R = 4$, $N < S = 1$ dan mengirim

   $r = {\sigma_1}^M \circ {\sigma_2}^N$

$$= \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_4 & a_5 & a_1 & a_2 \end{pmatrix}^4 \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix}^1$$

$$= \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_5 & a_1 & a_2 & a_3 \end{pmatrix} \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix}$$

$$r = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_3 & a_2 & a_5 & a_1 \end{pmatrix}$$

(ii) Penerima juga melakukan hal yang sama, memilih bilangan asli

$P < R = 3$, $Q < S = 1$ dan mengirim

$$s = {\sigma_1}^P \circ {\sigma_2}^Q$$

$$s = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_4 & a_5 & a_1 & a_2 \end{pmatrix}^3 \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix}^1$$

$$s = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_2 & a_3 & a_4 & a_5 & a_1 \end{pmatrix} \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix}$$

$$s = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_2 & a_1 & a_5 & a_3 & a_4 \end{pmatrix}$$

(iii) Pengirim pesan menerima s dari penerima pesan

(iv) Penerima pesan menerima r dari pengirim pesan

(v) Pengirim pesan menghitung kunci

$$K_1 = {\sigma_1}^M \circ s \circ {\sigma_2}^N$$

$$K_1 = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_4 & a_5 & a_1 & a_2 \end{pmatrix}^4 \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_2 & a_1 & a_5 & a_3 & a_4 \end{pmatrix} \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix}^1$$

$$K_1 = \left(\begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_5 & a_1 & a_2 & a_3 \end{pmatrix} \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_2 & a_1 & a_5 & a_3 & a_4 \end{pmatrix} \right) \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix}$$

$$K_1 = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_4 & a_3 & a_1 & a_2 \end{pmatrix} \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix}$$

$$K_1 = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix}$$

(vi) Penerima pesan menghitung

$$K_2 = {\sigma_1}^P \circ r \circ {\sigma_2}^Q$$

$$K_2 = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_4 & a_5 & a_1 & a_2 \end{pmatrix}^3 \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_3 & a_2 & a_5 & a_1 \end{pmatrix} \circ$$

$$\begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix}^1$$

$$K_2 = \left(\begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_2 & a_3 & a_4 & a_5 & a_1 \end{pmatrix} \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_4 & a_3 & a_2 & a_5 & a_1 \end{pmatrix} \right) \circ$$

$$\begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix}$$

$$K_2 = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_4 & a_3 & a_1 & a_2 \end{pmatrix} \circ \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_5 & a_4 & a_2 & a_3 \end{pmatrix}$$

$$K_2 = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix}$$

Jadi kunci yang disepakati oleh kedua belah pihak yaitu

$$K = K_1 = \ K_2 = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix}$$

2. Selanjutnya membagi *plaintext* tersebut menjadi lima blok dengan ketentuan apabila terdapat spasi, maka spasinya tidak dihilangkan karena merupakan salah satu karakter dalam Tabel ASCII *printable characters*. Sehingga karakter spasi dapat dituliskan dengan sp. Sedangkan untuk kekurangan pada blok dapat ditambahkan dengan karakter yang disukai seperti $\emptyset$.

ASspTH	EFORE	CASTsp	BECAM
EspMOR	EspCER	TAINsp	ACROS
SspTHE	spWEEK	ENDspT	HEspGR
APHIC	SspWER	EspUSE	DspEXT

ENSIV	ELYspA	SspTHE	spEXTE
NTspOF	spTHEsp	SNOWsp	FALL$\emptyset$

3. Setelah itu setiap blok diubah menjadi seperti di bawah ini dengan menggunakan kunci grup simetri yang telah disepakati.

Blok 1: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ A & S & sp & T & H \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ H & S & A & T & sp \end{pmatrix}$

Blok 2: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & F & O & R & E \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ E & F & E & R & O \end{pmatrix}$

Blok 3: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ C & A & S & T & sp \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ sp & A & C & T & S \end{pmatrix}$

Blok 4: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ B & E & C & A & M \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ M & E & B & A & C \end{pmatrix}$

Blok 5: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & sp & M & O & R \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ R & sp & E & O & M \end{pmatrix}$

Blok 6: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & sp & C & E & R \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ R & sp & E & E & C \end{pmatrix}$

Blok 7: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ T & A & I & N & sp \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ sp & A & T & N & I \end{pmatrix}$

Blok 8: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ A & C & R & O & S \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ S & C & A & O & R \end{pmatrix}$

Blok 9: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ S & sp & T & H & E \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ E & sp & S & H & T \end{pmatrix}$

Blok 10: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ sp & W & E & E & K \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ K & W & sp & E & E \end{pmatrix}$

Blok 11: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & N & D & sp & T \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ T & N & E & sp & D \end{pmatrix}$

Blok 12: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ H & E & sp & G & R \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ R & E & H & G & sp \end{pmatrix}$

Blok 13: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ A & P & H & I & C \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ C & P & A & I & H \end{pmatrix}$

Blok 14: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ S & sp & W & E & R \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ R & sp & S & E & W \end{pmatrix}$

Blok 15: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & sp & U & S & E \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ E & sp & E & S & U \end{pmatrix}$

Blok 16: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ D & sp & E & X & T \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ T & sp & D & X & E \end{pmatrix}$

Blok 17: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & N & S & I & V \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ V & N & E & I & S \end{pmatrix}$

Blok 18: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & L & Y & sp & A \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ A & L & E & sp & Y \end{pmatrix}$

Blok 19: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ S & sp & T & H & E \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ E & sp & S & H & T \end{pmatrix}$

Blok 20: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ sp & E & X & T & E \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ E & E & sp & T & X \end{pmatrix}$

Blok 21: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ N & T & sp & O & F \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ F & T & N & O & sp \end{pmatrix}$

Blok 22: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ sp & T & H & E & sp \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ sp & T & sp & E & H \end{pmatrix}$

