ENKRIPSI DAN DEKRIPSI PESAN MENGGUNAKAN ALGORITMA RSA DAN *AFFINE CIPHER* DENGAN METODE MATRIKS

SKRIPSI

OLEH
MUHAMAD WAIS AL QORNY
NIM. 13610118

JURUSAN MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2018

ENKRIPSI DAN DEKRIPSI PESAN MENGGUNAKAN ALGORITMA RSA DAN *AFFINE CIPHER* DENGAN METODE MATRIKS

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
Muhamad Wais Al Qorny
NIM. 13610118

JURUSAN MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2018

ENKRIPSI DAN DEKRIPSI PESAN MENGGUNAKAN ALGORITMA RSA DAN *AFFINE CIPHER* DENGAN METODE MATRIKS

SKRIPSI

Oleh
Muhamad Wais Al Qorny
NIM. 13610118

Telah Diperiksa dan Disetujui untuk Diuji
Tanggal 03 Mei 2018

Pembimbing I,	Pembimbing II,
Dr. Abdussakir, M.Pd	Ari Kusumastuti, M.Pd., M.Si
NIP. 19751006 200312 1 001	NIP. 19770521 200501 2 004

Mengetahui,
Ketua Jurusan Matematika

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

ENKRIPSI DAN DEKRIPSI PESAN MENGGUNAKAN ALGORITMA RSA DAN *AFFINE CIPHER* DENGAN METODE MATRIKS

SKRIPSI

Oleh
Muhamad Wais Al Qorny
NIM. 13610118

Telah Dipertahankan di Depan Dewan Penguji Skripsi dan
Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S.Mat)
Tanggal 31 Mei 2018

Penguji Utama	: Dr. H. Imam Sujarwo, M.Pd	.....
Ketua Penguji	: H. Wahyu H. Irawan, M.Pd	.....
Sekretaris Penguji	: Dr. Abdussakir, M.Pd	.....
Anggota Penguji	: Ari Kusumastuti, M.Pd., M.Si	.....

Mengetahui,
Ketua Jurusan Matematika

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

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Muhamad Wais Al Qorny

NIM : 13610118

Jurusan : Matematika

Fakultas : Sains dan Teknologi

Judul Skripsi : Enkripsi dan Dekripsi Pesan Menggunakan Algoritma RSA dan *Affine Cipher* dengan Metode Matriks

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

Malang, 03 Mei 2018
Yang membuat pernyataan,

Muhamad Wais Al Qorny
NIM. 13610118

MOTO

"Salah satu kunci sukses adalah sabar"

PERSEMBAHAN

Dengan rasa syukur skripsi ini penulis persembahkan untuk:

Kedua orang tua penulis ayah Moch. Toha dan ibu Siti Ismariyah yang selalu memberikan doa dukungan dan lain sebagainya yang mungkin tidak bisa penulis balas dengan apapun dan kakak-kakak tersayang Dewi Puspita Sari dan Ismamun Toha Putri yang selalu memberikan doa dan dukungan kepada penulis.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Segala puji bagi Allah Swt atas rahmat, taufik serta hidayah-Nya sehingga penulis mampu menyelesaikan skripsi ini sebagai salah satu syarat untuk memperoleh gelar sarjana dalam bidang matematika di Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Shalawat serta salam kepada nabi Muhammad Saw yang telah membimbing umat manusia menuju jalan yang terang.

Proses penyusunan skripsi ini, penulis mendapat banyak bimbingan dan arahan dari berbagai pihak. Untuk itu penulis memberikan ucapan terima kasih kepada:

1. Prof. Dr. H. Abd. Haris, M.Ag, selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Sri Harini, M.Si, selaku dekan Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim.
3. Dr. Usman Pagalay, M.Si, selaku ketua Jurusan Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Dr. Abdussakir M.Pd, selaku dosen pembimbing I yang telah memberikan ide mengenai permasalahan skripsi ini serta meluangkan waktunya untuk memberikan bimbingan dengan baik sehingga penulis dapat menyelesaikan skripsi ini.
5. Ari Kusumastuti, M.Pd, M.Si, selaku dosen pembimbing II yang telah

memberikan bimbingan, arahan, dan berbagai ilmunya kepada penulis.

6. Muhammad Khudzaifah, M.Si, selaku dosen yang selalu memberikan dukungan, bimbingan, arahan, dan berbagai ilmunya kepada penulis.
7. Segenap sivitas akademika Jurusan Matematika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang terutama seluruh dosen, terima kasih atas ilmu dan bimbingannya.
8. Segenap keluarga terutama Ayah dan Ibu yang selalu memberikan doa, semangat, serta motivasi kepada penulis sampai saat ini.
9. Seluruh teman-teman di Jurusan Matematika angkatan 2013 yang telah banyak memberikan semangat, motivasi, dan arahan untuk mengerjakan skripsi secara baik dan cepat.
10. Seluruh pihak yang ikut membantu dalam menyelesaikan skripsi ini baik moril maupun materiil.

Penulis berharap supaya skripsi ini dapat bermanfaat bagi pembaca maupun bagi penulis.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Malang, Mei 2018

Penulis

DAFTAR ISI

DAFTAR GAMBAR

ABSTRAK

Qorny, Muhamad Wais Al. 2018. **Enkripsi dan Dekripsi Pesan Menggunakan Algoritma RSA dan *Affine Cipher* dengan Metode Matriks**. Skripsi. Jurusan Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Abdussakir, M.Pd (II) Ari Kusumastuti, M.Pd., M.Si.

Kata Kunci: Keamanan, Enkripsi, Dekripsi, RSA, *Affine Cipher*, Matriks.

Enkripsi merupakan proses mengubah suatu yang terbaca menjadi tidak terbaca, sedangkan dekripsi adalah kebalikan proses enkripsi yaitu mengubah suatu yang tidak terbaca menjadi terbaca. Terdapat dua algoritma yang sering digunakan yaitu algoritma simetri dan asimetri. Umumnya algoritma simetri cepat dalam proses enkripsi dan dekripsi tetapi kuncinya kurang aman, sedangkan algoritma asimetri umumnya lama dalam proses enkripsi dan dekripsi tetapi kuncinya sangat aman. Untuk mendapatkan proses enkripsi dan dekripsi yang cepat dan keamanan kunci yang kuat maka dapat menggabungkan algoritma simetri dengan asimetri yang disebut dengan algoritma hibrida. Pada penelitian ini proses enkripsi dan dekripsi pesan menggunakan algoritma simetri yaitu *affine cipher* dengan metode matriks menggunakan kunci sesi, sedangkan untuk mengamankan kunci sesinya menggunakan algoritma asimetri yaitu RSA dengan kunci publik. Hasil yang didapatkan setelah proses enkripsi pesan adalah perubahan setiap karakter lebih dari satu selain itu kunci yang digunakan tidak terbatas hanya bergantung pada ukuran matriks dan determinannya harus relatif prima dengan modulo yang digunakan. Selain itu kunci sesi yang digunakan untuk mengenkripsi pesan juga diamankan dengan RSA yang terkenal sulitnya memfaktorkan bilangan bulat besar untuk mendapatkan faktor primanya. Keamanan proses enkripsi terletak pada keamanan kunci simetri sedangkan keamanan proses dekripsi terletak pada keamanan kunci asimetri. Pada penelitian selanjutnya disarankan menggunakan metode lain dalam mengamankan pesan.

ABSTRACT

Qorny, Muhamad Wais Al. 2018. **Encryption and Decryption of Message Using RSA and Affine Cipher Algorithms with Matrix Method**. Thesis. Department of Mathematics, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang. Promotor: (I) Dr. Abdussakir, M.Pd (II) Ari Kusumastuti, M.Pd., M.Si.

Keywords: Safety, Encryption, Decryption, RSA, Affine Cipher, Matrix.

Encryption is the process of changing something readable becomes unreadable, while decryption is the reverse of the encryption process which is to change something unreadable into the readable one. There are two commonly used cryptography algorithms, the symmetry and asymmetry algorithms. Generally, symmetry algorithms fast in the process of encryption and decryption but the key is less secure, while the asymmetry algorithm is generally long in the process of encryption and decryption but the key is very secure. To get a fast encryption and decryption process and strong key security it can combine symmetry with asymmetry algorithm called hybrid algorithm. In this research, the process of encryption and decryption of message used symmetry algorithm that is affine cipher with matrix method with session key, while to secure session key it used asymmetry algorithm that is RSA with public key. The results obtained after encryption process of the message is the change of each character more than one, additionally the key used is not limited only depending on the size of the matrix and the determinant must be relative prime with the modulo used. In addition, session key that are used to encrypt message are also secure with RSA which is famously difficult to factor large integers to obtain their prime factor. The security of the encryption process lies in the security of the symmetry key while the security of the decryption process lies in the security of asymmetry key. In the next research, it is suggested to use other methods in securing the message.

ملخص

قوربني، محمد وإيس آل . ٢٠١٨. **تشفير وفك التشفيرعن الرسائل باستخدام الخوارزمية RSAو *AffineCipher* بطريقة مصفوفة**. البحث الجا معي. شعبة الرياضية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف. (١) : الدكتور عبد الشاكر, الماجستير، (٢) أري كوسوماستوتي، الماجستير، الماجستير.

الكلمات الرئيسية: الأمن، التشفير، فك التشفير، RSA، *Affine Cipher*, مصفوفة

في العصر الحديث أصبح أمن البيانات مهما جدا. هناك العديد من الطرق التي يمكن استخدامها منها هو التشفير وفك التشفير. التشفير هو عملية تغيير الشيء المقروء إلى شيء غير مقروء،أما فك التشفير هو عكس عملية التشفير وهو تغيير الشيء غير مقروء إلى شيء يمكن قراءته. هناك نوعان من خوارزميات المستخدمة عادة: خوارزمية *asymmetry* و*symmetry*. عادة خوارزمية *symmetry* سريع في عملية التشفير وفك التشفير ولكن أقل أمنا في المفتاح، في حين أن خوارزمية *asymmetry* طويلة بشكل عام في عملية التشفير وفك التشفير ولكن هي آمن جدا في المفتاح. للحصول على عملية التشفير وفك التشفير السريع وأمن المفتاح القوي، فيمكن الجمع بين خوارزمية *symmetry* و *asymmetry* يسمى خوارزمية *hybrid*. في هذا البحث، عملية التشفير وفك تشفير الرسائل باستخدام خوارزمية *symmetry* وهي *affine cipher* بطريقة مصفوفة مع مفتاح الدورة، أما لتأمين مفتاح الدورة باستخدام خوارزمية *symmetry* وهو RSA بالمفتاح العمومي. النتيجة المحصولة عليها بعد عملية تشفير الرسالة هي تغيير كل حرفأكثرمن واحد بخلاف ذللك، لايقتصر المفتاح المستخدم اعتمادًا على حجم المصفوفة ويجب أن يكو ن محددة مستخدمًا بشكل نسبي مع الطريقة المستخدمةز. بلإضافة إلى ذلك، فإن مفاتيح الجلسات المستخدمة لتشفير الرسائل يتم تأمينها أيضًا مع RSA، وهو صعب فهمه في الأعداد الكبيرة للحصول على عو امل أولية لها. يمكن أمان عملية في أمان مفتاح التماثل بينما يكمن أمان عملية فك التشفير في أمان مفتاح *symmetry*. في الدر اسة التالية يقتر ح استخدام طرق أخرى في تأمين الر سالة.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam kehidupan sehari-hari manusia membutuhkan manusia lain, ini karena manusia merupakan makhluk sosial. Dalam kehidupan sosial, manusia akan saling berkomunikasi, berpesan atau beramanat, dan lain sebagainya. Dalam hal tertentu biasanya pihak yang berpesan atau beramanat hanya ingin diketahui oleh pihak tertentu saja sehingga pihak lain tidak mengetahuinya. Oleh karena itu diperlukan suatu keamanan supaya pesan yang akan disampaikan terjaga kerahasiaannya. Untuk menjaga keamanan pesan supaya tetap terjaga kerahasiaannya, maka perlu diberikan suatu perilaku khusus sehingga pesan tersebut tidak dapat diketahui pihak lain dan biasanya membutuhkan kunci untuk membuka kembali pesan tersebut. Dengan demikian pesan yang ingin disampaikan hanya dapat dibaca atau diketahui pihak tertentu saja. Berkaitan dengan menyampaikan pesan kepada yang berhak disinggung dalam al-Quran surat an-Nisaa'/4:58, yaitu:

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 ٥٨

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

Kriptografi adalah salah satu metode untuk mengamankan pesan supaya tetap terjaga kerahasiaannya dengan cara enkripsi dan dekripsi pada pesan.

Enkripsi adalah proses penyandian pesan asli (*plaintext*) menjadi pesan tersandi (*ciphertext*). Untuk proses mengembalikan *ciphertext* menjadi *plaintext* disebut dekripsi. Setiap proses enkripsi dan dekripsi membutuhkan parameter untuk transformasi yang dinamakan kunci (Munir, 2004).

Kriptografi mengalami perkembangan sangat pesat, mulai dari algoritma sederhana hingga yang kompleks. Perkembangan algoritma ini bertujuan supaya pesan yang dienkripsi aman dari serangan kriptaanalisis, yaitu teknik untuk memecahkan kunci enkripsi, sedangkan orangnya disebut kriptaanalis (Kromodimoeljo, 2010). Akibatnya terdapat berbagai macam algoritma kriptografi. Pada dasarnya terdapat dua jenis kriptografi, yaitu kriptografi klasik dan kriptografi modern (Ariyus, 2008).

Dalam kriptografi klasik, teknik enkripsi yang digunakan adalah enkripsi simetris, yaitu kunci dekripsi sama dengan kunci enkripsi (Kromodimoeljo, 2010). Ada dua algoritma yang dapat digunakan yaitu substitusi dan transposisi (permutasi). Kriptografi modern memiliki penerapan yang sama seperti kriptografi klasik tetapi memiliki algoritma yang lebih kompleks (Munir, 2004). Jika dilihat berdasarkan kunci yang digunakan ada tiga jenis kriptografi, yaitu kriptografi klasik (algoritma simetri), kriptografi kunci publik (algoritma asimetri), dan fungsi *hash* (*hash function*) (Ariyus, 2008).

Algoritma simetri sering disebut dengan kriptografi klasik karena menggunakan satu kunci untuk proses enkripsi dan dekripsinya. Apabila kunci tersebut diketahui orang lain maka orang tersebut dapat melakukan proses enkripsi dan dekripsi suatu pesan. Kurang aman jika digunakan untuk mengamankan suatu pesan. Algoritma asimetri sering disebut dengan kriptografi

kunci publik, dengan kata lain kunci yang digunakan untuk melakukan enkripsi dan dekripsi berbeda (Ariyus, 2008). Dalam proses enkripsi, algoritma asimetri menggunakan kunci publik, sedangkan untuk proses dekripsi menggunakan kunci privat. Dengan kata lain pesan yang dienkripsi menggunakan kunci publik hanya dapat didekripsi menggunakan kunci privat (Kromodimoeljo, 2010). Fungsi *hash* merupakan fungsi yang menerima masukan *string* yang panjangnya sebarang dan mengkonversinya menjadi *string* keluaran yang panjangnya tetap (Munir, 2004).

Affine cipher merupakan kriptografi klasik yaitu algoritma substitusi. Algoritma ini mengandalkan kunci berupa dua nilai bilangan bulat (*integer*). Kelebihan *affine cipher* sebagai algoritma substitusi adalah pada barisan bilangan yang berfungsi sebagai pengali dengan kunci (Ariyus, 2008). Untuk proses enkripsi pada suatu pesan yang panjang dapat diubah ke dalam bentuk matriks. Untuk kuncinya dapat dimodifikasi ke dalam bentuk matriks atau disebut matriks enkripsi.

Salah satu algoritma asimetri adalah RSA. RSA adalah singkatan dari para penemunya yaitu Rivest, Shamir, dan Adleman yang nama lengkapnya adalah Rivest, Adi Shamir, dan Len Adleman di MIT (*Massachussets Institute of Technology*) pada tahun 1979 (Stallings, 2005). RSA merupakan algoritma yang banyak digunakan saat ini untuk mengamankan data. RSA menggunakan algoritma pemfaktoran bilangan yang sangat besar, sehingga RSA dianggap paling aman dari serangan kriptaanalisis (Ariyus, 2008). Meskipun begitu, RSA mempunyai kelemahan seperti algoritma asimetri lainnya, yaitu lambat dalam proses enkripsi dan dekripsi daripada algoritma simetri.