Blok 23: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ S & N & O & W & sp \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ sp & N & S & W & O \end{pmatrix}$

Blok 24: $K = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 & a_4 & a_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ F & A & L & L & \emptyset \\ a_5 & a_2 & a_1 & a_4 & a_3 \\ \emptyset & A & F & L & L \end{pmatrix}$

Sehingga diperoleh *ciphertext1* yaitu

HSAT*sp* EFERO *sp*ACTS MEBAC R*sp*EOM R*sp*EEC *sp*ATNI SCAOR E*sp*SHT KW*sp*EE TNE*sp*D REHG*sp* CPAIH R*sp*SEW E*sp*ESU T*sp*DXE VNEIS ALE*sp*Y E*sp*SHT EE*sp*TX FTNO*sp* *sp*T*sp*EH *sp*NSWO ∅AFLL

4. Setelah karakter-karakter pada *plaintext* teracak menjadi *ciphertext1,* selanjutnya menentukan kunci Vigenere Cipher. Kunci Vigenere Cipher yang digunakan yaitu EASY.
5. Mengkonversi karakter-karakter pada *ciphertext*1 dan kunci Vigenere Cipher ke dalam bentuk angka desimal dengan melihat tabel ASCII *Printable Characters* (Lampiran 2). Sehingga,

Tabel 4.5 Enkripsi Modifikasi Vigenere Cipher

Ciphertext1

P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}	P_{14}	P_{15}	P_{16}
H	S	A	T		E	F	E	R	O		A	C	T	S	M
72	83	65	84	32	69	70	69	82	79	32	65	67	84	83	77

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y
69	65	83	89	69	65	83	89	69	65	83	89	69	65	83	89

Subtitusikan *ciphertext1* dan kunci Vigenere Cipher ke dalam persamaan (4.1)

$$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

46	53	53	78	101	39	58	63	56	49	115	59	41	54	71	71
.	5	5	N	e	'	:	?	8	1	s	;	)	6	G	G
C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	C_{16}

Lanjutan **Tabel 4.5** Enkripsi Modifikasi Vigenere Cipher

Ciphertext1

P_{17}	P_{18}	P_{19}	P_{20}	P_{21}	P_{22}	P_{23}	P_{24}	P_{25}	P_{26}	P_{27}	P_{28}	P_{29}	P_{30}	P_{31}
E	B	A	C	R		E	O	M	R		E	E	C	
69	66	65	67	82	32	69	79	77	82	32	69	69	67	32

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S
69	65	83	89	69	65	83	89	69	65	83	89	69	65	83

Subtitusikan *ciphertext1* dan kunci Vigenere Cipher ke dalam persamaan (4.1)

$$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

43	36	53	61	56	97	57	73	51	52	115	63	43	37	115
+	$	5	=	8	a	9	I	3	4	s	?	+	%	s
C_{17}	C_{18}	C_{19}	C_{20}	C_{21}	C_{22}	C_{23}	C_{24}	C_{25}	C_{26}	C_{27}	C_{28}	C_{29}	C_{30}	C_{31}

Lanjutan **Tabel 4.5** Enkripsi Modifikasi Vigenere Cipher

Ciphertext1

P_{32}	P_{33}	P_{34}	P_{35}	P_{36}	P_{37}	P_{38}	P_{39}	P_{40}	P_{41}	P_{42}	P_{43}	P_{44}	P_{45}	P_{46}
A	T	B	I	S	C	A	O	R	E		S	H	T	K
65	84	66	73	83	67	65	79	82	69	32	83	72	84	75

Kunci Vigenere Cipher

K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2
Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A
89	69	65	83	89	69	65	83	89	69	65	83	89	69	65

Subtitusikan *ciphertext1* dan kunci Vigenere Cipher ke dalam persamaan (4.1)

$$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

59	58	36	61	77	20	35	67	76	43	97	71	66	58	45
;	:	$	=	M	)	#	C	L	+	a	G	B	:	-
C_{32}	C_{33}	C_{34}	C_{35}	C_{36}	C_{37}	C_{38}	C_{39}	C_{40}	C_{41}	C_{42}	C_{43}	C_{44}	C_{45}	C_{46}

Lanjutan **Tabel 4.5** Enkripsi Modifikasi Vigenere Cipher

Ciphertext1

P_{47}	P_{48}	P_{49}	P_{50}	P_{51}	P_{52}	P_{53}	P_{54}	P_{55}	P_{56}	P_{57}	P_{58}	P_{59}	P_{60}	P_{61}
W		E	E	T	N	E		D	R	E	H	G		C
87	32	69	69	84	78	69	32	68	82	69	72	71	32	67

Kunci Vigenere Cipher

K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1
S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E
83	89	69	65	83	89	69	65	83	89	69	65	83	89	69

Subtitusikan *ciphertext1* dan kunci Vigenere Cipher ke dalam persamaan (4.1)

$$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

75	121	43	39	72	72	43	97	56	76	43	42	59	121	41
K	y	+	′	H	H	+	A	8	L	+	*	;	y	)
C_{47}	C_{48}	C_{49}	C_{50}	C_{51}	C_{52}	C_{53}	C_{54}	C_{55}	C_{56}	C_{57}	C_{58}	C_{59}	C_{60}	C_{61}

Lanjutan **Tabel 4.5** Enkripsi Modifikasi Vigenere Cipher

Ciphertext1

P_{62}	P_{63}	P_{64}	P_{65}	P_{66}	P_{67}	P_{68}	P_{69}	P_{70}	P_{71}	P_{72}	P_{73}	P_{74}	P_{75}	P_{76}
P	A	I	H	R		S	E	W	E		E	S	U	T
80	65	73	72	82	32	83	69	87	69	32	69	83	85	84

Kunci Vigenere Cipher

K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4
A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y
65	83	89	69	65	83	89	69	65	83	89	69	65	83	89