Setiap metode kriptografi mempunyai kelebihan dan kelemahan masing-

masing baik dari segi kecepatan enkripsi atau dekripsi maupun dari segi keamanannya. Algoritma simetri memiliki sistem keamanan yang lemah karena kunci yang digunakan untuk proses enkripsi sama dengan kunci yang digunakan untuk dekripsi (Khudzaifah, 2014). Untuk mengatasi kekurangan masing-masing algoritma antara algoritma simetri dan algoritma asimetri maka digunakan kombinasi yang efisien dari algoritma simetri yang ditingkatkan keamanannya dengan algoritma asimetri yang disebut dengan algoritma hibrida (Mollin, 2007).

Pada penelitian Wibowo, dkk (2014) digunakan algoritma *affine cipher* sebagai algoritma enkripsi dan dekripsi yang diimplementasikan pada aplikasi berbasis android. Pada penelitian Hamzah (2011) digunakan algoritma RSA dan *blowfish* sebagai algoritma enkripsi dan dekripsi yang diimplementasikan pada aplikasi delphi 7. Pada penelitian Khudzaifah (2014) digunakan algoritma hibrida yang algoritma simetrinya menggunakan algoritma *quasigroup* dan algoritma asimetrinya menggunakan RSA. Pada penelitian ini algoritma simetri menggunakan algoritma *affine cipher* dan algoritma asimetri menggunakan RSA, yang proses enkripsi dan dekripsi pesan menggunakan metode matriks. Pesan dienkripsi menggunakan kunci sesi dari *affine cipher* dengan metode matriks, sedangkan kunci sesi diamankan menggunakan algoritma asimetri dari RSA dengan kunci publik.

Berdasarkan uraian di atas, maka peneliti perlu menyusunnya dalam penelitian dengan judul "Enkripsi dan Dekripsi Pesan Menggunakan Algoritma RSA dan *Affine Cipher* dengan Metode Matriks".

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah dalam penelitian ini adalah bagaimana analisis keamanan proses enkripsi dan dekripsi menggunakan algoritma RSA dan *affine cipher* dengan metode matriks?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah di atas, maka tujuan yang ingin dicapai dalam penelitian ini adalah untuk mengetahui keamanan proses enkripsi dan dekripsi menggunakan algoritma RSA dan *affine cipher* dengan metode matriks.

1.4 Manfaat Penelitian

Adapun manfaat dalam penelitian ini adalah dapat memahami keamanan proses enkripsi dan dekripsi menggunakan algoritma RSA dan *affine cipher* dengan metode matriks.

1.5 Batasan Masalah

Untuk mendekati sasaran yang diharapkan, maka perlu diadakan pembatasan permasalahan di antaranya adalah:

1. Kunci a yang digunakan untuk enkripsi dan dekripsi pesan menggunakan matriks persegi, yaitu matriks berordo 2×2 sampai 4×4.
2. Kunci b yang digunakan untuk enkripsi dan dekripsi pesan menggunakan matriks kolom, yaitu matriks berordo 2×1 sampai 4×1
3. Pesan diubah ke dalam bentuk matriks yang jumlah barisnya sama dengan jumlah baris dari kunci a, yaitu matriks berordo $2 \times j$ sampai $4 \times j$.

4. Setiap entri yang kosong pada matriks pesan diisi dengan 32 sebelum proses enkripsi untuk diganti menjadi karakter spasi.

1.6 Metode Penelitian

Metode yang digunakan dalam penelitian ini adalah studi literatur (*library research*). Kajian pada buku-buku dan jurnal berkaitan dengan topik enkripsi dan dekripsi dengan metode matriks, algoritma RSA, algoritma *affine cipher*, dan algoritma hibrida. Kajian secara komprehensif meliputi ketiga algoritma tersebut (RSA, *affine cipher*, dan hibrida) dengan metode matriks adalah membuat algoritma RSA dan *affine cipher* dengan metode matriks dalam mengamankan suatu pesan.

Adapun langkah-langkah penelitian ini adalah sebagai berikut:

1. Menganalisis algoritma RSA dan *affine cipher* secara teoritis.
2. Menganalisis enkripsi dan dekripsi algoritma RSA dan *affine cipher* dengan metode matriks secara implementatif.
3. Menyimpulkan kelebihan dan kelemahan enkripsi dan dekripsi algoritma RSA dan *affine cipher* dengan metode matriks.
4. Memberikan contoh simulasi enkripsi dan dekripsi algoritma RSA dan *affine cipher* dengan metode matriks.

1.7 Sistematika Penulisan

Dalam penulisan ini, peneliti menggunakan sistematika penulisan yang terdiri dari empat bab dan masing-masing bab dibagi dalam subbab dengan sistematika penulisan sebagai berikut:

Bab I Pendahuluan

Meliputi latar belakang masalah yang diteliti, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, metode penelitian, dan sistematika penulisan.

Bab II Kajian Pustaka

Berisi teori-teori yang digunakan meliputi definisi, teorema, serta contoh yang berhubungan dengan pembahasan antara lain matriks, keterbagian, kriptografi dan kajian keagamaan.

Bab III Pembahasan

Bab ini berisi penjabaran algoritma RSA dan algoritma *affine cipher*. Mengkonstruksi kedua algoritma RSA dan *affine cipher*. Melakukan enkripsi dan dekripsi pesan dengan metode matriks dan kajian keagamaan.

Bab IV Penutup

Bab ini berisi kesimpulan dari pembahasan dan saran untuk penelitian selanjutnya.

BAB II

KAJIAN PUSTAKA

2.1 Matriks

Matriks merupakan susunan dari bilangan atau elemen yang disusun menurut baris dan kolom. Matriks yang mempunyai m baris dan n kolom disebut matriks berordo $m \times n$. Bilangan yang disusun pada matriks disebut entri pada matriks (Anton dan Rorres, 2010). Matriks disimbolkan dengan huruf kapital dan entrinya disimbolkan dengan huruf non kapital. Matriks $\boldsymbol{A}$ yang berordo $m \times n$ dapat ditulis dengan $\boldsymbol{A}_{m\times n}$. Berikut merupakan bentuk umum matriks $\boldsymbol{A}$ berordo $m \times n$

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}.$$

Matriks yang hanya mempunyai satu baris disebut matriks baris, dan matriks yang hanya mempunyai satu kolom disebut matriks kolom (Anton dan Rorres, 2010). Matriks baris $\boldsymbol{A}$ ditulis dengan $\boldsymbol{A}_{1\times n}$ dan matriks kolom $\boldsymbol{A}$ ditulis dengan $\boldsymbol{A}_{m\times 1}$. Berikut merupakan bentuk umum matriks baris $\boldsymbol{A}$ dan matriks kolom $\boldsymbol{B}$

$$\boldsymbol{A} = [a_{11} \quad a_{12} \quad \cdots \quad a_{1n}] \text{ dan } \boldsymbol{B} = \begin{bmatrix} b_{11} \\ b_{21} \\ \vdots \\ b_{m1} \end{bmatrix}.$$

Matriks yang banyak baris sama dengan banyak kolom dinamakan matriks persegi. Jika matriks persegi $\boldsymbol{A}$ berordo $m \times n$ maka $m = n$. Berikut merupakan bentuk umum matriks persegi $\boldsymbol{A}$ berordo $n \times n$

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}.$$

Entri $a_{11}, a_{22}, \cdots, a_{nn}$ pada matriks persegi dikatakan berada pada diagonal utama. Matriks persegi dengan entri 1 pada diagonal utama dan 0 untuk yang lainnya disebut matriks identitas (Anton dan Rorres, 2010). Matriks identitas disimbolkan dengan huruf $\boldsymbol{I}$. Matriks identitas dengan ukuran $n \times n$ dapat ditulis dengan $\boldsymbol{I}_n$ atau $\boldsymbol{I}_{n\times n}$. Berikut merupakan bentuk umum matriks identitas $\boldsymbol{I}_n$

$$\boldsymbol{I} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}.$$

Dalam aljabar, matriks didefinisikan beberapa operasi yang dikenakan pada matriks, yaitu penjumlahan, pengurangan, perkalian matriks dengan skalar, dan perkalian matriks dengan matriks.

2.1.1 Operasi Matriks

2.1.1.1 Penjumlahan Matriks

Penjumlahan dua matriks yang berordo sama adalah penjumlahan bersama-sama entri yang bersesuaian dalam kedua matriks tersebut. Jika dua matriks $\boldsymbol{A} = [a_{ij}]$ dan $\boldsymbol{B} = [b_{ij}]$ mempunyai ukuran sama, maka jumlah matriks $\boldsymbol{A}$ dengan matriks $\boldsymbol{B}$ diperoleh dari penjumlahan entri matriks $\boldsymbol{B}$ yang bersesuaian dengan entri matriks $\boldsymbol{A}$ (Anton dan Rorres, 2010). Misal matriks $\boldsymbol{A}$ dan $\boldsymbol{B}$ berordo sama yaitu $m \times n$, maka $\boldsymbol{A} + \boldsymbol{B}$ dapat ditulis

$$\boldsymbol{A} + \boldsymbol{B} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mn} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} & \cdots & a_{1n}+b_{1n} \\ a_{21}+b_{21} & a_{22}+b_{22} & \cdots & a_{2n}+b_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1}+b_{m1} & a_{m2}+b_{m2} & \cdots & a_{mn}+b_{mn} \end{bmatrix}.$$

Dari penjelasan di atas diketahui bahwa ordo yang sama menjadi syarat perlu yang harus dipenuhi supaya penjumlahan dua matriks atau lebih dapat terpenuhi. Hasil penjumlahan dua matriks memiliki ordo yang sama dengan kedua matriks.

Contoh 1:

Misalkan matriks $\boldsymbol{A}$ dan $\boldsymbol{B}$ berordo 2×2

$$\boldsymbol{A} = \begin{bmatrix} 4 & 8 \\ 2 & 9 \end{bmatrix} \text{ dan } \boldsymbol{B} = \begin{bmatrix} 7 & 4 \\ 5 & 12 \end{bmatrix},$$

maka

$$\boldsymbol{A}+\boldsymbol{B} = \begin{bmatrix} 4 & 8 \\ 2 & 9 \end{bmatrix} + \begin{bmatrix} 7 & 4 \\ 5 & 12 \end{bmatrix} = \begin{bmatrix} 4+7 & 8+4 \\ 2+5 & 9+12 \end{bmatrix} = \begin{bmatrix} 11 & 12 \\ 7 & 21 \end{bmatrix}.$$

2.1.1.2 Pengurangan Matriks

Jika dua matriks $\boldsymbol{A} = [a_{ij}]$ dan $\boldsymbol{B} = [b_{ij}]$ mempunyai ukuran sama, maka selisih matriks $\boldsymbol{A}$ dengan matriks $\boldsymbol{B}$ yang ditulis $\boldsymbol{A}-\boldsymbol{B}$ diperoleh dari pengurangan entri matriks $\boldsymbol{A}$ yang bersesuaian dengan entri matriks $\boldsymbol{B}$ (Anton dan Rorres, 2010). Misal matriks $\boldsymbol{A}$ dan $\boldsymbol{B}$ berordo sama yaitu $m \times n$, maka $\boldsymbol{A}-\boldsymbol{B}$ dapat ditulis

$$\boldsymbol{A}-\boldsymbol{B} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} - \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mn} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11}-b_{11} & a_{12}-b_{12} & \cdots & a_{1n}-b_{1n} \\ a_{21}-b_{21} & a_{22}-b_{22} & \cdots & a_{2n}-b_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1}-b_{m1} & a_{m2}-b_{m2} & \cdots & a_{mn}-b_{mn} \end{bmatrix}.$$

Contoh 2:

Misalkan matriks $\boldsymbol{A}$ dan $\boldsymbol{B}$ berordo 2×2

$$\boldsymbol{A} = \begin{bmatrix} 21 & 8 \\ 2 & 15 \end{bmatrix} \text{ dan } \boldsymbol{B} = \begin{bmatrix} 7 & 4 \\ 5 & 12 \end{bmatrix},$$

maka

$$\boldsymbol{A} - \boldsymbol{B} = \begin{bmatrix} 21 & 8 \\ 2 & 15 \end{bmatrix} - \begin{bmatrix} 7 & 4 \\ 5 & 12 \end{bmatrix} = \begin{bmatrix} 21-7 & 8-4 \\ 2-5 & 15-12 \end{bmatrix} = \begin{bmatrix} 14 & 4 \\ -3 & 3 \end{bmatrix}.$$

2.1.1.3 Perkalian Matriks dengan Skalar

Hasil kali matriks $\boldsymbol{A}$ dengan skalar k yang ditulis $k\boldsymbol{A}$ adalah matriks dari perkalian setiap entri $\boldsymbol{A}$ dengan k (Lipschutz dan Lipson, 2009). Hasil kali matriks $\boldsymbol{A}$ berordo $m \times n$ dengan skalar k dapat ditulis

$$k\boldsymbol{A} = k[a_{ij}] = [ka_{ij}] = \begin{bmatrix} ka_{11} & ka_{12} & \cdots & ka_{1n} \\ ka_{21} & ka_{22} & \cdots & ka_{2n} \\ \vdots & \vdots & & \vdots \\ ka_{m1} & ka_{m2} & \cdots & ka_{mn} \end{bmatrix}.$$

Contoh 3:

Misalkan matriks $\boldsymbol{A}$ berordo 2×2 dan skalar k

$$\boldsymbol{A} = \begin{bmatrix} 21 & 8 \\ 2 & 15 \end{bmatrix} \text{ dan } k = 2,$$

maka

$$k\boldsymbol{A} = 2\begin{bmatrix} 21 & 8 \\ 2 & 15 \end{bmatrix} = \begin{bmatrix} 2 \times 21 & 2 \times 8 \\ 2 \times 2 & 2 \times 15 \end{bmatrix} = \begin{bmatrix} 42 & 16 \\ 4 & 30 \end{bmatrix}.$$

2.1.1.4 Perkalian Matriks

Jika matriks $\boldsymbol{A}$ berordo $m \times r$ dan matriks $\boldsymbol{B}$ berordo $r \times n$, maka hasil kali matriks $\boldsymbol{AB}$ berordo $m \times n$ yang entrinya ditentukan sebagai berikut: untuk mencari entri pada baris i dan kolom j dari $\boldsymbol{AB}$, sendirikan baris i dari matriks $\boldsymbol{A}$ dan kolom j dari matriks $\boldsymbol{B}$. Kalikan entri dari baris dan kolom yang bersesuaian bersama-sama, kemudian jumlahkan hasilnya (Anton dan Rorres,

2010). Misal matriks $\boldsymbol{A}$ berordo $m \times r$ dan matriks $\boldsymbol{B}$ berordo $r \times n$, maka $\boldsymbol{AB}$ dapat ditulis

$$\boldsymbol{A} = [a_{ij}] = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1r} \\ a_{21} & a_{22} & \cdots & a_{2r} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mr} \end{bmatrix}$$

dan

$$\boldsymbol{B} = [b_{ij}] = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & & \vdots \\ b_{r1} & b_{r2} & \cdots & b_{rn} \end{bmatrix}.$$

Entri $(\boldsymbol{AB})_{ij}$ pada baris i dan kolom j dari $\boldsymbol{AB}$ diberikan oleh

$$(\boldsymbol{AB})_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{ir}b_{rj},$$

untuk setiap $i = 1, 2, 3, \cdots, m$ dan $j = 1, 2, 3, \cdots, n$.

Contoh 4:

Misalkan matriks $\boldsymbol{A}$ dan $\boldsymbol{B}$ berordo 2×2

$$\boldsymbol{A} = \begin{bmatrix} 1 & 8 \\ 2 & 15 \end{bmatrix} \text{ dan } \boldsymbol{B} = \begin{bmatrix} 4 & 7 & 0 \\ 6 & 12 & 2 \end{bmatrix},$$

maka

$$\boldsymbol{AB} = \begin{bmatrix} 1 & 8 \\ 2 & 15 \end{bmatrix} \begin{bmatrix} 4 & 7 & 0 \\ 6 & 12 & 2 \end{bmatrix}$$

$$= \begin{bmatrix} (1 \times 4) + (8 \times 6) & (1 \times 7) + (8 \times 12) & (1 \times 0) + (8 \times 2) \\ (2 \times 4) + (15 \times 6) & (2 \times 7) + (15 \times 12) & (2 \times 0) + (15 \times 2) \end{bmatrix}$$

$$= \begin{bmatrix} 52 & 103 & 16 \\ 98 & 194 & 30 \end{bmatrix}.$$

2.1.2 Transpos dan Invers Matriks

2.1.2.1 Transpos Matriks

Jika $\boldsymbol{A}$ adalah sebarang matriks berordo $m \times n$, maka transpos $\boldsymbol{A}$ dinyatakan oleh $\boldsymbol{A}^t$ dan didefinisikan dengan matriks berordo $n \times m$ yang

kolom pertamanya adalah baris pertama dari $\boldsymbol{A}$, kolom keduanya adalah baris kedua dari $\boldsymbol{A}$, demikian dengan kolom ketiga adalah baris ketiga dari $\boldsymbol{A}$, dan seterusnya (Anton dan Rorres, 2010).

Misal matriks $\boldsymbol{A}$ berordo $m \times n$

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix},$$

maka transpos dari matriks $\boldsymbol{A}$ dapat ditulis sebagai berikut:

$$\boldsymbol{A}^t = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \vdots & \vdots & & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{bmatrix}.$$

Contoh 5:

Misalkan matriks $\boldsymbol{A}$ berordo 3×3

$$\boldsymbol{A} = \begin{bmatrix} 3 & 1 & -1 \\ 2 & 5 & 6 \\ 1 & 4 & 8 \end{bmatrix}.$$

Maka transpos $\boldsymbol{A}$ adalah

$$\boldsymbol{A}^t = \begin{bmatrix} 3 & 2 & 1 \\ 1 & 5 & 4 \\ -1 & 6 & 8 \end{bmatrix}.$$

2.1.2.2 Invers Matriks

Jika matriks persegi $\boldsymbol{A}$ dikalikan dengan matriks persegi $\boldsymbol{B}$ yang berordo sama, menghasilkan matriks identitas, yaitu: $\boldsymbol{AB} = \boldsymbol{BA} = \boldsymbol{I}$, maka $\boldsymbol{A}$ merupakan invers dari $\boldsymbol{B}$, atau $\boldsymbol{B}$ merupakan invers dari $\boldsymbol{A}$. Maka notasi yang digunakan adalah $\boldsymbol{B} = \boldsymbol{A}^{-1}$, sehingga $\boldsymbol{AA}^{-1} = \boldsymbol{I}$ (Andrianto dan Prijono, 2006).

Contoh 6:

Misal matriks $\boldsymbol{A}$ dan $\boldsymbol{B}$ berordo 2×2

$$\boldsymbol{A} = \begin{bmatrix} 2 & -5 \\ -1 & 3 \end{bmatrix} \text{ dan } \boldsymbol{B} = \begin{bmatrix} 3 & 5 \\ 1 & 2 \end{bmatrix},$$

maka

$$\boldsymbol{AB} = \begin{bmatrix} 2 & -5 \\ -1 & 3 \end{bmatrix}\begin{bmatrix} 3 & 5 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \boldsymbol{I}$$

$$\boldsymbol{BA} = \begin{bmatrix} 3 & 5 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} 2 & -5 \\ -1 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \boldsymbol{I}.$$

2.1.3 Determinan Matriks

2.1.3.1 Permutasi

Sebelum berbicara mengenai determinan matriks, terlebih dahulu akan dibahas mengenai permutasi.

Definisi 1

Suatu permutasi bilangan bulat $\{1, 2, 3, \cdots, n\}$ merupakan suatu susunan bilangan-bilangan bulat tersebut dalam suatu urutan tertentu tanpa menghilangkan atau mengurangi (Purwanto, dkk, 2005).

Secara umum banyaknya permutasi n bilangan bulat adalah banyaknya cara menyusun bilangan-bilangan tersebut, yaitu $n(n-1)(n-2)\cdots 3 \cdot 2 \cdot 1 = n!$, dapat ditulis $P_n = n!$

Contoh 7:

Terdapat enam permutasi yang berbeda dari himpunan bilangan bulat $\{1, 2, 3\}$, yaitu:

$(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2)$ dan $(3, 2, 1)$.

a. Permutasi Ganjil dan Genap

Untuk membahas mengenai permutasi genap dan ganjil, maka dikenalkan terlebih dahulu tentang inversi. Dikatakan terjadi suatu inversi di dalam permutasi apabila terdapat bilangan yang lebih besar berada di depan bilangan yang lebih

kecil dalam urutan permutasi tersebut.

Definisi 2

Suatu permutasi dikatakan genap jika jumlah inversi seluruhnya adalah genap dan dikatakan ganjil jika jumlah inversi seluruhnya adalah ganjil (Purwanto, dkk, 2005).

Contoh 8:

Misalkan diambil permutasi $(2, 3, 1)$ dan $(1, 3, 2)$ dari himpunan bilangan bulat $\{1, 2, 3\}$. Karena 2 mendahului 1 dan 3 mendahului 1, maka permutasi $(2, 3, 1)$ mempunyai 2 inversi, sehingga permutasinya adalah genap. Karena 3 mendahului 2 maka permutasi $(1, 3, 2)$ hanya mempunyai 1 inversi, sehingga permutasinya adalah ganjil.

b. Hasil Kali Elementer

Hasil kali elementer bertanda dari matriks $\boldsymbol{A}$ yang berordo $n \times n$ adalah perkalian dari elemen-elemen matriks $\boldsymbol{A}$ sebanyak n yang tidak berasal dari baris yang sama maupun dari kolom yang sama (Purwanto, dkk, 2005).

Contoh 9:

Misalkan matriks $\boldsymbol{A}$ berordo 3×3

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

Maka daftar dari semua hasil kali elementer dari matriks $\boldsymbol{A}$ tersebut adalah $a_{11}\, a_{22}\, a_{33}$, $a_{12}\, a_{21}\, a_{33}$, $a_{13}\, a_{21}\, a_{32}$, $a_{11}\, a_{23}\, a_{32}$, $a_{12}\, a_{23}\, a_{31}$, dan $a_{13}\, a_{22}\, a_{31}$. Ada sebanyak $n! = 3! = 6$ permutasi.

Dari contoh di atas diketahui bahwa hasil kali elementer dari matriks $\boldsymbol{A}$ tersebut berbentuk $a_{1\, j_1}, a_{2\, j_2}, \cdots, a_{n\, j_n}$, dengan $(j_1, j_2, \cdots, j_n)$ adalah permutasi

dari himpunan $(1, 2, \cdots, n)$.

c. Hasil Kali Elementer Bertanda

Hasil kali elementer bertanda adalah hasil kali elementer $(a_{1\,j_1}, a_{2\,j_2}, \cdots, a_{n\,j_n})$ yang dikalikan dengan 1 jika $(j_1, j_2, \cdots, j_n)$ merupakan permutasi genap dan dikalikan dengan -1 jika $(j_1, j_2, \cdots, j_n)$ merupakan permutasi ganjil (Purwanto, dkk, 2005).

Contoh 10:

Misalkan matriks $\boldsymbol{A}$ berordo 2×2

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}.$$

Maka hasil kali elementer $(a_{11}\, a_{22})$ dari permutasi $(1, 2)$ mempunyai 0 inversi merupakan permutasi genap, sehingga hasil kali elementer $(a_{11}\, a_{22})$ dikali 1 menjadi $a_{11}\, a_{22}$. Hasil kali elementer $(a_{12}\, a_{21})$ dari permutasi $(2, 1)$ mempunyai 1 inversi merupakan permutasi ganjil, sehingga hasil kali elementer $(a_{12}\, a_{21})$ dikali -1 menjadi $-a_{12}\, a_{21}$.

2.1.3.2 Determinan Matriks

Secara umum determinan untuk sebarang matriks persegi berordo $n \times n$ didefinisikan sebagai berikut:

Definisi 3

Jika $\boldsymbol{A}$ adalah matriks persegi, maka determinan dari matriks $\boldsymbol{A}$ dinotasikan dengan $\det(\boldsymbol{A})$ atau $|\boldsymbol{A}|$ didefinisikan sebagai jumlah semua hasil kali elementer bertanda dari matriks $\boldsymbol{A}$ (Purwanto, dkk, 2005).

Contoh 11:

Misalkan matriks $\boldsymbol{A}$ berordo 2×2

$$\boldsymbol{A} = \begin{bmatrix} 1 & 4 \\ 5 & 2 \end{bmatrix},$$

maka

$$\det(\boldsymbol{A}) = \begin{vmatrix} 1 & 4 \\ 5 & 2 \end{vmatrix} = 1 \cdot 2 - 4 \cdot 5 = -18.$$

2.1.3.3 Minor dan Kofaktor

Sebelum menguraikan bagaimana menghitung determinan khususnya untuk matriks berordo tinggi, perlu didefinisikan dahulu konsep yang mendasari perhitungan yaitu tentang minor dan kofaktor. Minor dan kofaktor didefinisikan sebagai berikut:

Definisi 4

Jika $\boldsymbol{A}$ adalah matriks persegi, maka minor entri a_{ij} dinyatakan oleh M_{ij} dan didefinisikan sebagai determinan submatriks yang tetap setelah baris ke i dan kolom ke j dicoret dari $\boldsymbol{A}$. Bilangan $(-1)^{i+j}M_{ij}$ dinyatakan dengan C_{ij} dan dinamakan kofaktor entri a_{ij} (Anton dan Rorres, 2010).

Contoh 12:

Misalkan matriks $\boldsymbol{A}$ berordo 3×3

$$\boldsymbol{A} = \begin{bmatrix} 3 & 1 & -4 \\ 2 & 5 & 6 \\ 1 & 4 & 8 \end{bmatrix}.$$

Minor entri a_{11} adalah

$$M_{11} = \begin{vmatrix} 5 & 6 \\ 4 & 8 \end{vmatrix} = 16,$$

dan kofaktor entri a_{11} adalah

$$C_{11} = (-1)^{1+1}M_{11} = 16.$$

Demikian pula, minor entri a_{32} adalah

$$M_{32} = \begin{vmatrix} 3 & -4 \\ 2 & 6 \end{vmatrix} = 26,$$

dan kofaktor entri a_{11} adalah

$$C_{32} = (-1)^{3+2} M_{32} = -26.$$

Dari contoh di atas terlihat bahwa perbedaan minor dan kofaktor adalah dari tandanya, yakni $C_{ij} = \pm M_{ij}$. Cara mudah untuk menentukan tanda + atau −, yaitu mengikuti pola pada matriks berikut:

$$\begin{bmatrix} + & - & + & \cdots \\ - & + & - & \cdots \\ + & - & + & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix}.$$

Selain itu elemen dari matriks $\boldsymbol{A}$ dapat diisi dengan kofaktornya, matriks tersebut dinamakan matriks kofaktor dari $\boldsymbol{A}$ (Anton dan Rorres, 2010).

2.1.3.4 Adjoin

Jika matriks kofaktor dari $\boldsymbol{A}$ ditranspos maka hasilnya disebut adjoin $\boldsymbol{A}$.

Definisi 5

Jika $\boldsymbol{A}$ adalah sebarang matriks berordo $n \times n$ dan C_{ij} adalah kofaktor a_{ij}, maka matriks

$$\begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1n} \\ C_{21} & C_{22} & \cdots & C_{2n} \\ \vdots & \vdots & & \vdots \\ C_{n1} & C_{n2} & \cdots & C_{nn} \end{bmatrix}$$

dinamakan matriks kofaktor dari $\boldsymbol{A}$. Transpos matriks ini dinamakan adjoin dari $\boldsymbol{A}$ dan dinyatakan dengan adj($\boldsymbol{A}$) (Anton dan Rorres, 2010).

Contoh 13:

Misalkan matriks $\boldsymbol{A}$ berordo 3×3

$$\boldsymbol{A} = \begin{bmatrix} 3 & 2 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 0 \end{bmatrix}.$$

Kofaktor $\boldsymbol{A}$ adalah $C_{11} = 12, C_{12} = 6, C_{13} = -16, C_{21} = 4, C_{22} = 2, C_{23} = 16,$

$C_{31} = 12$, $C_{32} = -10$, dan $C_{33} = 16$. Matriks kofaktor $\boldsymbol{A}$ adalah

$$\begin{bmatrix} 12 & 6 & -16 \\ 4 & 2 & 16 \\ 12 & -10 & 16 \end{bmatrix},$$

dan adjoin $\boldsymbol{A}$ adalah

$$\text{adj}(\boldsymbol{A}) = \begin{bmatrix} 12 & 4 & 12 \\ 6 & 2 & -10 \\ -16 & 16 & 16 \end{bmatrix}.$$

1.1.3.5 Determinan Matriks dari Ekspansi Kofaktor

Selain mendapatkan determinan dari jumlah perkalian elementer bertanda, determinan dapat diperoleh dari kofaktor.

Definisi 6

Jika $\boldsymbol{A}$ adalah matriks berordo $n \times n$, maka diperoleh bilangan hasil kali antara entri setiap baris atau kolom dengan kofaktor yang bersesuaian dan menambahkan hasil kalinya disebut determinan $\boldsymbol{A}$, dan penjumlahannya disebut ekspansi kofaktor $\boldsymbol{A}$ (Anton dan Rorres, 2010).

Cara menentukan ekspansi kofaktor matriks $\boldsymbol{A}$ berordo $n \times n$ sebagai berikut:

$$\boldsymbol{A} = [a_{ij}] = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix},$$

maka ekspansi kofaktor sepanjang baris i adalah

$$\det(\boldsymbol{A}) = a_{i1}C_{i1} + a_{i2}C_{i2} + a_{i3}C_{i3} + \cdots + a_{in}C_{in},$$

dan ekspansi kofaktor sepanjang kolom j adalah

$$\det(\boldsymbol{A}) = a_{1j}C_{1j} + a_{2j}C_{2j} + a_{3j}C_{3j} + \cdots + a_{nj}C_{nj}.$$

Contoh 14:

Misalkan matriks $\boldsymbol{A}$ berordo 3×3

$$\boldsymbol{A} = \begin{bmatrix} 3 & 1 & 0 \\ -2 & -4 & 3 \\ 5 & 4 & -2 \end{bmatrix}.$$

Maka determinan $\boldsymbol{A}$ dari ekspansi kofaktor sepanjang baris partama adalah

$$\det(\boldsymbol{A}) = \begin{vmatrix} 3 & 1 & 0 \\ -2 & -4 & 3 \\ 5 & 4 & -2 \end{vmatrix}$$

$$= 3C_{11} + 1C_{12} + 0C_{13}$$

$$=3(-1)^{1+1}\begin{vmatrix} -4 & 3 \\ 4 & -2 \end{vmatrix} + 1(-1)^{1+2}\begin{vmatrix} -2 & 3 \\ 5 & -2 \end{vmatrix} +$$

$$0(-1)^{1+}\begin{vmatrix} -2 & -4 \\ 5 & 4 \end{vmatrix}$$

$$= 3\begin{vmatrix} -4 & 3 \\ 4 & -2 \end{vmatrix} - 1\begin{vmatrix} -2 & 3 \\ 5 & -2 \end{vmatrix} + 0\begin{vmatrix} -2 & -4 \\ 5 & 4 \end{vmatrix}$$

$$= 3(-4) - 1(-11) = -1$$

1.1.3.6 Invers Matriks Menggunakan Determinan dan Adjoin

Untuk mencari invers matriks dapat menggunakan determinan dan adjoin sebagai betikut.