Subtitusikan *ciphertext1* dan kunci Vigenere Cipher ke dalam persamaan (4.1)

$$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

50	53	67	46	52	115	77	43	57	57	121	43	53	73	78
2	5	C	.	4	s	M	+	9	9	y	+	5	I	N
C_{62}	C_{63}	C_{64}	C_{65}	C_{66}	C_{67}	C_{68}	C_{69}	C_{70}	C_{71}	C_{72}	C_{73}	C_{74}	C_{75}	C_{76}

Lanjutan **Tabel 4.5** Enkripsi Modifikasi Vigenere Cipher

Ciphertext1

P_{77}	P_{78}	P_{79}	P_{80}	P_{81}	P_{82}	P_{83}	P_{84}	P_{85}	P_{86}	P_{87}	P_{88}	P_{89}	P_{90}	P_{91}
	D	X	E	V	N	E	I	S	A	L	E		Y	E
32	68	88	69	118	78	69	73	83	65	76	69	32	121	69

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S
69	65	83	89	69	65	83	89	69	65	83	89	69	65	83

Subtitusikan *ciphertext1* dan kunci Vigenere Cipher ke dalam persamaan (4.1)

$$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

101	38	76	63	92	48	57	67	57	35	64	63	101	91	57
e	&	L	?	\	0	9	C	9	#	@	?	e	[	9
C_{77}	C_{78}	C_{79}	C_{80}	C_{81}	C_{82}	C_{83}	C_{84}	C_{85}	C_{86}	C_{87}	C_{88}	C_{89}	C_{90}	C_{91}

Lanjutan **Tabel 4.5** Enkripsi Modifikasi Vigenere Cipher

Ciphertext1

P_{92}	P_{93}	P_{94}	P_{95}	P_{96}	P_{97}	P_{98}	P_{99}	P_{100}	P_{101}	P_{102}	P_{103}	P_{104}	P_{105}	P_{106}
	S	H	T	E	E		T	X	F	T	N	O		
32	83	72	84	69	69	32	84	120	70	84	78	79	32	32

Kunci Vigenere Cipher

K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2
Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A
89	69	65	83	89	69	65	83	89	69	65	83	89	69	65

Subtitusikan *ciphertext1* dan kunci Vigenere Cipher ke dalam persamaan (4.1)

$$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

121	57	42	72	63	43	97	72	114	44	54	66	73	101	97
y	9	$*$	H	?	+	a	H	r	,	6	B	I	E	a
C_{92}	C_{93}	C_{94}	C_{95}	C_{96}	C_{97}	C_{98}	C_{99}	C_{100}	C_{101}	C_{102}	C_{103}	C_{104}	C_{105}	C_{106}

Lanjutan **Tabel 4.5** Enkripsi Modifikasi Vigenere Cipher

Ciphertext1

P_{107}	P_{108}	P_{109}	P_{110}	P_{111}	P_{112}	P_{113}	P_{114}	P_{115}	P_{116}	P_{117}	P_{118}	P_{119}	P_{120}
T		E	H		N	S	W	O	Ø	A	F	L	L
84	32	69	72	32	78	83	87	79		65	70	76	76

Kunci Vigenere Cipher

K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3		K_4	K_1	K_2	K_3
S	Y	E	A	S	Y	E	A	S		Y	E	A	S
83	89	69	65	83	89	69	65	83		89	69	65	83

Subtitusikan *ciphertext1* dan kunci Vigenere Cipher ke dalam persamaan (4.1)

$$C_i = \big((P_i - 32 + K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

72	121	43	42	115	72	57	57	67		59	44	46	64
H	y	+	*	s	H	9	9	C	Ø	;	,	·	@
C_{107}	C_{108}	C_{109}	C_{110}	C_{111}	C_{112}	C_{113}	C_{114}	C_{115}	C_{116}	C_{117}	C_{118}	C_{119}	C_{120}

Maka *ciphertext* akhir yang diperoleh adalah

```
.55Ne':?81s;)6GG+$5=8a9I34s?+%s;:$=M)#CL+aGB:-
Ky+'HH+a8L+*;y)25C.4sM+99y+5INe&L?\09C9#@?e[9y9*H?+aH
r,6BIeaHy+*sH99CØ;,.@
```

B. Proses dekripsi

Selanjutnya dari proses enkripsi yang telah dilakukan sebelumnya, *ciphertext* akhir yang diperoleh akan didekripsi menggunakan modifikasi algoritma Vigenere Cipher dengan penyandian grup simetri S_5, sehingga proses dekripsinya yaitu sebagai berikut:

1. Mengkonversi karakter-karakter pada *ciphertext* akhir dan kunci Vigenere Cipher ke dalam bentuk angka dengan melihat tabel ASCII *printable characters* (Lampiran 2). Sehingga,

Tabel 4.6 Dekripsi Modifikasi Vigenere Cipher

Ciphertext Akhir

C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	C_{16}
.	5	5	N	*e*	'	:	?	8	1	*s*	;	)	6	G	G
46	53	53	78	101	39	58	63	56	49	115	59	41	54	71	71

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y
69	65	83	89	69	65	83	89	69	65	83	89	69	65	83	89

Subtitusikan *ciphertext* akhir dan kunci Vigenere Cipher ke dalam persamaan (4.2)

$$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

72	83	65	84	32	69	70	69	82	79	32	65	67	84	83	77
H	S	A	T		E	F	E	R	O		A	C	T	S	M
P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}	P_{14}	P_{15}	P_{16}

Lanjutan **Tabel 4.6** Dekripsi Modifikasi Vigenere Cipher

Ciphertext Akhir

C_{17}	C_{18}	C_{19}	C_{20}	C_{21}	C_{22}	C_{23}	C_{24}	C_{25}	C_{26}	C_{27}	C_{28}	C_{29}	C_{30}	C_{31}
+	$	5	=	8	a	9	I	3	4	s	?	+	%	s
43	36	53	61	56	97	57	73	51	52	115	63	43	37	115