Teorema 1

Jika matriks $\boldsymbol{A}$ dapat dibalik jika hanya jika $\det(\boldsymbol{A}) \neq 0$, maka

$$\boldsymbol{A}^{-1} = \frac{1}{\det(\boldsymbol{A})} adj(\boldsymbol{A})$$

Bukti

Misalkan $\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$ dengan $\det(\boldsymbol{A}) \neq 0$

$\boldsymbol{A}\text{adj}(\boldsymbol{A}) = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}\begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1n} \\ C_{21} & C_{22} & \cdots & C_{2n} \\ \vdots & \vdots & & \vdots \\ C_{m1} & C_{m2} & \cdots & C_{mn} \end{bmatrix}$, hasil kali matriks $\boldsymbol{A}$ dengan $\text{adj}(\boldsymbol{A})$ yaitu, baris pertama kolom pertama dari hasil

kali adalah $a_{11}C_{11} + a_{12}C_{12} + a_{13}C_{13} + \cdots + a_{1n}C_{1n}$, baris pertama kolom kedua dari hasil kali adalah $a_{11}C_{21} + a_{12}C_{22} + a_{13}C_{23} + \cdots + a_{1n}C_{2n}$, dan seterusnya. Secara umum hasil kali matriks $\boldsymbol{A}$ dengan adj$(\boldsymbol{A})$ baris ke i kolom ke j adalah $a_{i1}C_{j1} + a_{i2}C_{j2} + a_{i3}C_{j3} + \cdots + a_{in}C_{jn}$. Ambil hasil kali pada diagonal utama yaitu $i = j$, maka diperoleh $a_{i1}C_{j1} + a_{i2}C_{j2} + a_{i3}C_{j3} + \cdots + a_{in}C_{jn} = \det(\boldsymbol{A})$. Sebaliknya hasil kali selain pada diagonal utama yaitu $i \neq j$, maka entri-entri a dan kofaktor-kofaktornya berasal dari baris-baris matriks $\boldsymbol{A}$ yang berbeda, sehingga hasilnya adalah 0. Diperoleh hasil kali matriks $\boldsymbol{A}$ dengan adj$(\boldsymbol{A})$ yaitu,

$$\boldsymbol{A}\text{adj}(\boldsymbol{A}) = \begin{bmatrix} |\boldsymbol{A}| & 0 & \cdots & 0 \\ 0 & |\boldsymbol{A}| & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & |\boldsymbol{A}| \end{bmatrix} = |\boldsymbol{A}| \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} = \det(\boldsymbol{A})\boldsymbol{I} \quad .$$

Diperoleh $\boldsymbol{A}\text{adj}(\boldsymbol{A}) = \det(\boldsymbol{A})\boldsymbol{I}$ atau $\frac{\boldsymbol{A}\text{adj}(\boldsymbol{A})}{\det(\boldsymbol{A})} = \boldsymbol{I}$, kemudian dikalikan dengan $\boldsymbol{A^{-1}}$ menjadi $\boldsymbol{A^{-1}}\left(\frac{\boldsymbol{A}\text{adj}(\boldsymbol{A})}{\det(\boldsymbol{A})}\right) = \boldsymbol{A^{-1}I}$ atau $\boldsymbol{AA^{-1}}\ \frac{1}{\det(\boldsymbol{A})}\text{adj}(\boldsymbol{A}) = \boldsymbol{A^{-1}I}$, karena $\boldsymbol{AA^{-1}} = \boldsymbol{I}$ dan $\boldsymbol{A^{-1}I} = \boldsymbol{A^{-1}}$ maka diperoleh $\frac{1}{\det(\boldsymbol{A})}\text{adj}(\boldsymbol{A}) = \boldsymbol{A^{-1}}$ atau $\boldsymbol{A^{-1}} = \frac{1}{\det(\boldsymbol{A})}\text{adj}(\boldsymbol{A})$.

2.2 Keterbagian

Teori bilangan merupakan teori yang mendasar dalam memahami algoritma kriptografi. Teori ini berkaitan dengan sifat-sifat dari bilangan bulat (*integer*). Salah satu yang menjadi topik utama dalam teori bilangan adalah keterbagian. Beberapa sifat dan relasi lain seperti kekongruenan dikembangkan dari masalah keterbagian.

Definisi 7

Misal a dan b adalah bilangan bulat dengan $a \neq 0$. Dikatakan a membagi b jika terdapat bilangan bulat c sedemikian sehingga $b = ac$, dinotasikan dengan $a|b$. Ketika a membagi b dikatakan a adalah faktor atau pembagi dari b, dan b adalah kelipatan dari a (Rosen, 2012).

Contoh 15:

Misal $a = 3$ dan $b = 18$

maka

$3|18$, karena ada $6 \in \mathbb{Z}$ sehingga $18 = 3 \cdot 6$.

2.2.1 Aritmetika Modular

Aritmetika modular sangat berperan dalam kriptografi karena banyak digunakan dalam algoritma enkripsi, baik algoritma enkripsi simetri maupun asimetri. Dalam aritmetika modular, konsep faktor persekutuan terbesar (FPB) antara lain digunakan untuk operasi invers. Selain FPB konsep lain seperti kongruensi modulo sangat penting dalam kriptografi (Kromodimoeljo, 2010).

2.2.1.1 Faktor Persekutuan Terbesar (FPB)

Jika $a, b \in \mathbb{Z}$ yang tidak keduanya 0, maka faktor persekutuan terbesar (FPB) dari a dan b adalah bilangan asli g sedemikian sehingga $g|a, g|b$, dan g adalah pembagi dari setiap faktor persekutuan dari a dan b (Mollin, 2007). Untuk selanjutnya notasi faktor persekutuan terbesar dari a dan b ditulis dengan $g = (a, b)$ dengan $a, b \in \mathbb{Z}$ dan yang tidak keduanya 0.

Contoh 16:

Himpunan semua faktor dari 16 adalah:

$A = \{-16, -8, -4, -2, -1, 1, 2, 4, 8, 16\}$,

dan himpunan semua faktor dari 18 adalah:

$B = \{-18, -9, -6, -3, -2, -1, 1, 2, 3, 6, 9, 18\}$.

Himpunan semua faktor persekutuan 16 dan 18 adalah:

$G = \{-2, -1, 1, 2\}$.

Karena unsur G yang terbesar adalah 2, maka $(16, 18) = 2$.

2.2.1.2 Relatif Prima

Bilangan bulat a dan b dikatakan relatif prima jika $(a, b) = 1$. Jika a dan b relatif prima, maka terdapat bilangan bulat m dan n sedemikian sehingga $ma + nb = 1$ (Ariyus, 2008).

Contoh 17:

Himpunan semua faktor dari 10 adalah:

$A = \{-10, -5, -2, -1, 1, 2, 5, 10\}$,

dan himpunan semua faktor dari 21 adalah:

$B = \{-21, -7, -3, -1, 1, 3, 7, 21\}$.

Himpunan semua faktor persekutuan dari 10 dan 21 adalah:

$G = \{-1, 1\}$.

Karena unsur G yang terbesar adalah 1, maka $(10, 21) = 1$. Jadi 10 dan 21 relatif prima. Selain itu diperoleh $-2 \cdot 10 + 1 \cdot 21 = 1$ dengan $m = -2$ dan $n = 1$.

2.2.1.3 Kongruensi Modulo

Jika bilangan bulat M yang tidak nol, membagi selisih $a - b$, maka dikatakan a kongruen dengan b modulo M, dan dapat ditulis $a \equiv b \pmod{M}$ (Irawan, dkk, 2014).

Contoh 18:

$8 \equiv 4 \pmod 2$ karena $2|(8 - 4)$ atau $2|4$

2.2.1.4 Invers Modulo

Teorema 2

Bilangan bulat a mempunyai invers modulo M jika dan hanya jika $(a, M) = 1$ (Ariyus, 2008).

Bukti

Jika $(a, M) = 1$ maka terdapat bilangan m dan n sedemikan hingga $ma + nM = 1$ yang memiliki arti bahwa $ma + nM \equiv 1 \pmod{M}$. Karena $nM = 0$ maka $ma \equiv 1 \pmod{M}$ yang berarti bahwa m adalah invers dari a modulo M.

2.2.2 Kongruensi Matriks

Jika $\boldsymbol{A}$ dan $\boldsymbol{B}$ adalah matriks $r \times n$ dengan entri-entrinya bilangan bulat, unsur ke (i, j) berturut-turut adalah a_{ij} dan b_{ij}. $\boldsymbol{A}$ dikatakan kongruensi dengan $\boldsymbol{B}$ modulo m, jika $a_{ij} \equiv b_{ij} \pmod{m}$ untuk setiap pasang (i, j) dengan $1 \leq i \leq r$ dan $1 \leq j \leq n$ dan dinotasikan dengan $\boldsymbol{A} \equiv \boldsymbol{B} \pmod{m}$ (Irawan, dkk, 2014).

Contoh 19:

$$\begin{bmatrix} 15 & 3 \\ 8 & 12 \end{bmatrix} \equiv \begin{bmatrix} 4 & 3 \\ -3 & 1 \end{bmatrix} \pmod{11}.$$

2.2.2.1 Invers Matriks Modulo

Jika $\boldsymbol{A}$ dan $\boldsymbol{B}$ adalah matriks berordo $n \times n$ dari bilangan-bilangan bulat, dan $\boldsymbol{AB} \pmod{m} \equiv \boldsymbol{BA} \pmod{m} \equiv \boldsymbol{I} \pmod{m}$ dengan $\boldsymbol{I}$ adalah matriks identitas berordo n, maka $\boldsymbol{B}$ dikatakan invers dari $\boldsymbol{A}$ modulo m (Irawan, dkk, 2014).

Contoh 20:

$$\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} 3 & 4 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 6 & 10 \\ 10 & 6 \end{bmatrix} \equiv \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \pmod{5}.$$

Dari sini terlihat bahwa $\begin{bmatrix} 3 & 4 \\ 1 & 2 \end{bmatrix}$ adalah invers dari $\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$ modulo 5.

2.2.2.2 Invers Matriks Modulo dari Adjoin

Untuk mencari invers matriks modulo berordo tinggi, perlu adjoin untuk mendapatkannya. Berikut teorema yang menggunakan adjoin untuk mencari invers matriks modulo berordo tinggi, yaitu:

Teorema 3

Jika $\boldsymbol{A}$ adalah matriks berordo $n \times n$ dengan unsur-unsurnya bilangan bulat dan m adalah bilangan bulat positif, sedemikian sehingga $(\det(\boldsymbol{A}), m) = 1$ dan $\det(\boldsymbol{A})^{-1}$ adalah invers dari $\det(\boldsymbol{A})$ modulo m maka invers dari $\boldsymbol{A}$ modulo m adalah $\boldsymbol{A^{-1}} = \det(\boldsymbol{A})^{-1}\,\mathrm{adj}(\boldsymbol{A})$ (Irawan, dkk, 2014).

Bukti

Jika $(\det(\boldsymbol{A}), m) = 1$ maka $\det(\boldsymbol{A}), \neq 0$ dan $\boldsymbol{A}\mathrm{adj}(\boldsymbol{A}) = \det(\boldsymbol{A})\,\boldsymbol{I}$. Karena $(\det(\boldsymbol{A}), m) = 1$, maka $\det(\boldsymbol{A})$ mempunyai invers $\det(\boldsymbol{A})^{-1}$ modulo m. Misal $\boldsymbol{A^{-1}} = \det(\boldsymbol{A})^{-1}\,\mathrm{adj}(\boldsymbol{A})$ maka $\boldsymbol{A}\det(\boldsymbol{A})^{-1}\,\mathrm{adj}(\boldsymbol{A}) = \boldsymbol{A}\mathrm{adj}(\boldsymbol{A}) \cdot \det(\boldsymbol{A})^{-1} = \det(\boldsymbol{A})\,\boldsymbol{I}\,\det(\boldsymbol{A})^{-1}$ atau $\det(\boldsymbol{A})\,\det(\boldsymbol{A})^{-1}\,\boldsymbol{I} \equiv \boldsymbol{I} \pmod{m}$ dan $\det(\boldsymbol{A})^{-1}\,\mathrm{adj}(\boldsymbol{A})\boldsymbol{A} = \det(\boldsymbol{A})^{-1}\,\boldsymbol{A}\mathrm{adj}(\boldsymbol{A}) = \det(\boldsymbol{A})^{-1}\,\det(\boldsymbol{A})$ atau $\det(\boldsymbol{A})\,\boldsymbol{I} \cdot \det(\boldsymbol{A})^{-1} \equiv \boldsymbol{I} \pmod{m}$ Ini menunjukkan bahwa $\boldsymbol{A^{-1}} = \det(\boldsymbol{A})^{-1}\,\mathrm{adj}(\boldsymbol{A})$ adalah invers dari $\boldsymbol{A}$ modulo m.

2.3 Kriptografi

Kata kriptografi berasal dari bahasa Yunani. Dalam bahasa Yunani kriptografi terdiri dari dua buah kata yaitu *cryptos* yang berarti rahasia dan *graphia* yang berarti tulisan. Berarti secara umum makna dari kriptografi adalah ilmu yang mempelajari tentang bagaimana menjaga kerahasiaan suatu pesan,

supaya isi pesan yang disampaikan tersebut aman sampai ke penerima pesan (Ariyus, 2008).

2.3.2 Enkripsi dan Dekripsi

Enkripsi merupakan hal yang sangat penting dalam kriptografi, merupakan pengamanan data yang dikirim supaya terjaga kerahasiaannya. Enkripsi adalah proses pengacakkan pesan asli (*plaintext*) menjadi pesan acak (*ciphertext*) yang sulit dibaca oleh orang yang tidak mempunyai kunci dekripsi. Dekripsi adalah kebalikan dari enkripsi yaitu mengembalikan *ciphertext* menjadi *plaintext*. Algoritma yang digunakan untuk dekripsi tentu berbeda dengan algoritma yang digunakan untuk enkripsi (Ariyus, 2008). Proses enkripsi dikatakan aman apabila menghasilkan *ciphertext* yang membutuhkan waktu lama (misalnya seribu tahun) untuk didekripsikan oleh orang yang tidak mempunyai kunci dekripsi atau kriptaanalis (Kromodimoeljo, 2010).

2.3.3 Algoritma Kriptografi

Algoritma merupakan urutan atau langkah-langkah untuk menyelesaikan masalah yang disusun secara sistematis. Algoritma kriptografi adalah langkah-langkah bagaimana cara menyembunyikan pesan dari orang-orang yang tidak berhak menerima pesan tersebut (Ariyus, 2008).

Algoritma kriptografi dibagi menjadi tiga bagian berdasarkan kunci yang dipakainya:

1. Algoritma simetri (menggunakan satu kunci untuk enkripsi dan dekripsi).
2. Algoritma asimetri (menggunakan kunci yang berbeda untuk enkripsi dan dekripsi).
3. *Hash function* (Ariyus, 2008).

2.3.3.1 Algoritma Simetri

Algoritma simetri disebut algoritma kunci rahasia. Merupakan algoritma kriptografi yang menggunakan kunci yang sama untuk proses enkripsi dan dekripsinya. Karena hanya menggunakan kunci yang sama maka si pengirim dan penerima pesan harus menjaga kerahasiaan kunci tersebut. Apabila kunci jatuh ke tangan orang lain, maka orang tersebut dapat mengenkripsi dan mendekripsikan pesan. Untuk menjaga keamanannya, maka setiap melakukan enkripsi dan dekripsi kuncinya harus sering diubah (Ariyus, 2008).

Terdapat beberapa algoritma yang dikembangkan oleh ahli-ahli kriptografi menggunakan algoritma simetri, antara lain (a) *Data Encryptios Standart* (DES), (b) *Advance Encryptios Standart* (AES), (c) *Affine cipher*, dan (d) Rivest *Code* 4 (RC4) (Ariyus, 2008).

2.3.3.2 Algoritma Asimetri

Algoritma asimetri disebut algoritma kunci publik. Algoritma asimetri memiliki dua kunci yang berbeda untuk proses enkripsi dan dekripsinya, yang kunci enkripsi dapat diketahui oleh publik, tetapi untuk kunci dekripsinya hanya dimiliki penerima pesan, sehingga siapa saja dapat mengenkripsi pesan menggunakan kunci publik tetapi tidak dapat mendekripsikannya. Hanya orang yang memiliki kunci privat yang dapat mendekripsikan pesan tersebut. Dengan demikian algoritma asimetri lebih aman dibandingkan dengan algoritma simetri (Kromodimoeljo, 2010).

Terdapat beberapa algoritma yang dikembangkan oleh ahli-ahli kriptografi menggunakan algoritma asimetri, antara lain (a) Rivest Shamir Adleman (RSA), (b) Elgamal, (c) Knapsack, dan (d) Lucas (LUC) (Ariyus, 2008).

2.3.3.3 Fungsi *Hash*

Fungsi *hash* sering disebut dengan fungsi satu arah (*one-way function*), *message digest*, *fingerprint*, fungsi kompresi dan *message authentication code* (MAC), merupakan suatu fungsi matematika yang mengambil masukan panjang variabel dan mengubahnya ke dalam urutan biner dengan panjang yang tepat. Fungsi *hash* biasanya diperlukan bila ingin membuat sidik jari dari suatu pesan. Sidik jari pada pesan merupakan suatu tanda yang menandakan bahwa pesan tersebut benar-benar berasal dari orang yang diinginkan (Ariyus, 2008).

2.3.3.4 Algoritma Hibrida

Algoritma hibrida adalah algoritma yang memanfaatkan dua tingkatan kunci, yaitu kunci rahasia (simetri) yang disebut kunci sesi untuk enkripsi dan pasangan kunci rahasia atau kunci publik (asimetri) untuk pemberian tanda tangan digital serta melindungi kunci simetri (Ariyus, 2008). Dalam prosesnya pesan dienkripsi menggunakan kunci sesi sedangkan kunci sesi dienkripsi menggunakan kunci publik. Proses tersebut bertujuan karena tidak hanya pesan yang dikirim tetapi dengan kunci sesinya sehingga perlu diamankan. Karena enkripsi dilakukan pada pesan dan kunci maka dikatakan memiliki dua tingkatan kunci.

2.3.4 Algoritma RSA

Algoritma RSA adalah algoritma yang melakukan pemfaktoran bilangan yang sangat besar. Oleh karena alasan tersebut RSA dianggap aman. Untuk membangkitkan dua kunci, dipilih dua bilangan prima acak yang besar. Algoritma untuk membangkitkan pasangan kunci RSA adalah:

1. Pilih dua bilangan prima sebarang, p dan q.
2. Hitung $n = pq$ (sebaiknya $p \neq q$ sebab jika $p = q$ maka $n = p^2$ sehingga p

dapat diperoleh dengan menarik akar pangkat dua dari (n).

3. Hitung $\phi(n) = (p-1)(q-1)$.
4. Pilih kunci publik, e yang relatif prima terhadap $\phi(n)$. Didapatkan $\big(e, \phi(n)\big) = 1$.
5. Bangkitkan kunci privat dengan menggunakan $ed \equiv 1\big(\mathrm{mod}\ \phi(n)\big)$. Perhatikan bahwa $ed \equiv 1(\mathrm{mod}\ \phi(n))$ ekivalen dengan $ed = 1 + k\phi(n)$, d dapat dihitung dengan $d = \frac{1+k\,\phi(n)}{e}$.

Akan terdapat bilangan bulat k yang memberikan bilangan bulat d. Hasil dari algoritma di atas:

- Kunci publik adalah pasangan e dan n.
- Kunci privat adalah pasangan d dan n (Munir, 2004).

Untuk proses enkripsi diperoleh dari pasangan kunci publik dengan *plaintext* dan menghasilkan *ciphertext*, yaitu

$$C \equiv P^e \ (\mathrm{mod}\ n).$$

Untuk proses dekripsi diperoleh dari pasangan kunci privat dengan *ciphertext* dan menghasilkan *plaintext*, yaitu

$$P \equiv C^e \ (\mathrm{mod}\ n).$$

2.3.5 Algoritma *Affine Cipher*

Affine cipher termasuk *monoalphabetic substitution cipher* yang setiap huruf-huruf alfabetnya dapat diubah ke dalam angka-angka, kemudian dienkripsi dengan *affine transformation* dengan rumus

$$C \equiv aP + b \ (\mathrm{mod}\ n),$$

dan untuk dekripsi dengan rumus

$$P \equiv a^{-1}(C - b) \ (\mathrm{mod}\ n).$$

Jadi kunci untuk enkripsi algoritma *affine cipher* terdiri dari dua parameter a dan b. Supaya a mempunyai invers (a^{-1}), maka a harus memenuhi $(a, n) = 1$ (Kromodimoeljo, 2010).

2.4 Kajian Islam Mengenai Pesan dan Adil

Di dalam al-Quran dijelaskan bahwasanya Allah Swt memerintahkan hamba-Nya untuk menyampaikan pesan atau amanat hanya kepada yang berhak saja dan menetapkan hukum dengan adil, yaitu terdapat di dalam al-Quran surat an-Nisaa'/4:58:

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 ٥٨

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

Di dalam tafsir Ibnu Katsir (2003) Allah Swt mengabarkan bahwa Dia memerintahkan menunaikan amanat kepada ahlinya. Di dalam hadits al-Hasan dari Samurah, bahwa Rasulullah Saw telah bersabda:

أَدِّالأَمَانَةَ إِلَى مَنِ ائْتَمَنَكَ ،وَلاَتَخُنْ مَنْ خَا نَكَ

"Tunaikanlah amanah kepada yang memberikan amanah dan jangan khianati orang yang berkhianat kepadamu." (HR. Ahmad dan Ahlus Sunah)

Hal itu mencakup seluruh amanah yang wajib bagi manusia, berupa hak-hak Allah Swt terhadap para hamba-Nya, seperti shalat, zakat, puasa, kafarat, nadzar, dan selain dari itu, yang kesemuanya adalah amanah yang diberikan tanpa pengawasan hamba-Nya yang lain. Serta amanah berupa hak-hak sebagian hamba dengan hamba lainnya, seperti titipan dan seterusnya, yang kesemuanya adalah

amanah yang dilakukan tanpa pengawasan saksi. Itulah yang diperintahkan oleh Allah Swt untuk dikerjakan. Barangsiapa yang tidak melakukannya di dunia, maka akan diminta pertanggungjawaban di hari kiamat, sebagaimana yang terdapat di dalam hadits shahih bahwasanya Rasulullah Saw bersabda:

"لَتُؤَدَّنَّ الْحُقُوقُ إِلَى أَهْلِهَا، حَتَّى يُقْتَصَّ لِلشَّاةِ الْجَمَّاءِ مِنَ الْقَرْنَاءِ"

"*Sungguh kamu akan tunaikan kepada ahlinya, hingga akan diqishas untuk (pembalasan) seekor kambing tidak bertanduk terhadap kambing yang bertanduk*".

Ibnu Jarir meriwayatkan dari Ibnu Juraij, ia berkata bahwa ayat ini diturunkan berkenaan dengan 'Utsman bin Thalha di saat Rasulullah Saw mengambil kunci Ka'bah darinya, lalu beliau masuk ke dalam Baitullah pada Fathu Makkah. Di saat beliau keluar, beliau membaca ayat ini, "*Sesungguhnya Allah menyuruh kamu menyampaikan amanat kepada yang berhak menerimanya,*" lalu beliau memanggil 'Utsman dan menyerahkan kunci itu kembali.

Di antara yang masyhur dalam masalah ini adalah bahwa ayat ini turun berkenaan dengan peristiwa tersebut atau tidak, yang pasti hukumnya tetap berlaku umum. Untuk itu Ibnu Abbas dan Muhammad bin al-Hanifiyah berkata: "*Hukumnya untuk orang yang baik dan yang zalim. Yaitu perintah untuk setiap orang.*"

Firman Allah Swt:

وَإِذا حَكَمْتُمْ بَيْنَ النَّاسِ أَنْ تَحْكُمُوا بِالْعَدْلِ

"dan (menyuruh kalian) apabila menetapkan hukum di antara manusia supaya kalian menetapkan dengan adil" (QS. An-Nisaa'/4:58).

Hal ini merupakan perintah Allah Swt yang menganjurkan menetapkan hukum di antara manusia dengan adil. Karena itulah maka Muhammad ibnu Ka'b, Zaid ibnu

Aslam, dan Syahr ibnu Hausyab mengatakan bahwa ayat ini diturunkan hanya berkenaan dengan para umara, yakni para penguasa yang memutuskan perkara di antara manusia (Katsir, 2003).

BAB III

PEMBAHASAN

3.1 Analisis Keamanan Enkripsi dan Dekripsi Algoritma RSA dan *Affine Cipher* dengan Metode Matriks

3.1.1 Analisis Algoritma RSA dan *Affine Cipher* dengan Metode Matriks

Algoritma RSA didasarkan pada dua bilangan prima besar yang berbeda. Dari hasil perkalian dua bilangan tersebut kemudian digunakan sebagai salah satu parameter untuk proses enkripsi dan dekripsi. Dalam algoritma RSA terdapat beberapa parameter yang digunakan untuk proses enkripsi dan dekripsi adalah sebagai berikut:

1. p dan q merupakan dua bilangan prima besar yang dipilih oleh penerima. Sebaiknya dalam memilih p dan q adalah $p \neq q$, sebab apabila $p = q$ maka $n = p^2$ sehingga p dapat diperoleh dengan menarik akar pangkat dua dari (n). Kedua bilangan tersebut bersifat rahasia, sehingga hanya penerima yang mengetahui bilangan tersebut.
2. $n = pq$, dengan n adalah salah satu parameter untuk proses enkripsi dan dekripsi dan sifatnya tidak rahasia dapat disebarluaskan.
3. $\phi(n) = (p-1)(q-1)$, $\phi(n)$ digunakan untuk mendapatkan kunci publik dan kunci privat dan sifatnya rahasia hanya penerima yang dapat mengetahui.
4. e (salah satu kunci publik), dengan $\left(e, \phi(n)\right) = 1$. Karena sebagai salah satu kunci publik maka dapat disebarluaskan.
5. d (salah satu kunci privat), dengan $d = \frac{1+k\,\phi(n)}{e}$. Karena sebagai salah satu kunci privat maka sifatnya rahasia hanya penerima yang dapat mengetahui.

6. Pasangan kunci publik adalah e dan n sedangkan pasangan kunci privat adalah d dan n.
7. P (*plaintext*) merupakan pesan asli yang hanya diketahui oleh pengirim dan penerima.
8. C (*ciphertext*) merupakan pesan acak yang sudah dienkripsi oleh pengirim dan sifatnya rahasia (Munir, 2004).

Dari beberapa parameter tersebut diperoleh bahwa ada parameter yang hanya diketahui oleh pengirim dan penerima pesan yaitu $\phi(n)$ dan d sedangkan yang dapat disebar luaskan yaitu n dan e.

Algoritma *affine cipher* mempunyai dua parameter sebagai kunci. Kedua parameter inilah yang nantinya digunakan dalam proses enkripsi dan dekripsi. Dalam algoritma *affine cipher* terdapat beberapa parameter yang digunakan untuk proses enkripsi dan dekripsi yaitu:

1. a adalah parameter yang digunakan sebagai pengali dengan pesan yang dalam penelitian ini sudah diubah ke dalam bentuk matriks. Untuk memperoleh pesan kembali pada proses dekripsi maka a harus mempunyai invers dan supaya mempunyai invers maka $(\det(a), m) = 1$. Karena sebagai parameter pada proses enkripsi dan dekripsi maka sifatnya rahasia hanya diketahui oleh pengirim dan penerima saja.
2. b adalah parameter yang digunakan sebagai penjumlah dengan hasil kali a dan pesan untuk menghasilkan *ciphertext*, sedangkan sebagai pengurang untuk menghasilkan *plaintext*. Karena sebagai parameter pada proses enkripsi dan dekripsi maka sifatnya rahasia hanya diketahui oleh pengirim dan penerima saja.

3. P (*plaintext*) merupakan pesan asli yang hanya diketahui oleh pengirim dan akan diketahui oleh penerima dengan proses dekripsi.
4. C (*ciphertext*) merupakan pesan acak yang sudah dienkripsi oleh pengirim.

Karena termasuk algoritma simetri maka semua parameter yang digunakan bersifat rahasia hanya diketahui oleh pengirim dan penerima saja.

3.1.2 Keamanan Algoritma RSA

Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan bulat menjadi faktor-faktor primanya. Pada RSA kunci enkripsi disebarkan secara bebas, sehingga semua orang mengetahui kuncinya yaitu e dan n. Nilai n inilah yang akan digunakan untuk mencari faktor-faktor primanya, karena nilai $n = pq$ dengan p dan q adalah bilangan prima. Jika berhasil menemukan nilai p dan q maka dapat mencari kunci dekripsinya d dan n, yaitu $\phi(n) = (p-1)(q-1)$ dan $ed \equiv 1(\text{mod } \phi(n))$ yang ekivalen dengan $ed = 1 + k\phi(n)$, sehingga d dapat dihitung dengan $d = \frac{1+k\,\phi(n)}{e}$ dengan k bilangan bulat. Untuk mendapatkan keamanan yang kuat maka sebaiknya ambil bilangan prima besar dan mempunyai digit yang berbeda. Semakin besar digitnya maka keamanannya semakin kuat karena sulitnya memfaktorkan bilangan bulat besar untuk mendapatkan faktor primanya. Butuh puluhan bahkan jutaan tahun sekalipun menggunakan komputer tercanggih karena belum adanya algoritma pemfaktoran yang mumpuni (Munir, 2004).

3.1.3 Keamanan Algoritma *Affine Cipher*

Keamanan algoritma *affine cipher* terletak pada dua parameter bilangan bulat a dan b, kedua parameter dirahasiakan. Kunci a harus relatif prima dengan m, yaitu 255 (jumlah kode ASCII). Jika $1 < a \leq n$ maka terdapat 200 (bilangan

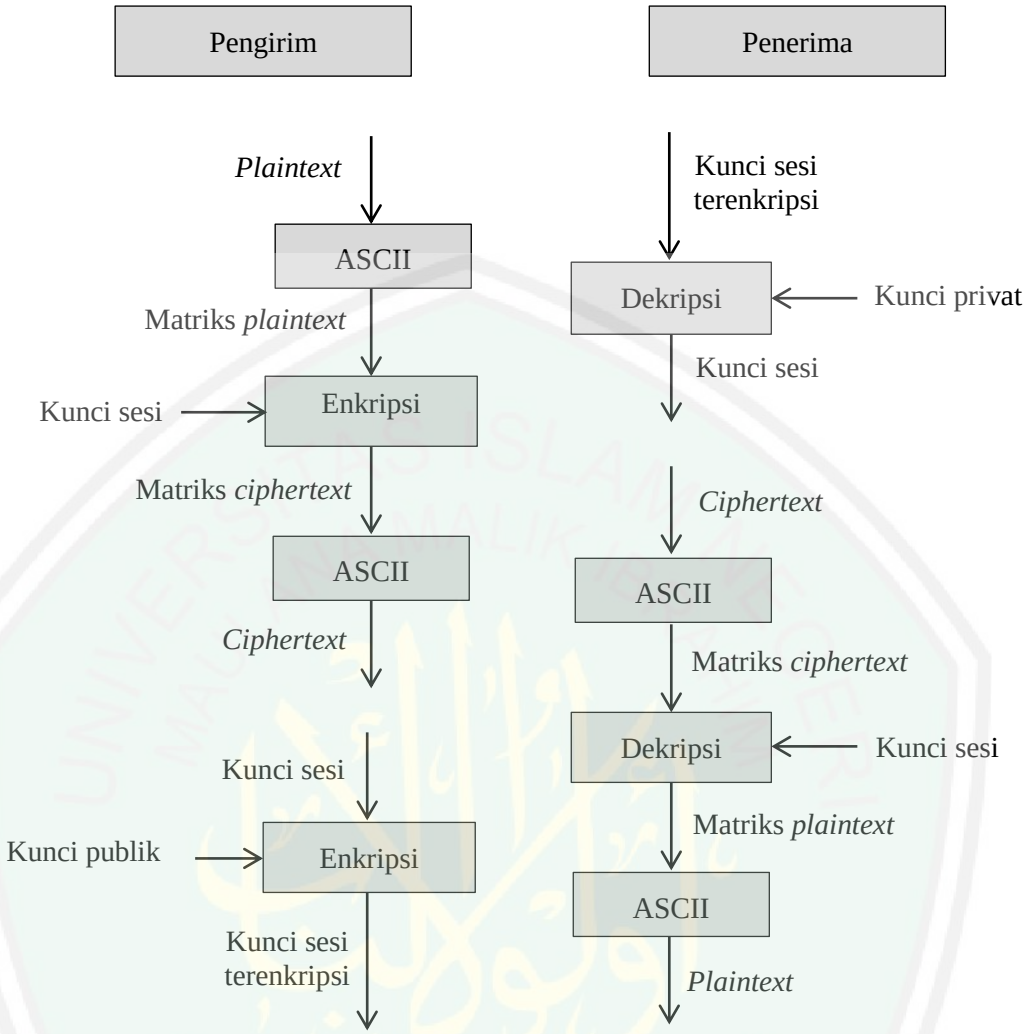
yang relatif prima dengan n) kemungkinan kunci yang dapat digunakan dan jika $1 < b \leq n$ maka terdapat 255 kemungkinan kunci yang dapat digunakan. Kombinasi kedua kunci tersebut yaitu $200 \cdot 255 = 5100$ pasangan kunci a dan b yang dapat digunakan. Untuk memecahkan kuncinya maka harus mendapatkan *ciphertext* dahulu, kemudian menggunakan minimal dua persamaan yang kunci dekripsinya dari 5100 pasangan kunci tersebut.