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S
69	65	83	89	69	65	83	89	69	65	83	89	69	65	83

Subtitusikan *ciphertext* akhir dan kunci Vigenere Cipher ke dalam persamaan (4.2)

$$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

69	66	65	67	82	32	69	79	77	82	32	69	69	67	32
E	B	A	C	R		E	O	M	R		E	E	C	
P_{17}	P_{18}	P_{19}	P_{20}	P_{21}	P_{22}	P_{23}	P_{24}	P_{25}	P_{26}	P_{27}	P_{28}	P_{29}	P_{30}	P_{31}

Lanjutan **Tabel 4.6** Dekripsi Modifikasi Vigenere Cipher

Ciphertext Akhir

C_{32}	C_{33}	C_{34}	C_{35}	C_{36}	C_{37}	C_{38}	C_{39}	C_{40}	C_{41}	C_{42}	C_{43}	C_{44}	C_{45}	C_{46}
;	:	$	=	M	)	#	C	L	+	a	G	B	:	-
59	58	36	61	77	20	35	67	76	43	97	71	66	58	45

Kunci Vigenere Cipher

K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2
Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A
89	69	65	83	89	69	65	83	89	69	65	83	89	69	65

Subtitusikan *ciphertext* akhir dan kunci Vigenere Cipher ke dalam persamaan (4.2)

$$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

65	84	66	73	83	67	65	79	82	69	32	83	72	84	75
A	T	B	I	S	C	A	O	R	E		S	H	T	K
P_{32}	P_{33}	P_{34}	P_{35}	P_{36}	P_{37}	P_{38}	P_{39}	P_{40}	P_{41}	P_{42}	P_{43}	P_{44}	P_{45}	P_{46}

Lanjutan **Tabel 4.6** Dekripsi Modifikasi Vigenere Cipher

Ciphertext Akhir

C_{47}	C_{48}	C_{49}	C_{50}	C_{51}	C_{52}	C_{53}	C_{54}	C_{55}	C_{56}	C_{57}	C_{58}	C_{59}	C_{60}	C_{61}
K	y	+	′	H	H	+	a	8	L	+	*	;	y	)
75	121	43	39	72	72	43	97	56	76	43	42	59	121	41

Kunci Vigenere Cipher

K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1
S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E
83	89	69	65	83	89	69	65	83	89	69	65	83	89	69

Subtitusikan *ciphertext* akhir dan kunci Vigenere Cipher ke dalam persamaan (4.2)

$$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

87	32	69	69	84	78	69	32	68	82	69	72	71	32	67
W		E	E	T	N	E		D	R	E	H	G		C
P_{47}	P_{48}	P_{49}	P_{50}	P_{51}	P_{52}	P_{53}	P_{54}	P_{55}	P_{56}	P_{57}	P_{58}	P_{59}	P_{60}	P_{61}

Lanjutan **Tabel 4.6** Dekripsi Modifikasi Vigenere Cipher

Ciphertext Akhir

C_{62}	C_{63}	C_{64}	C_{65}	C_{66}	C_{67}	C_{68}	C_{69}	C_{70}	C_{71}	C_{72}	C_{73}	C_{74}	C_{75}	C_{76}
2	5	C	.	4	s	M	+	9	9	y	+	5	I	N
50	53	67	46	52	115	77	43	57	57	121	43	53	73	78

Kunci Vigenere Cipher

K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4
A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S	Y
65	83	89	69	65	83	89	69	65	83	89	69	65	83	89

Subtitusikan *ciphertext* akhir dan kunci Vigenere Cipher ke dalam persamaan (4.2)

$$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

80	65	73	72	82	32	83	69	87	69	32	69	83	85	84
P	A	I	H	R		S	E	W	E		E	S	U	T
P_{62}	P_{63}	P_{64}	P_{65}	P_{66}	P_{67}	P_{68}	P_{69}	P_{70}	P_{71}	P_{72}	P_{73}	P_{74}	P_{75}	P_{76}

Lanjutan **Tabel 4.6** Dekripsi Modifikasi Vigenere Cipher

Ciphertext Akhir

C_{77}	C_{78}	C_{79}	C_{80}	C_{81}	C_{82}	C_{83}	C_{84}	C_{85}	C_{86}	C_{87}	C_{88}	C_{89}	C_{90}	C_{91}
e	&	L	?	\	0	9	C	9	#	@	?	e	[	9
101	38	76	63	92	48	57	67	57	35	64	63	101	91	57

Kunci Vigenere Cipher

K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3
E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A	S
69	65	83	89	69	65	83	89	69	65	83	89	69	65	83

Subtitusikan *ciphertext* akhir dan kunci Vigenere Cipher ke dalam persamaan (4.2)

$$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

32	68	88	69	118	78	69	73	83	65	76	69	32	121	69
	D	X	E	V	N	E	I	S	A	L	E		Y	E
P_{77}	P_{78}	P_{79}	P_{80}	P_{81}	P_{82}	P_{83}	P_{84}	P_{85}	P_{86}	P_{87}	P_{88}	P_{89}	P_{90}	P_{91}

Lanjutan **Tabel 4.6** Dekripsi Modifikasi Vigenere Cipher

Ciphertext Akhir

C_{92}	C_{93}	C_{94}	C_{95}	C_{96}	C_{97}	C_{98}	C_{99}	C_{100}	C_{101}	C_{102}	C_{103}	C_{104}	C_{105}	C_{106}
y	9	$*$	H	?	+	a	H	r	,	6	B	I	E	a
121	57	42	72	63	43	97	72	114	44	54	66	73	101	97

Kunci Vigenere Cipher

K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2
Y	E	A	S	Y	E	A	S	Y	E	A	S	Y	E	A
89	69	65	83	89	69	65	83	89	69	65	83	89	69	65