3.1.4 Konstruksi Algoritma RSA dan *Affine Cipher* dengan Metode Matriks

Untuk melakukan proses enkripsi dan dekripsi perlu dibuat kunci dahulu kemudian dilanjutkan dengan proses enkripsi dan dekripsi. Supaya mudah memahami dibuatkan ilustrasi sebagai berikut:

1. A membangkitkan kunci publik dan kunci privat dari algoritma RSA. Setelah diperoleh kunci publik, maka A mengirimkan kunci tersebut kepada B.
2. B membangkitkan kunci sesi dari algoritma *affine cipher*. Kemudian B membuat *plaintext P* yang sudah diubah ke dalam bentuk desimal sesuai tabel ASCII dan dibentuk ke dalam matriks, B mengenkripsi pesan menggunakan kunci sesi dengan metode matriks dan mengenkripsi kunci sesi menggunakan kunci publik. Setelah diperoleh pesan dan kunci sesi terenkripsi, kemudian dikirim kepada A.
3. A mendekripsikan kunci sesi terenkripsi dari B menggunakan kunci privat.
4. Setelah diperoleh kunci sesi dari dekripsi tersebut, B mendekripsikan pesan terenkripsi menggunakan kunci sesi dengan metode matriks. Hasil akhir diperoleh pesan dari B kepada A.

Secara sederhana proses enkripsi dan dekripsi algoritma RSA dan *affine cipher* dengan metode matriks dapat digambarkan sebagai berikut:



Gambar 3.1 Enkripsi dan Dekripsi Algoritma RSA dan *Affine Cipher* dengan Metode Matriks

Secara matematis dijelaskan sebagai berikut:

1. A memilih bilangan prima besar p dan q (p tidak boleh sama dengan q).
2. A menghitung n dan $\phi(n)$ dengan $n = pq$ dan $\phi(n) = (p-1)(q-1)$.
3. A mengambil bilangan bulat e sedemikian sehingga relatif prima dengan $\phi(n)$ atau $\left(e, \phi(n)\right) = 1$.
4. A menghitung d dengan $d = \frac{1+k\,\phi(n)}{e}$. Terdapat bilangan bulat k yang memberikan bilangan bulat d.
5. A mendapatkan kunci publik e dan n dan kunci privat d dan n. Kemudian A mengirim kunci publik e dan n kepada B.

6. B membuat kunci sesi dalam bentuk matriks persegi $\boldsymbol{A}_{i\times i}$ dan $\boldsymbol{B}_{i\times 1}$ berordo 2×2 sampai 4×4, dengan $\boldsymbol{A}_{i\times i}$ harus mempunyai invers atau memenuhi $(\det(\boldsymbol{A}_{i\times i}), m) = 1 \pmod m$ (dengan m adalah ukuran alfabet, dalam penelitian ini menggunakan kode ASCII).
7. B membuat *plaintext P* yang sudah diubah ke dalam bentuk desimal sesuai tabel ASCII dan dibentuk ke dalam matriks.
8. B mengenkripsi *plaintext P* untuk mendapatkan *ciphertext C* menggunakan kunci sesi dengan metode matriks, yaitu $C \equiv \boldsymbol{A}_{i\times i}P + \boldsymbol{B}_{i\times 1} \pmod m$ (proses perhitungan berbantuan dengan program MATLAB).
9. B mengenkripsi kunci sesi (a dan b) supaya aman jika dikirim kepada A menggunakan kunci publik e dan n, yaitu $a_c \equiv a^e \pmod n$ dan $b_c \equiv b^e \pmod n$ (proses perhitungan berbantuan dengan program MATLAB). Diperoleh kunci sesi terenkripsi, yaitu a_c dan b_c. B mengirimkan *ciphertext* C dan kunci sesi terenkripsi a_c dan b_c kepada A.
10. Setelah mendapatkan *ciphertext* C dan kunci sesi terenkripsi a_c dan b_c, lalu A melakukan proses dekripsi.
11. A mendekripsi kunci sesi terenkripsi a_c dan b_c untuk mendapatkan kunci sesi yang semula (a dan b) menggunakan kunci privat d dan n, yaitu $a \equiv {a_c}^d \pmod n$ dan $b \equiv {b_c}^d \pmod n$ (proses perhitungan berbantuan dengan program MATLAB). Diperoleh kunci sesi semula, yaitu a dan b.
12. Sebelum melakukan dekripsi, A mencari nilai invers dari $\boldsymbol{A}_{i\times i} \pmod m$ atau ${\boldsymbol{A}_{i\times i}}^{-1} \pmod m$ supaya dapat melakukan dekripsi. Setelah mendapatkan ${\boldsymbol{A}_{i\times i}}^{-1} \pmod m$, A mendekripsi *ciphertext* C untuk mendapatkan pesan P menggunakan kunci sesi dengan metode matriks, yaitu $P \equiv {\boldsymbol{A}_{i\times i}}^{-1}(C -$

$\boldsymbol{B}_{i\times 1}$) (mod m) (proses perhitungan berbantuan dengan program MATLAB).

A mendapatkan pesan *P* dari B.

3.1.5 Implementasi Algoritma RSA dan *Affine Cipher* dengan Metode Matriks

3.1.5.1 Proses Pembentukan Kunci Publik dan Kunci Privat

Berikut ini langkah-langkah pembentukan kunci publik dan kunci privat:

1. Pilih dua bilangan prima sebarang, sebut $p = 13$ dan $q = 17$.
2. Hitung $n = pq$ dan $\phi(n) = (p-1)(q-1)$, sehingga diperoleh $n = 221$ dan $\phi(n) = 192$.
3. Ambil bilangan bulat e, sebut $e = 5$ yang relatif prima dengan 221 sebagai kunci publik.
4. Hitung d dengan $d = \frac{1+k\,\phi(n)}{e}$ dengan nilai $k = 1, 2, 3, \cdots, n-1$, $d = \frac{1+3\cdot 192}{5} = \frac{385}{5} = 77$ sehingga diperoleh nilai d yang bulat adalah $d = 77$ sebagai kunci privat.
5. Diperoleh kunci publik $e = 5$ dan $n = 221$ dan kunci privat $d = 77$ dan $n = 2217$. Selanjutnya kunci publik dikirim kepada pengirim pesan.

3.1.5.2 Proses Enkripsi Pesan dan Kunci Menggunakan Algoritma RSA dan *Affine Cipher* pada Matriks $2 \times l$

Dalam proses enkripsi pesan menggunakan algoritma *affine cipher* determinan kunci a harus relatif prima dengan m supaya dapat melakukan proses dekripsi (disini menggunakan kode ASCII 32 sampai 127 supaya mudah pembacaan karakter), sebut $m = 96$. Pada proses ini diambil contoh kunci $a =$ w41s, kunci $b =$ Al, dan pesan = Pesan rahasia, yang dalam kode ASCII kunci a adalah w = 119, 4 = 52, 1 = 49, dan s = 115, kunci b adalah A = 65 dan l = 108, dan pesan adalah p = 80, e = 101, s = 115, a = 97, n = 110, spasi = 32,

r = 114, a = 97, h = 104, a = 97, s = 115, i = 105, dan a = 97. Masukkan kunci a dan kunci b yang sudah diubah ke dalam kode ASCII pada matriks $\boldsymbol{A}$ yang berordo 2×2 dan matriks $\boldsymbol{B}$ yang berordo 2×1. Sebelum proses enkripsi pesan periksa apakah determinan kunci a relatif prima dengan 96.

$$\boldsymbol{A} = \begin{bmatrix} 119 & 52 \\ 49 & 115 \end{bmatrix},$$

$$\det(\boldsymbol{A}) = 119 \cdot 115 - 52 \cdot 49$$

$$= 13685 - 2548 = 11137.$$

Diperoleh $(11137, 96) = 1$. Selanjutnya proses enkripsi pesan. Pesan yang sudah diubah ke dalam kode ASCII dikurangi 32 supaya sesuai modulo 96 menjadi p $= 80 - 32 = 48$, e $= 101 - 32 = 69$, s $= 115 - 32 = 83$, a $= 97 - 32 = 65$, n $= 110 - 32 = 78$, spasi $= 32 - 32 = 0$, r $= 114 - 32 = 82$, a $= 97 - 32 = 65$, h $= 104 - 32 = 72$, a $= 97 - 32 = 65$, s $= 115 - 32 = 83$, i $= 105 - 32 = 73$, dan a $= 97 - 32 = 65$, kemudian masukkan pada matriks *plaintext* $\boldsymbol{P}$ yang berordo 2×7 dan entri yang kosong diisi dengan spasi$= 32 - 32 = 0$ sehingga diperoleh:

$$\boldsymbol{P} = \begin{bmatrix} 48 & 69 & 83 & 65 & 78 & 0 & 82 \\ 65 & 72 & 65 & 83 & 73 & 65 & 0 \end{bmatrix}.$$

Pada proses enkripsi, pesan dipotong menjadi matriks-matriks berordo 2×1 yaitu $\boldsymbol{P}_1 = \begin{bmatrix} 48 \\ 65 \end{bmatrix}$, $\boldsymbol{P}_2 = \begin{bmatrix} 69 \\ 72 \end{bmatrix}$, $\boldsymbol{P}_3 = \begin{bmatrix} 83 \\ 65 \end{bmatrix}$, $\boldsymbol{P}_4 = \begin{bmatrix} 65 \\ 83 \end{bmatrix}$, $\boldsymbol{P}_5 = \begin{bmatrix} 78 \\ 73 \end{bmatrix}$, $\boldsymbol{P}_6 = \begin{bmatrix} 0 \\ 65 \end{bmatrix}$, dan $\boldsymbol{P}_7 = \begin{bmatrix} 82 \\ 0 \end{bmatrix}$. Kemudian dilakukan enkripsi pesan yaitu $\boldsymbol{C}_i \equiv \boldsymbol{A}\boldsymbol{P}_i + \boldsymbol{B} \pmod{96}$.

$$\boldsymbol{C}_1 \equiv \begin{bmatrix} 119 & 52 \\ 49 & 115 \end{bmatrix} \cdot \begin{bmatrix} 48 \\ 65 \end{bmatrix} + \begin{bmatrix} 65 \\ 108 \end{bmatrix} \pmod{96} \equiv \begin{bmatrix} 37 \\ 47 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{C}_2 \equiv \begin{bmatrix} 119 & 52 \\ 49 & 115 \end{bmatrix} \cdot \begin{bmatrix} 69 \\ 72 \end{bmatrix} + \begin{bmatrix} 65 \\ 108 \end{bmatrix} \pmod{96} \equiv \begin{bmatrix} 20 \\ 57 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{C}_3 \equiv \begin{bmatrix}119 & 52\\ 49 & 115\end{bmatrix} \cdot \begin{bmatrix}83\\ 65\end{bmatrix} + \begin{bmatrix}65\\ 108\end{bmatrix} \pmod{96} \equiv \begin{bmatrix}74\\ 34\end{bmatrix} \pmod{96}$$

$$\boldsymbol{C}_4 \equiv \begin{bmatrix}119 & 52\\ 49 & 115\end{bmatrix} \cdot \begin{bmatrix}65\\ 83\end{bmatrix} + \begin{bmatrix}65\\ 108\end{bmatrix} \pmod{96} \equiv \begin{bmatrix}20\\ 70\end{bmatrix} \pmod{96}$$

$$\boldsymbol{C}_5 \equiv \begin{bmatrix}119 & 52\\ 49 & 115\end{bmatrix} \cdot \begin{bmatrix}78\\ 73\end{bmatrix} + \begin{bmatrix}65\\ 108\end{bmatrix} \pmod{96} \equiv \begin{bmatrix}87\\ 37\end{bmatrix} \pmod{96}$$

$$\boldsymbol{C}_6 \equiv \begin{bmatrix}119 & 52\\ 49 & 115\end{bmatrix} \cdot \begin{bmatrix}0\\ 65\end{bmatrix} + \begin{bmatrix}65\\ 108\end{bmatrix} \pmod{96} \equiv \begin{bmatrix}85\\ 95\end{bmatrix} \pmod{96}$$

$$\boldsymbol{C}_7 \equiv \begin{bmatrix}119 & 52\\ 49 & 115\end{bmatrix} \cdot \begin{bmatrix}82\\ 0\end{bmatrix} + \begin{bmatrix}65\\ 108\end{bmatrix} \pmod{96} \equiv \begin{bmatrix}31\\ 94\end{bmatrix} \pmod{96}$$

Sehingga diperoleh $\boldsymbol{C} = \begin{bmatrix}37 & 20 & 74 & 20 & 87 & 85 & 31\\ 47 & 57 & 34 & 70 & 37 & 95 & 94\end{bmatrix}$. Setelah diperoleh hasil enkripsi $\boldsymbol{C}$ kemudian ditambah dengan 32 supaya memenuhi karakter 32 sampai 127 menjadi $C = 37 + 32 = 69, 20 + 32 = 52, 74 + 32 = 106, 20 + 32 = 52, 87 + 32 = 119, 85 + 32 = 117, 31 + 32 = 63, 47 + 32 = 79, 57 + 32 = 89, 34 + 32 = 66, 70 + 32 = 102, 37 + 32 = 69, 95 + 32 = 127,$ dan $94 + 32 = 126$. Dengan kode ASCII diperoleh karakter aslinya menjadi $C =$ E4j4wu?OYBfE□~. Hasil akhir diperoleh pesan terenkripsi.

Setelah mengenkripsi pesan selanjutnya mengenkripsi kunci sesi dengan kunci publik yang sudah diterima. Dengan kode ASCII kunci a dan kunci b diperoleh $a_1 = 119, a_2 = 52, a_3 = 49,$ dan $a_4 = 115$ dan $b_1 = 65, b_2 = 108$. Kemudian lakukan proses enkripsi kunci yaitu $a_{ci} \equiv {a_i}^5 \pmod{221}$ dan $b_{ci} \equiv {b_i}^5 \pmod{221}$, sehingga diperoleh:

$a_{c1} \equiv 119^5 \pmod{221} \equiv 136 \pmod{221}$

$a_{c2} \equiv 52^5 \pmod{221} \equiv 53 \pmod{221}$

$a_{c3} \equiv 49^5 \pmod{221} \equiv 121 \pmod{221}$

$a_{c4} \equiv 115^5 \pmod{221} \equiv 98 \pmod{221}$

dan

$b_{c1} \equiv 65^5 \pmod{221} \equiv 182 \pmod{221}$

$b_{c2} \equiv 108^5 \pmod{221} \equiv 75 \pmod{221}$

Hasil akhir diperoleh kunci a dan kunci b terenkripsi yaitu $a_{c1} = 136, a_{c2} = 53, a_{c3} = 121$, dan $a_{c4} = 98$ dan $b_{c1} = 182$, dan $b_{c2} = 75$.

Setelah diperoleh pesan dan kunci terenkripsi kemudian dikirim kepada penerima pesan.