Subtitusikan *ciphertext* akhir dan kunci Vigenere Cipher ke dalam persamaan (4.2)

$$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

32	83	72	84	69	69	32	84	120	70	84	78	79	32	32
	S	H	T	E	E		T	X	F	T	N	O		
P_{92}	P_{93}	P_{94}	P_{95}	P_{96}	P_{97}	P_{98}	P_{99}	P_{100}	P_{101}	P_{102}	P_{103}	P_{104}	P_{105}	P_{106}

Lanjutan **Tabel 4.6** Dekripsi Modifikasi Vigenere Cipher

Ciphertext Akhir

C_{107}	C_{108}	C_{109}	C_{110}	C_{111}	C_{112}	C_{113}	C_{114}	C_{115}	C_{116}	C_{117}	C_{118}	C_{119}	C_{120}
H	y	+	*	S	H	9	9	C	∅	;	,	.	@
72	121	43	42	115	72	57	57	67		59	44	46	64

Kunci Vigenere Cipher

K_3	K_4	K_1	K_2	K_3	K_4	K_1	K_2	K_3		K_4	K_1	K_2	K_3
S	Y	E	A	S	Y	E	A	S		Y	E	A	S
83	89	69	65	83	89	69	65	83		89	69	65	83

Subtitusikan *ciphertext* akhir dan kunci Vigenere Cipher ke dalam persamaan (4.2)

$$P_i = \big((C_i - 32 - K_r)\ mod\ 95\big) + 32$$

Selanjutnya, hasil perhitungannya dikembalikan ke dalam bentuk karakter

84	32	69	72	32	78	83	87	79		65	70	76	76
T		E	H		N	S	W	O	∅	A	F	L	L
P_{107}	P_{108}	P_{109}	P_{110}	P_{111}	P_{112}	P_{113}	P_{114}	P_{115}	P_{116}	P_{117}	P_{118}	P_{119}	P_{120}

Maka diperoleh *ciphertext1* yaitu

HSAT spEFEROspACTSMEBACRspEOMRspEECspATNISCAOREspSHTKW

spEETNEspDREHGspCPAIHRspSEWEspESUTspDXEVNEISALEspYEspS

HTEEspTXFTNO$sp$$sp$T$spEHsp$NSWO∅AFLL

2. Kemudian menginvers kunci grup simetri yang telah disepakati yaitu:

$$K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix}$$

3. *Ciphertext1* terlebih dahulu diubah menjadi blok-blok yang terdiri dari 5 karakter sehingga diperoleh sebagai berikut:

HSATsp	EFORE	spACTS	MEBAC
RspEOM	RspEEC	spATNI	SCAOR
EspSHT	KWspEE	TNEspD	REHGsp

CPAIH	R*sp*SEW	E*sp*ESU	T*sp*DXE
VNEIS	ELY*sp*A	E*sp*SHT	EE*sp*TX
FTNO*sp*	*sp*T*sp*EH	*sp*NSWO	∅AFLL

4. Setelah itu setiap blok diubah menjadi seperti di bawah ini dengan menggunakan kunci invers.

Blok 1: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ H & S & A & T & sp \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ A & S & sp & T & H \end{pmatrix}$

Blok 2: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & F & E & R & O \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ E & F & O & R & E \end{pmatrix}$

Blok 3: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ sp & A & C & T & S \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ C & A & S & T & sp \end{pmatrix}$

Blok 4: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ M & E & B & A & C \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ B & E & C & A & M \end{pmatrix}$

Blok 5: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ R & sp & E & O & M \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ E & sp & M & O & R \end{pmatrix}$

Blok 6: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ R & sp & E & E & C \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ E & sp & C & E & R \end{pmatrix}$

Blok 7: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ sp & A & T & N & I \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ T & A & I & N & sp \end{pmatrix}$

Blok 8: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ S & C & A & O & R \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ A & C & R & O & S \end{pmatrix}$

Blok 9: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & sp & S & H & T \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ S & sp & T & H & E \end{pmatrix}$

Blok 10: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ K & W & sp & E & E \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ E & W & E & E & K \end{pmatrix}$

Blok 11: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ T & N & E & sp & D \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ E & N & D & sp & T \end{pmatrix}$

Blok 12: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ R & E & H & G & sp \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ H & E & sp & G & R \end{pmatrix}$

Blok 13: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ C & P & A & I & H \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ A & P & H & I & C \end{pmatrix}$

Blok 14: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ R & sp & S & E & W \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ S & sp & W & E & R \end{pmatrix}$

Blok 15: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & sp & E & S & U \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ E & sp & U & S & E \end{pmatrix}$

Blok 16: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ T & sp & D & X & E \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ D & sp & E & X & T \end{pmatrix}$

Blok 17: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ V & N & E & I & S \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ E & N & S & I & V \end{pmatrix}$

Blok 18: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ A & L & E & sp & Y \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ E & L & Y & sp & A \end{pmatrix}$

Blok 19: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & sp & S & H & T \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ S & sp & T & H & E \end{pmatrix}$

Blok 20: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ E & E & sp & T & X \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ sp & E & X & T & E \end{pmatrix}$

Blok 21: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ F & T & N & O & sp \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ N & T & sp & O & F \end{pmatrix}$

Blok 22: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ sp & T & sp & E & H \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ sp & T & H & E & sp \end{pmatrix}$

Blok 23: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ sp & N & S & W & O \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ S & N & O & W & sp \end{pmatrix}$

Blok 24: $K^{-1} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_2 & a_5 & a_4 & a_1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ \emptyset & A & F & L & L \\ a_3 & a_2 & a_5 & a_4 & a_1 \\ F & A & L & L & \emptyset \end{pmatrix}$

Sehingga diperoleh *plaintext* yaitu

AS THE FORECAST BECAME MORE CERTAIN ACROSS THE WEEKEND THE GRAPICHS WERE USED EXTENSIVELY AS THE EXTENT OF THE SNOW FALL

4.3 Uji Hasil Enkripsi Algoritma Modifikasi Vigenere Cipher

Berikut merupakan uji hasil enkripsi algoritma modifikasi Vigenere Cipher kemudian dibandingkan dengan algoritma Vigenere Cipher Biasa. Pengujian dilakukan menggunakan metode Kasiski untuk menemukan panjang kuncinya dan metode *exhaustive key search* untuk mengetahui kemungkinan kunci yang digunakan.