3.1.5.3 Proses Dekripsi Pesan dan Kunci Menggunakan Algoritma RSA dan *Affine Cipher* pada Matriks $2 \times l$

Sebelum melakukan proses dekripsi pesan, penerima pesan terlebih dahulu mendekripsikan kunci sesi terenkripsi menggunakan kunci privat yaitu $a_i \equiv {a_{ci}}^{77} \pmod{221}$ dan $b_i \equiv {b_{ci}}^{77} \pmod{221}$ sehingga diperoleh:

$a_1 \equiv 136^{77} \pmod{221} \equiv 119 \pmod{221}$

$a_2 \equiv 52^{77} \pmod{221} \equiv 52 \pmod{221}$

$a_3 \equiv 121^{77} \pmod{221} \equiv 49 \pmod{221}$

$a_4 \equiv 98^{77} \pmod{221} \equiv 115 \pmod{221}$

dan

$b_1 \equiv 182^{77} \pmod{221} \equiv 65 \pmod{221}$

$b_2 \equiv 75^{77} \pmod{221} \equiv 108 \pmod{221}$

Setelah diperoleh kunci a dan kunci b masukkan pada matriks $\boldsymbol{A}$ yang berordo 2×2 dan matriks $\boldsymbol{B}$ yang berordo 2×1 kemudian mencari invers dari $\boldsymbol{A} \pmod{96}$ atau $\boldsymbol{A}^{-1}$ untuk melakukan dekripsi. Terlebih dahulu mencari invers determinan $\boldsymbol{A} \pmod{m}$ atau $\det(\boldsymbol{A})^{-1}$ kemudian adjoin dari matriks $\boldsymbol{A}$ atau $\text{adj}(\boldsymbol{A})$ lalu diperoleh $\boldsymbol{A}^{-1} \equiv \det(\boldsymbol{A})^{-1} \cdot \text{adj}(\boldsymbol{A}) \pmod{m}$.

$$\boldsymbol{A} = \begin{bmatrix} 119 & 52 \\ 49 & 115 \end{bmatrix},$$

$$\det(\boldsymbol{A}) = 119 \cdot 115 - 52 \cdot 49 = 13685 - 2548 = 11137 \equiv 1 \pmod{96}.$$

Maka invers determinan $\boldsymbol{A}$ adalah $\det(\boldsymbol{A})^{-1} \equiv 1 \pmod{96}$. Adjoin untuk matriks $\boldsymbol{A}$ adalah

$$\text{adj}(\boldsymbol{A}) = \begin{bmatrix} 115 & -52 \\ -49 & 119 \end{bmatrix}.$$

Setelah diperoleh invers determinan matriks $\boldsymbol{A}$ dan adjoin matriks $\boldsymbol{A}$ kemudian mencari invers matriks $\boldsymbol{A} \pmod{96}$ yaitu $\boldsymbol{A}^{-1} \equiv \det(\boldsymbol{A})^{-1} \cdot \text{adj}(\boldsymbol{A}) \pmod{m}$.

$$\boldsymbol{A}^{-1} = 1 \cdot \begin{bmatrix} 115 & -52 \\ -49 & 119 \end{bmatrix} \equiv \begin{bmatrix} 19 & 44 \\ 47 & 23 \end{bmatrix} \pmod{96}.$$

Sebelum proses dekripsi mengubah pesan ke dalam kode ASCII sehingga diperoleh E = 69, 4 = 52, j = 106, 4 = 52, w = 119, u = 117, ? = 63, O = 79, Y = 89, B = 66, f = 102, E = 69, □= 127, dan ~ = 126. Masukkan pesan yang sudah diubah ke dalam kode ASCII pada matriks *ciphertext* $\boldsymbol{C}$ yang berordo 2×7 supaya sesuai modulo 96 maka setiap entri dikurangi 32 sehingga diperoleh:

$$\boldsymbol{C} = \begin{bmatrix} 37 & 20 & 74 & 20 & 87 & 85 & 31 \\ 47 & 57 & 34 & 70 & 37 & 95 & 94 \end{bmatrix}.$$

Selanjutnya proses dekripsi *ciphertext* yaitu $\boldsymbol{P}_i \equiv \boldsymbol{A}^{-1}(\boldsymbol{C}_i - \boldsymbol{B}) \pmod{96}$. Untuk dekripsi, *ciphertext* dipotong menjadi matriks-matriks berordo 2×1 yaitu $\boldsymbol{C}_1 = \begin{bmatrix} 37 \\ 47 \end{bmatrix}, \boldsymbol{C}_2 = \begin{bmatrix} 20 \\ 57 \end{bmatrix}, \boldsymbol{C}_3 = \begin{bmatrix} 74 \\ 34 \end{bmatrix}, \boldsymbol{C}_4 = \begin{bmatrix} 20 \\ 70 \end{bmatrix}, \boldsymbol{C}_5 = \begin{bmatrix} 87 \\ 37 \end{bmatrix}, \boldsymbol{C}_6 = \begin{bmatrix} 85 \\ 95 \end{bmatrix}$, dan $\boldsymbol{C}_7 = \begin{bmatrix} 82 \\ 0 \end{bmatrix}$ sehingga diperoleh:

$$\boldsymbol{P}_1 \equiv \begin{bmatrix} 19 & 44 \\ 47 & 23 \end{bmatrix} \cdot \left(\begin{bmatrix} 37 \\ 47 \end{bmatrix} - \begin{bmatrix} 65 \\ 108 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 48 \\ 65 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_2 \equiv \begin{bmatrix} 19 & 44 \\ 47 & 23 \end{bmatrix} \cdot \left(\begin{bmatrix} 20 \\ 57 \end{bmatrix} - \begin{bmatrix} 65 \\ 108 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 69 \\ 72 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_3 \equiv \begin{bmatrix} 19 & 44 \\ 47 & 23 \end{bmatrix} \cdot \left(\begin{bmatrix} 74 \\ 34 \end{bmatrix} - \begin{bmatrix} 65 \\ 108 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 83 \\ 65 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_4 \equiv \begin{bmatrix} 19 & 44 \\ 47 & 23 \end{bmatrix} \cdot \left(\begin{bmatrix} 20 \\ 70 \end{bmatrix} - \begin{bmatrix} 65 \\ 108 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 65 \\ 83 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_5 \equiv \begin{bmatrix}19 & 44\\ 47 & 23\end{bmatrix} \cdot \left(\begin{bmatrix}87\\ 37\end{bmatrix} - \begin{bmatrix}65\\ 108\end{bmatrix}\right) (\text{mod } 96) \equiv \begin{bmatrix}78\\ 73\end{bmatrix} (\text{mod } 96)$$

$$\boldsymbol{P}_6 \equiv \begin{bmatrix}19 & 44\\ 47 & 23\end{bmatrix} \cdot \left(\begin{bmatrix}85\\ 95\end{bmatrix} - \begin{bmatrix}65\\ 108\end{bmatrix}\right) (\text{mod } 96) \equiv \begin{bmatrix}0\\ 65\end{bmatrix} (\text{mod } 96)$$

$$\boldsymbol{P}_7 \equiv \begin{bmatrix}19 & 44\\ 47 & 23\end{bmatrix} \cdot \left(\begin{bmatrix}31\\ 94\end{bmatrix} - \begin{bmatrix}65\\ 108\end{bmatrix}\right) (\text{mod } 96) \equiv \begin{bmatrix}82\\ 0\end{bmatrix} (\text{mod } 96)$$

Sehingga diperoleh *plaintext* $\boldsymbol{P} = \begin{bmatrix}48 & 69 & 83 & 65 & 78 & 0 & 82\\ 65 & 72 & 65 & 83 & 73 & 65 & 0\end{bmatrix}$. Setelah diperoleh hasil dekripsi $\boldsymbol{P}$ kemudian ditambah dengan 32 supaya memenuhi karakter 32 sampai 127 menjadi $P = 48 + 32 = 80, 69 + 32 = 101, 83 + 32 = 115,\ 65 + 32 = 97,\ 78 + 32 = 110,\ 0 + 32 = 32, 82 + 32 = 114,\ 65 + 32 = 97,\ 72 + 32 = 104, 65 + 32 = 97, 83 + 32 = 115, 73 + 32 = 105,\ 65 + 32 = 97$ dan $0 + 32 = 32$. Dengan kode ASCII diperoleh karakter aslinya menjadi P =Pesan rahasia .

3.1.5.4 Proses Enkripsi Pesan dan Kunci Menggunakan Algoritma RSA dan *Affine Cipher* pada Matriks $3 \times l$

Dalam proses enkripsi pesan menggunakan algoritma *affine cipher* determinan kunci a harus relatif prima dengan m supaya dapat melakukan proses dekripsi (disini menggunakan kode ASCII 32 sampai 127 supaya mudah pembacaan karakter), sebut $m = 96$. Pada proses ini diambil contoh kunci a = Dari w41s, kunci b = Aal, dan pesan = Pesan rahasia, yang dalam kode ASCII kunci a adalah D = 68, a = 97, r = 114, i = 105, spasi = 32, w = 119, 4 = 52, 1 = 49, dan s = 115, kunci b adalah A = 65, a = 97, dan l = 108, dan pesan adalah p = 80, e = 101, s = 115, a = 97, n = 110, spasi = 32, r = 114, a = 97, h = 104, a = 97, s = 115, i = 105, dan a = 97. Masukkan kunci a dan kunci b yang sudah diubah ke dalam kode ASCII pada matriks $\boldsymbol{A}$ yang berordo 3×3 dan matriks $\boldsymbol{B}$ yang berordo 3×1. Sebelum proses enkripsi pesan periksa apakah

determinan kunci a relatif prima dengan 96.

$$\boldsymbol{A} = \begin{bmatrix} 68 & 97 & 114 \\ 105 & 32 & 119 \\ 52 & 49 & 115 \end{bmatrix}.$$

Dengan menggunakan kofaktor untuk menghitung determinan maka $|\boldsymbol{A}| = a_{11}C_{11} + a_{21}C_{21} + a_{31}C_{31}$.

Menentukan kofaktor C_{11}:

$$C_{11} = (-1)^{1+1} \cdot \begin{vmatrix} 32 & 119 \\ 49 & 115 \end{vmatrix} = (-1^2) \cdot -2151 = -2151$$

Menentukan kofaktor C_{21}:

$$C_{21} = (-1)^{2+1} \cdot \begin{vmatrix} 97 & 114 \\ 49 & 115 \end{vmatrix} = (-1^3) \cdot 5569 = -5569$$

Menentukan kofaktor C_{31}:

$$C_{31} = (-1)^{3+1} \cdot \begin{vmatrix} 97 & 114 \\ 32 & 119 \end{vmatrix} = (-1^4) \cdot 7895 = 7895$$

$$\begin{aligned} \det(\boldsymbol{A}) &= a_{11}C_{11} + a_{21}C_{21} + a_{31}c_{31} \\ &= 68 \cdot (-2151) + 105 \cdot (-5569) + 52 \cdot 7895 \\ &= -14628 + (-584745) + 410540 = -320473. \end{aligned}$$

Diperoleh $(-320473, 96) = 1$. Selanjutnya proses enkripsi pesan. Pesan yang sudah diubah ke dalam kode ASCII dikurangi 32 supaya sesuai modulo 96 menjadi p $= 80 - 32 = 48$, e $= 101 - 32 = 69$, s $= 115 - 32 = 83$, a $= 97 - 32 = 65$, n $= 110 - 32 = 78$, spasi $= 32 - 32 = 0$, r $= 114 - 32 = 82$, a $= 97 - 32 = 65$, h $= 104 - 32 = 72$, a $= 97 - 32 = 65$, s $= 115 - 32 = 83$, i $= 105 - 32 = 73$, dan a $= 97 - 32 = 65$, kemudian masukkan pada matriks *plaintext* $\boldsymbol{P}$ yang berordo 3×5 dan entri yang kosong diisi dengan spasi$= 32 - 32 = 0$ sehingga diperoleh:

$$\boldsymbol{P} = \begin{bmatrix} 48 & 69 & 83 & 65 & 78 \\ 0 & 82 & 65 & 72 & 65 \\ 83 & 73 & 65 & 0 & 0 \end{bmatrix}.$$

Pada proses enkripsi, pesan dipotong menjadi matriks-matriks berordo 3×1 yaitu $\boldsymbol{P}_1 = \begin{bmatrix} 48 \\ 0 \\ 83 \end{bmatrix}$, $\boldsymbol{P}_2 = \begin{bmatrix} 69 \\ 82 \\ 73 \end{bmatrix}$, $\boldsymbol{P}_3 = \begin{bmatrix} 83 \\ 65 \\ 65 \end{bmatrix}$, $\boldsymbol{P}_4 = \begin{bmatrix} 65 \\ 72 \\ 0 \end{bmatrix}$, dan $\boldsymbol{P}_1 = \begin{bmatrix} 78 \\ 65 \\ 0 \end{bmatrix}$. Kemudian lakukan enkripsi pesan yaitu $\boldsymbol{C}_i \equiv \boldsymbol{A}\boldsymbol{P}_i + \boldsymbol{B} \pmod{96}$.

$$C_1 \equiv \begin{bmatrix} 68 & 97 & 114 \\ 105 & 32 & 119 \\ 52 & 49 & 115 \end{bmatrix} \cdot \begin{bmatrix} 48 \\ 0 \\ 83 \end{bmatrix} + \begin{bmatrix} 65 \\ 97 \\ 108 \end{bmatrix} \pmod{96} \equiv \begin{bmatrix} 23 \\ 38 \\ 53 \end{bmatrix} \pmod{96}$$

$$C_2 \equiv \begin{bmatrix} 68 & 97 & 114 \\ 105 & 32 & 119 \\ 52 & 49 & 115 \end{bmatrix} \cdot \begin{bmatrix} 69 \\ 82 \\ 73 \end{bmatrix} + \begin{bmatrix} 65 \\ 97 \\ 108 \end{bmatrix} \pmod{96} \equiv \begin{bmatrix} 9 \\ 29 \\ 77 \end{bmatrix} \pmod{96}$$

$$C_3 \equiv \begin{bmatrix} 68 & 97 & 114 \\ 105 & 32 & 119 \\ 52 & 49 & 115 \end{bmatrix} \cdot \begin{bmatrix} 83 \\ 65 \\ 65 \end{bmatrix} + \begin{bmatrix} 65 \\ 97 \\ 108 \end{bmatrix} \pmod{96} \equiv \begin{bmatrix} 32 \\ 3 \\ 12 \end{bmatrix} \pmod{96}$$

$$C_4 \equiv \begin{bmatrix} 68 & 97 & 114 \\ 105 & 32 & 119 \\ 52 & 49 & 115 \end{bmatrix} \cdot \begin{bmatrix} 65 \\ 72 \\ 0 \end{bmatrix} + \begin{bmatrix} 65 \\ 97 \\ 108 \end{bmatrix} \pmod{96} \equiv \begin{bmatrix} 45 \\ 10 \\ 8 \end{bmatrix} \pmod{96}$$

$$C_5 \equiv \begin{bmatrix} 68 & 97 & 114 \\ 105 & 32 & 119 \\ 52 & 49 & 115 \end{bmatrix} \cdot \begin{bmatrix} 78 \\ 65 \\ 0 \end{bmatrix} + \begin{bmatrix} 65 \\ 97 \\ 108 \end{bmatrix} \pmod{96} \equiv \begin{bmatrix} 58 \\ 95 \\ 53 \end{bmatrix} \pmod{96}$$

Sehingga diperoleh $\boldsymbol{C} = \begin{bmatrix} 23 & 9 & 32 & 45 & 58 \\ 38 & 29 & 3 & 10 & 95 \\ 53 & 77 & 12 & 8 & 53 \end{bmatrix}$. Setelah diperoleh hasil enkripsi C kemudian ditambah dengan 32 supaya memenuhi karakter 32 sampai 127 menjadi $C = 23 + 32 = 55, 9 + 32 = 41, 32 + 32 = 64, 45 + 32 = 77, 58 + 32 = 90, 38 + 32 = 70, 29 + 32 = 61, 3 + 32 = 35, 10 + 32 = 42, 95 + 32 = 127, 53 + 32 = 85, 77 + 32 = 109, 12 + 32 = 44, 8 + 32 = 40$, dan $53 + 32 = 85$. Dengan kode ASCII diperoleh karakter aslinya menjadi $C =$ 7)@MZF=#*□Um,(U. Hasil akhir diperoleh pesan terenkripsi.

Setelah mengenkripsi pesan selanjutnya mengenkripsi kunci sesi dengan

kunci publik yang sudah diterima. Dengan kode ASCII kunci a dan kunci b diperoleh $a_1 = 68, a_2 = 97, a_3 = 114, a_4 = 105, a_5 = 32, a_6 = 119, a_7 = 52,$ $a_8 = 49,$ dan $a_9 = 115,$ dan $b_1 = 65, b_2 = 97,$ dan $b_3 = 108.$ Kemudian lakukan proses enkripsi kunci yaitu $a_{ci} \equiv {a_i}^5 \pmod{221}$ dan $b_{ci} \equiv {b_i}^5 \pmod{221}$, sehingga diperoleh:

$a_{c1} \equiv 68^5 \pmod{221} \equiv 204 \pmod{221}$

$a_{c2} \equiv 97^5 \pmod{221} \equiv 54 \pmod{221}$

$a_{c3} \equiv 114^5 \pmod{221} \equiv 173 \pmod{221}$

$a_{c4} \equiv 105^5 \pmod{221} \equiv 209 \pmod{221}$

$a_{c5} \equiv 32^5 \pmod{221} \equiv 2 \pmod{221}$

$a_{c6} \equiv 119^5 \pmod{221} \equiv 136 \pmod{221}$

$a_{c7} \equiv 52^5 \pmod{221} \equiv 53 \pmod{221}$

$a_{c8} \equiv 49^5 \pmod{221} \equiv 121 \pmod{221}$

$a_{c9} \equiv 115^5 \pmod{221} \equiv 98 \pmod{221}$

dan

$b_{c1} \equiv 65^5 \pmod{221} \equiv 182 \pmod{221}$

$b_{c2} \equiv 97^5 \pmod{221} \equiv 54 \pmod{221}$

$b_{c3} \equiv 108^5 \pmod{221} \equiv 75 \pmod{221}$

Hasil akhir diperoleh kunci a dan kunci b terenkripsi yaitu $a_{c1} = 204, a_{c2} = 54, a_{c3} = 173, a_{c4} = 209, a_{c5} = 2, a_{c6} = 136, a_{c7} = 53, a_{c8} = 121,$ dan $a_{c9} = 98,$ dan $b_{c1} = 182, b_{c2} = 54,$ dan $b_{c3} = 75.$

Setelah diperoleh pesan dan kunci terenkripsi kemudian dikirim kepada penerima pesan.