Tabel 4.7 Hasil Enkripsi Algoritma Modifikasi Vigenere Cipher

Ciphertext Algoritma Modifikasi Vigenere Cipher	Kunci	Ciphertext Algortima Vigenere Cipher
.55**Ne':**?81**s;**)6GG+$5=8a9I3 4s?+%**s;:**$=M)#CL**+a**GB:-K**y+'** HH**+a**8**L+***;y)25C.4sM+**99y+**5I **Ne**&L?\0**9C**9#@?e[**9y**9*H?**+a**Hr ,6BIeaHy**+***sH**99C**Ø;,.@	EASY	ES**LFI**FGPICSQXBWAEMWKSRWA IRLYMNSAVOKQXHWUIECCRD**LF I**GJYTHAAWWWWIUKCH**EPRIN**KG ZEDWES**LFIEPRIN**IMJTZCWNGU JADJ

Perhatikan **Tabel 4.7** terlihat bahwa hasil enkripsi modifikasi Vigenere Cipher terdapat huruf kapital, huruf kecil, angka, dan simbol. Hal tersebut menunjukkan bahwa algoritma modifikasi Vigenere Cipher dapat mendukung penggunaan huruf kapital, huruf kecil, angka, dan simbol. Namun juga masih terlihat adanya perulangan kriptogram pada *ciphertext* modifikasi Vigenere Cipher. Berikut tabel yang memuat beberapa kriptogram yang sering muncul, beserta jarak antar kriptogram dan faktor pembaginya.

Tabel 4.8 Metode Kasiski

Kriptogram	Jarak	Faktor pembagi
Ne	72	2, 3, 4, 6, 8, 9, 12, 14, 24, 36, 72
s;	20	2, 4, 5, 10, 20
+a(1)	12	2, 3, 4, 6, 12
+a(2)	44	2, 4, 11. 22, 44
y+	24	2, 3, 4, 6, 8, 12, 24
y+	36	2, 3, 4, 9, 12, 18, 36
L+	16	2, 4, 8, 16
99	43	43
9y	20	2, 4, 5, 10, 20
9C	29	29
+*	52	2, 4, 13, 26, 52

Berdasarkan **Tabel 4.8** terlihat bahwa irisan paling banyak dari semua faktor pembagi adalah 2 dan 4. Jadi, dapat diperkirakan panjang kunci yang digunakan adalah 2 atau 4. Kita asumsikan terlebih dahulu panjang kunci adalah 4 karakter (tanpa mengabaikan kemungkinan panjang kuncinya 2 karakter). Kemudian untuk menentukan kemungkinan karakter kunci digunakan metode *exhaustive key search* yaitu dengan mencoba semua kemungkinan kunci yang panjangnya 4 karakter. Karena yang digunakan pada modifikasi Vigenere Cipher berdasarkan ASCII *printable characters* yang memiliki karakter sebanyak 95 karakter. Sehingga cara ini membutuhkan usaha percobaan sebanyak 95^p , untuk p adalah panjang kunci. Jadi, terdapat $95^4 = 8.145.065$ kemungkinan kunci. Sedangkan pada Vigenere Cipher biasa $26^4 = 456.976$. Hal tersebut membuktikan bahwa modifikasi Vigenere Cipher lebih aman jika dibandingkan dengan Vigenere Cipher biasa. Walaupun demikia n masih terdapat kemungkinan untuk terpecahkan,

oleh sebab itu *plaintext* yang sebenarnya disembunyikan menggunakan penyandian grup simetri. Jadi, apabila dilakukan pembobolan menggunakan metode Kasiski dan *exhaustive key search* tidak akan dapat menemukan *plaintext* yang sebenarnya.

4.4 Kajian Keislaman

Penyandian pesan menggunakan kriptografi adalah bentuk upaya yang dapat dilakukan seseorang yang ingin menyampaikan pesan rahasia kepada seseorang yang hanya berhak menerimanya. Menjaga pesan rahasia tersebut tetap aman hingga sampai kepada yang berhak menerimanya adalah hal yang harus dilakukan sebagaimana amanah yang dititipkan kepadanya. Sebagai seorang mukmin hendaklah menjaga amanah dengan sebaik-baiknya, agar tidak termasuk golongan orang-orang yang munafik. Sebagaimana sabda Rasulullah SAW: "tanda-tanda orang munafik itu ada tiga yaitu apabila berbicara dia berdusta, apabila berjanji dia mengingkari, dan apabila diberi amanah dia berkhianat" (HR. Al-Bukhari).

Bahkan karena sangat pentingnya menjaga amanah, kita dianjurkan untuk tidak berkhianat kepada orang yang mengkhianati kita. Sebagaimana sabda Rasulullah SAW: "tunaikanlah amanah kepada orang yang mempercayaimu dan janganlah kamu berkhianat kepada orang yang mengkhianatimu".

Pentingnya menjaga amanah juga dijelaskan dalam Al-Qur'an surat Al-Anfal ayat 27 yang berarti:

"Wahai orang-orang yang beriman! Janganlah kamu mengkhianati Allah dan Rasul dan (juga) janganlah kamu mengkhianati amanat yang dipercayakan kepadamu, sedang kamu mengetahui".

Berdasarkan ayat tersebut, Allah memerintahkan untuk tidak berkhianat kepada Allah dan Rasulullahnya. Selain itu, dalam ayat ini diperintahkan agar menjaga amanah yang dipercayakan kepada kita. Menjaga suatu informasi atau pesan rahasia juga merupakan amanah yang harus disampaikan kepada yang berhak menerimanya, agar informasi atau pesan rahasia tersebut tidak disalahgunakan oleh pihak yang tidak berwenang. Oleh karena itu, perlu dilakukan usaha untuk mengamankan pesan rahasia tersebut dengan menyandikannnya menggunakan teknik penyandian yang sulit dipecahkan.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pembahasan dapat ditarik kesimpulan sebagai berikut: Modifikasi Vigenere cipher yang telah dilakukan yaitu dengan mengacak *plaintext* menggunakan penyandian grup simetri S_n sebelum dioperasikan dengan kunci Vigenere Cipher. Grup simetri S_n yang digunakan haruslah tidak komutatif. Proses penyandian diawali dengan membentuk kunci grup simetri menggunakan protokol perjanjian kunci Stickel atas grup simetri S_n. Kemudian dienkripsi menggunakan teknik transposisi (permutasi). Setelah *plaintext* teracak menggunakan penyandian grup simetri, maka dapat dioperasikan dengan kunci Vigenere Cipher menggunakan persamaan $C_i = \big((P_i - 32 + K_i)\ mod\ 95\big) + 32$. Sedangkan untuk proses dekripsi menggunakan persamaan $P_i = \big((C_i - 32 - K_i)\ mod\ 95\big) + 32$, kemudian didekripsi lagi menggunakan penyandian grup simetri.

Hasil enkripsi modifikasi Vigenere Cipher terdapat huruf kapital, huruf kecil, angka, dan simbol. Hal tersebut menunjukkan bahwa algoritma modifikasi Vigenere Cipher dapat mendukung penggunaan huruf kapital, huruf kecil, angka, dan simbol. Kemudian, apabila dilakukan kriptanalisis secara *exhaustive key search* terhadap hasil enkripsi modifikasi Vigenere Cipher maka dibutuhkan usaha percobaan sebanyak 95^p. Sedangkan pada Vigenere Cipher dibutuhkan usaha percobaan sebanyak 26^p. Hal tersebut menunjukkan bahwa modifikasi Vigenere Cipher lebih sulit untuk dipecahkan. Selain itu, pada penyandian modifikasi Vigenere Cipher *plaintext* yang sebenarnya telah diacak terlebih dahulu

menggunakan penyandian grup simetri, sehingga aman terhadap serangan metode Kasiski dan *exhaustive key search*. Hal tersebut membuktikan bahwa algoritma modifikasi Vigenere Cipher memiliki tingkat keamanan yang lebih kuat jika dibandingkan dengan Vigenere Cipher.

5.2 Saran untuk Penelitian Lanjutan

Penelitian ini memodifikasi Vigenere Cipher menggunakan grup simetri S_n untuk mengamankan pesan teks. Saran untuk penelitian selanjutnya yaitu melakukan modifikasi menggunakan metode lain yang dapat meningkatkan keamanan menjadi lebih kuat dan sulit untuk dipecahkan atau melakukan pengembangan algortima modifikasi Vigenere Cipher supaya dapat menyandikan pesan yang berupa gambar, suara, atau video. Penelitian ini hanya membahas tingkat keamanan *ciphertext* pada pesan berbahasa Inggris. Oleh karena itu, untuk penelitian selanjutnya disarankan meneliti tingkat keamanan *ciphertext* pada pesan berbahasa Indonesia.

DAFTAR PUSTAKA

Abdussakir. (2009). *Matematika 1 Kajian Integratif Matematika dan Al-Qur'an.* Malang: UIN Malang Press.

Abidin, Z., & Khairudin, F. (2017). Penafsiran Ayat-Ayat Amanah dalam Al-Qur'an. *Jurnal Syahadah*.

Aliyu, A.-A. M., & Olaniyan, A. (2016). Vigenere Cipher: Trends, Review and Possible Modifications. *International Journal of Computer Application*, 46-50.

Ariyus, D. (2006). *Kriptografi Keamanan Data dan Komunikasi.* Yogyakarta: Graha Ilmu.

Ariyus, D. (2008). *Pengantar Ilmu Kriptografi Teori Analisis dan Implementasi.* Yogyakarta: C.V ANDI OFFSET.

Bartle, R. G., & Sherbert, D. R. (2000). *Introduction to Real Analysis Third Edition.* New York: John Wiley & Sons.

Danial, E., & Warsiah. (2009). *Metode Penulisan Karya Ilmiah* . Bandung: Laboraterium Pendidikan Kewarganegaraan.

Dummit, D., & Foote, R. (2004). *Abstract Algebra Third Edition.* New York: Prentice-Hall International, Inc.

Gallian, J. A. (2017). *Contemporary Abstract Algebra Ninth Edition.* Boston: Cengage Learning.

Gilbert, L., & Gilbert, J. (2015). *Elements of Modern Algebra, Eighth Edition.* USA: Cengange Learning.

Hariati, A., Hardiyanti, K., & Putri, W. E. (2018). Kombinasi Algoritma Playfair Cipher dengan Metode Zig-Zag dalam Penyandian Teks. *Sinkron*, 13-17.

Latifah, R., Ambo, S. N., & Kurnia, S. I. (2017). Modifikasi Algoritma Caesar Chiper dan Reil Fence Chiper untuk Peningkatan Keamanan Teks Alfanumerik dan Karakter Khusus. *Prosiding Semnastek*.

Mukhtar, H. (2018). *Kriptografi untuk Keamanan Data.* Yogyakarta: Deepublish.

Munir, R. (2019). *Kriptografi.* Bandung: Informatika Bandung.

Musthofa, & Lestari, D. (2014). Metode Perjanjian Password Berdasarkan Operasi Matriks Atas Aljabar Min-Plus untuk Keamanan Pengiriman Informasi Rahasia. *J. Sains Dasar*, 3(1) 25-33.

Myasnikov, A., Shpilrain, V., & Ushakov, A. (2008). *Group-based Cryptography.* Basel Switzerland: Birkhauser Verlag.

Octavianingrum, M., Siambaton, D. A., & Dewi, A. F. (2018). Modifikasi Vigenere Cipher dengan Kunci Geser Metode Enkripsi Blok. *Prosiding Sendika*.

Rachmawati, D., Budiman, M. A., & Aulia, I. (2018). Super-Encryption Using Monoalphabetic Algorithm and XOR Algorithm for Data Security. *Journal of Physic: Conference Series 979.*

Raisinghania, M., & Aggarwal, R. (1980). *Modern Algebra.* New Delhi: S. Chand & Company LTD.

Ruffini, P. (1799). *Teoria Generale Delle Equazioni.* S. Tommaso d'Aquino: Nella Stamperia.

Sadikin, R. (2012). *Kriptografi untuk Keamanan Jaringan.* Yogyakarta: C.V ANDI OFFSET.

Widarma, A., Siregar, H. F., & Irawan, M. (2019). Teknik Keamanan Data Menggunakan Vigenere Cipher dan Electronic Code Book (ECB). *Jurnal Sains Komputer & Informatika*, 393-400.

Wasiatun, R. (2016). "Enkripsi dan Dekripsi Pesan Menggunakan Grup Simetri untuk Mengamankan Informasi". Skripsi. Malang: UIN Malang.

LAMPIRAN

Lampiran 1. Tabel Konversi Huruf Alfabet

Karakter	Angka Desimal
A	0
B	1
C	2
D	3
E	4
F	5
G	6
H	7
I	8
J	9
K	10
L	11
M	12
N	13
O	14
P	15
Q	16
R	17
S	18
T	19
U	20
V	21
W	22
X	23
Y	24
Z	25

Lampiran 2. Tabel ASCII *Printable Characters*

ASCII *Printable Characters*							
Decimal	Character		Decimal	Character		Decimal	Character
32	Space		64	@		96	`
33	!		65	A		97	A
34	“		66	B		98	b
35	#		67	C		99	c
36	$		68	D		100	d
37	%		69	E		101	e
38	&		70	F		102	f
39	'		71	G		103	g
40	(		72	H		104	h
41	)		73	I		105	i
42	*		74	J		106	j
43	+		75	K		107	k
44	,		76	L		108	l
45	-		77	M		109	m
46	.		78	N		110	n
47	/		79	O		111	o
48	0		80	P		112	p
49	1		81	Q		113	q
50	2		82	R		114	r
51	3		83	S		115	s
52	4		84	T		116	t
53	5		85	U		117	u
54	6		86	V		118	v
55	7		87	W		119	w
56	8		88	X		120	x
57	9		89	Y		121	y
58	:		90	Z		122	z
59	;		91	[		123	{
60	<		92	\		124	|
61	=		93	]		125	}
62	>		94	^		126	~
63	?		95	_			

RIWAYAT HIDUP

Niken Dwi Cahyanti lahir di Kediri pada tanggal 22 Juli 1998, biasa dipanggil Niken. Alamatnya berada di Dusun Kandangan RT/RW:02/01 Desa Pagu, Kecamatan Pagu, Kabupaten Kediri. Merupakan anak kedua dari Bapak Supardi dan Ibu Sulastri. Pernah menempuh pendidikan dasar di SDN 1 Pagu dan lulus pada Tahun 2011. Kemudian melanjutkan sekolahnya di SMPN 1 Pagu dan lulus pada Tahun 2014. Selanjutnya, menempuh pendidikan sekolah menengah atas di SMAN 1 Gurah lulus pada Tahun 2017. Pada Tahun 2017 melanjutkan pendidikan di Universitas Islam Negeri Maulana Malik Ibrahim Malang dan mengambil Jurusan Matematika, Fakultas Sains dan Teknologi.

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 : Niken Dwi Cahyanti
NIM : 17610087
Jurusan : Matematika
Fakultas : Sains dan Teknologi
Judul Skripsi : Modifikasi Vigenere Cipher Menggunakan Grup Simetri untuk Mengamankan Pesan Teks
Pembimbing I : Prof. Dr. H. Turmudi, M.Si., Ph.D.
Pembimbing II: Muhammad Khudzaifah, M.Si

No	Tanggal	Hal	Tanda Tangan
1	3 Mei 2021	Konfirmasi Bimbingan Proposal Skripsi	1
2	3 Mei 2021	Konfirmasi Bimbingan Proposal Skripsi	2
3	18 Juni 2021	Bimbingan Bab I, II, III	3
4	15 Juni 2021	Bimbingan Bab I dan Kajian Keislaman	4
5	26 Juni 2021	Bimbingan Bab I, II, III	5
6	5 Juli 2021	Revisi Bab I, II, III	6
7	26 Juli 2021	Konsultasi Bab I, II, III	7
8	4 Agustus 2021	Revisi Bab I, II, III	8
9	24 September 2021	Acc Pendaftaran Seminar Proposal	9
10	29 September	Konsultasi kajian keislaman, Bab III, dan Acc Pendaftaran Seminar Proposal	10
11	24, 26, 29 November 2021	Bimbingan Revisi Seminar Proposal Bab I-V	11
12	16, 29 November 2021	Bimbingan Revisi Seminar Proposal Bab I-V	12

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

13	18 Februari 2022	Bimbingan Revisi Bab IV dan V	13
14	4 Maret 2022	Bimbingan Revisi dan Acc Seminar Hasil	14
15	10 Maret 2022	Konsultasi dan Acc Seminar Hasil	15
16	24 Mei 2022	Konsultasi Revisi Seminar Hasil dan Acc Sidang	16
17	24 Mei 2022	Konsultasi Revisi Seminar Hasil dan Acc Sidang	17

Malang, 23 Juni 2022
Mengetahui,
Ketua Program Studi
Matematika

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