3.1.5.5 Proses Dekripsi Pesan dan Kunci Menggunakan Algoritma RSA dan *Affine Cipher* pada Matriks $3 \times l$

Sebelum melakukan proses dekripsi pesan terlebih dahulu mendekripsikan kunci sesi terenkripsi menggunakan kunci privat yaitu $a_i \equiv {a_{ci}}^{77} \pmod{221}$ dan $b_i \equiv {b_{ci}}^{77} \pmod{221}$ sehingga diperoleh:

$a_1 \equiv 204^{77} \pmod{221} \equiv 68 \pmod{221}$

$a_2 \equiv 54^{77} \pmod{221} \equiv 97 \pmod{221}$

$a_3 \equiv 173^{77} \pmod{221} \equiv 114 \pmod{221}$

$a_4 \equiv 209^{77} \pmod{221} \equiv 105 \pmod{221}$

$a_5 \equiv 2^{77} \pmod{221} \equiv 32 \pmod{221}$

$a_6 \equiv 136^{77} \pmod{221} \equiv 119 \pmod{221}$

$a_7 \equiv 52^{77} \pmod{221} \equiv 52 \pmod{221}$

$a_8 \equiv 121^{77} \pmod{221} \equiv 49 \pmod{221}$

$a_9 \equiv 98^{77} \pmod{221} \equiv 115 \pmod{221}$

dan

$b_1 \equiv 182^{77} \pmod{221} \equiv 65 \pmod{221}$

$b_2 \equiv 54^{77} \pmod{221} \equiv 97 \pmod{221}$

$b_3 \equiv 75^{77} \pmod{221} \equiv 108 \pmod{221}$

Setelah diperoleh kunci a dan kunci b masukkan pada matriks $\boldsymbol{A}$ yang berordo 3×3 dan matriks $\boldsymbol{B}$ yang berordo 3×1 kemudian mencari invers dari $\boldsymbol{A} \pmod{96}$ atau $\boldsymbol{A}^{-1}$ untuk melakukan dekripsi. Terlebih dahulu mencari invers determinan $\boldsymbol{A} \pmod{m}$ atau $\det(\boldsymbol{A})^{-1}$ kemudian adjoin dari matriks $\boldsymbol{A}$ atau $\text{adj}(\boldsymbol{A})$ lalu diperoleh $\boldsymbol{A}^{-1} \equiv \det(\boldsymbol{A})^{-1} \cdot \text{adj}(\boldsymbol{A}) \pmod{96}$.

$$\boldsymbol{A} = \begin{bmatrix} 68 & 97 & 114 \\ 105 & 32 & 119 \\ 52 & 49 & 115 \end{bmatrix}.$$

Dengan menggunakan kofaktor untuk menghitung determinan maka $|\boldsymbol{A}| = a_{11}C_{11} + a_{21}C_{21} + a_{31}C_{31}$.

Menentukan kofaktor C_{11}:

$$C_{11} = (-1)^{1+1} \cdot \begin{vmatrix} 32 & 119 \\ 49 & 115 \end{vmatrix} = (-1^2) \cdot -2151 = -2151$$

Menentukan kofaktor C_{21}:

$$C_{21} = (-1)^{2+1} \cdot \begin{vmatrix} 97 & 114 \\ 49 & 115 \end{vmatrix} = (-1^3) \cdot 5569 = -5569$$

Menentukan kofaktor C_{31}:

$$C_{31} = (-1)^{3+1} \cdot \begin{vmatrix} 97 & 114 \\ 32 & 119 \end{vmatrix} = (-1^4) \cdot 7895 = 7895$$

$$\begin{aligned} \det(\boldsymbol{A}) &= a_{11}C_{11} + a_{21}C_{21} + a_{31}C_{31} \\ &= 68 \cdot (-2151) + 105 \cdot (-5569) + 52 \cdot 7895 \\ &= -14628 + (-584745) + 410540 \\ &= -320473 \equiv 71 \pmod{96}. \end{aligned}$$

Maka invers determinan $\boldsymbol{A}$ adalah $\boldsymbol{A}^{-1} \equiv 23 \pmod{96}$. Adjoin untuk matriks $\boldsymbol{A}$ adalah

$$\text{adj}(\boldsymbol{A}) \equiv \begin{bmatrix} 57 & 95 & 23 \\ 65 & 68 & 38 \\ 25 & 80 & 55 \end{bmatrix} \pmod{96}.$$

Setelah diperoleh invers determinan matriks $\boldsymbol{A}$ dan adjoin matriks $\boldsymbol{A}$ kemudian mencari invers matriks $\boldsymbol{A} \pmod{96}$ yaitu $\boldsymbol{A}^{-1} \equiv \det(\boldsymbol{A})^{-1} \cdot \text{adj}(\boldsymbol{A}) \pmod{96}$.

$$\boldsymbol{A}^{-1} \equiv 23 \cdot \begin{bmatrix} 57 & 95 & 23 \\ 65 & 68 & 38 \\ 25 & 80 & 55 \end{bmatrix} \pmod{96} \equiv \begin{bmatrix} 63 & 73 & 49 \\ 45 & 28 & 10 \\ 95 & 16 & 17 \end{bmatrix} \pmod{96}.$$

Sebelum proses dekripsi mengubah pesan ke dalam kode ASCII sehingga diperoleh 7 = 55,) = 41, @ = 64, M = 77, Z = 90, F = 70, = = 61, # = 35, ∗ = 42, DEL = 127, U = 85, m = 109, , = 44, (= 40, dan U = 85. Masukkan

pesan yang sudah diubah pada ke dalam kode ASCII matriks *ciphertext* $\boldsymbol{C}$ yang berordo 3×5 supaya sesuai modulo 96 maka setiap entri dikurangi 32 sehingga diperoleh:

$$\boldsymbol{C} = \begin{bmatrix} 23 & 9 & 32 & 45 & 58 \\ 38 & 29 & 3 & 10 & 95 \\ 53 & 77 & 12 & 8 & 53 \end{bmatrix}.$$

Selanjutnya proses dekripsi pesan yaitu $\boldsymbol{P}_i \equiv \boldsymbol{A}^{-1}(\boldsymbol{C}_i - \boldsymbol{B}) \pmod{96}$. Untuk dekripsi, *ciphertext* dipotong menjadi matriks-matriks berordo 3×1 yaitu $\boldsymbol{C}_1 = \begin{bmatrix} 23 \\ 38 \\ 53 \end{bmatrix}, \boldsymbol{C}_2 = \begin{bmatrix} 9 \\ 29 \\ 77 \end{bmatrix}, \boldsymbol{C}_3 = \begin{bmatrix} 32 \\ 3 \\ 12 \end{bmatrix}, \boldsymbol{C}_4 = \begin{bmatrix} 45 \\ 10 \\ 8 \end{bmatrix}$, dan $\boldsymbol{C}_1 = \begin{bmatrix} 58 \\ 95 \\ 53 \end{bmatrix}$ sehingga diperoleh:

$$\boldsymbol{P}_1 \equiv \begin{bmatrix} 63 & 73 & 49 \\ 45 & 28 & 10 \\ 95 & 16 & 17 \end{bmatrix} \cdot \left(\begin{bmatrix} 23 \\ 38 \\ 53 \end{bmatrix} - \begin{bmatrix} 65 \\ 97 \\ 108 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 48 \\ 0 \\ 83 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_2 \equiv \begin{bmatrix} 63 & 73 & 49 \\ 45 & 28 & 10 \\ 95 & 16 & 17 \end{bmatrix} \cdot \left(\begin{bmatrix} 9 \\ 29 \\ 77 \end{bmatrix} - \begin{bmatrix} 65 \\ 97 \\ 108 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 69 \\ 82 \\ 73 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_3 \equiv \begin{bmatrix} 63 & 73 & 49 \\ 45 & 28 & 10 \\ 95 & 16 & 17 \end{bmatrix} \cdot \left(\begin{bmatrix} 32 \\ 3 \\ 12 \end{bmatrix} - \begin{bmatrix} 65 \\ 97 \\ 108 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 83 \\ 65 \\ 65 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_4 \equiv \begin{bmatrix} 63 & 73 & 49 \\ 45 & 28 & 10 \\ 95 & 16 & 17 \end{bmatrix} \cdot \left(\begin{bmatrix} 45 \\ 10 \\ 8 \end{bmatrix} - \begin{bmatrix} 65 \\ 97 \\ 108 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 65 \\ 72 \\ 0 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_5 \equiv \begin{bmatrix} 63 & 73 & 49 \\ 45 & 28 & 10 \\ 95 & 16 & 17 \end{bmatrix} \cdot \left(\begin{bmatrix} 58 \\ 95 \\ 53 \end{bmatrix} - \begin{bmatrix} 65 \\ 97 \\ 108 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 78 \\ 65 \\ 0 \end{bmatrix} \pmod{96}$$

Sehingga diperoleh $\boldsymbol{P} = \begin{bmatrix} 48 & 69 & 83 & 65 & 78 \\ 0 & 82 & 65 & 72 & 65 \\ 83 & 73 & 65 & 0 & 0 \end{bmatrix}$. Setelah diperoleh hasil dekripsi $\boldsymbol{P}$ kemudian ditambah dengan 32 supaya memenuhi karakter 32 sampai 127 menjadi $P = 48 + 32 = 80, 69 + 32 = 101, 83 + 32 = 115, 65 + 32 = 97,\ 78 + 32 = 110,\ 0 + 32 = 32\ , 82 + 32 = 114,\ 65 + 32 = 97,\ 72 + 32 = 104,\ 65 + 32 = 97, 83 + 32 = 115,\ 73 + 32 = 105,\ 65 + 32 = 97, 0 + 32 =$

32, dan $0 + 32 = 32$. Dengan kode ASCII diperoleh karakter aslinya menjadi $P =$ Pesan rahasia .

3.1.5.6 Proses Enkripsi Pesan dan Kunci Menggunakan Algoritma RSA dan *Affine Cipher* pada Matriks $4 \times l$

Dalam proses enkripsi pesan menggunakan algoritma *affine cipher* determinan kunci a harus relatif prima dengan m supaya dapat melakukan proses dekripsi (disini menggunakan kode ASCII 32 sampai 127 supaya mudah pembacaan karakter), sebut $m = 96$. Pada proses ini diambil contoh kunci $a =$ Muhamad wais al, kunci $b =$ Wais, dan pesan = Pesan rahasia, yang dalam kode ASCII kunci a adalah M = 77, u = 117, h = 104, a = 97, m = 109, a = 97, d = 100, spasi = 32, w = 119, a = 97, i = 105, s = 115, spasi = 32, a = 97, l = 108, kunci b adalah W = 87, a = 97, i = 105, dan s = 115, dan pesan adalah p = 80, e = 101, s = 115, a = 97, n = 110, spasi = 32, r = 114, a = 97, h = 104, a = 97, s = 115, i = 105, dan a = 97. Masukkan kunci a dan kunci b yang sudah diubah ke dalam kode ASCII pada matriks $\boldsymbol{A}$ yang berordo 4×4 dan matriks $\boldsymbol{B}$ yang berordo 4×1. Sebelum proses enkripsi pesan periksa apakah determinan kunci a relatif prima dengan 96.

$$\boldsymbol{A} = \begin{bmatrix} 77 & 117 & 104 & 97 \\ 109 & 97 & 100 & 32 \\ 119 & 97 & 105 & 115 \\ 32 & 97 & 108 & 32 \end{bmatrix}.$$

Dengan menggunakan kofaktor untuk menghitung determinan maka $|\boldsymbol{A}| = a_{11}C_{11} + a_{21}C_{21} + a_{31}C_{31} + a_{41}C_{41}$.

Menentukan kofaktor C_{11}:

$$C_{11} = (-1)^{1+1} \cdot \begin{vmatrix} 97 & 100 & 32 \\ 97 & 105 & 115 \\ 97 & 108 & 32 \end{vmatrix} = (-1^2) \cdot -64408 = -64408$$

Menentukan kofaktor C_{21}:

$$C_{21} = (-1)^{2+1} \cdot \begin{vmatrix} 117 & 104 & 97 \\ 97 & 105 & 115 \\ 97 & 108 & 32 \end{vmatrix} = (-1^3) \cdot -194489 = 194489$$

Menentukan kofaktor C_{31}:

$$C_{31} = (-1)^{3+1} \cdot \begin{vmatrix} 117 & 104 & 97 \\ 97 & 100 & 32 \\ 97 & 108 & 32 \end{vmatrix} = (-1^4) \cdot 45320 = 45320$$

Menentukan kofaktor C_{41}:

$$C_{41} = (-1)^{4+1} \cdot \begin{vmatrix} 117 & 104 & 97 \\ 97 & 100 & 32 \\ 97 & 105 & 115 \end{vmatrix} = (-1^5) \cdot 162121 = -162121$$

$$\begin{aligned} \det(\boldsymbol{A}) &= a_{11}C_{11} + a_{21}C_{21} + a_{31}C_{31} + a_{41}C_{41} \\ &= 77 \cdot (-64408) + 109 \cdot 194489 + 119 \cdot 45320 + 32 \cdot (-162121) \\ &= -4959416 + 21199301 + 5393080 + (-5187872) = 16445093. \end{aligned}$$

Diperoleh $(16445093, 96) = 1$. Selanjutnya proses enkripsi pesan. Pesan yang sudah diubah ke dalam kode ASCII dikurangi 32 supaya sesuai modulo 96 menjadi p $= 80 - 32 = 48$, e $= 101 - 32 = 69$, s $= 115 - 32 = 83$, a $= 97 - 32 = 65$, n $= 110 - 32 = 78$, spasi $= 32 - 32 = 0$, r $= 114 - 32 = 82$, a $= 97 - 32 = 65$, h $= 104 - 32 = 72$, a $= 97 - 32 = 65$, s $= 115 - 32 = 83$, i $= 105 - 32 = 73$, dan a $= 97 - 32 = 65$, kemudian masukkan pada matriks *plaintext* $\boldsymbol{P}$ yang berordo 4×4 dan entri yang kosong diisi dengan spasi$= 32 - 32 = 0$ sehingga diperoleh:

$$\boldsymbol{P} = \begin{bmatrix} 48 & 69 & 83 & 65 \\ 78 & 0 & 82 & 65 \\ 72 & 65 & 83 & 73 \\ 65 & 0 & 0 & 0 \end{bmatrix}.$$

Pada proses enkripsi, pesan dipotong menjadi matriks-matriks berordo 4×1 yaitu

$\boldsymbol{P}_1 = \begin{bmatrix}48\\78\\72\\65\end{bmatrix}, \boldsymbol{P}_2 = \begin{bmatrix}69\\0\\65\\0\end{bmatrix}, \boldsymbol{P}_3 = \begin{bmatrix}83\\82\\83\\0\end{bmatrix}$, dan $\boldsymbol{P}_4 = \begin{bmatrix}65\\65\\73\\0\end{bmatrix}$. Kemudian dilakukan enkripsi pesan yaitu $\boldsymbol{C}_i \equiv \boldsymbol{A}\boldsymbol{P}_i + \boldsymbol{B} \pmod{96}$.

$$\boldsymbol{C}_1 \equiv \begin{bmatrix}77 & 117 & 104 & 97\\109 & 97 & 100 & 32\\119 & 97 & 105 & 115\\32 & 97 & 108 & 32\end{bmatrix} \cdot \begin{bmatrix}48\\78\\72\\65\end{bmatrix} + \begin{bmatrix}87\\97\\105\\115\end{bmatrix} \pmod{96} \equiv \begin{bmatrix}14\\95\\2\\65\end{bmatrix} \pmod{96}$$

$$\boldsymbol{C}_2 \equiv \begin{bmatrix}77 & 117 & 104 & 97\\109 & 97 & 100 & 32\\119 & 97 & 105 & 115\\32 & 97 & 108 & 32\end{bmatrix} \cdot \begin{bmatrix}69\\0\\65\\0\end{bmatrix} + \begin{bmatrix}87\\97\\105\\115\end{bmatrix} (\bmod 96) \equiv \begin{bmatrix}64\\6\\69\\31\end{bmatrix} \pmod{96}$$

$$\boldsymbol{C}_3 \equiv \begin{bmatrix}77 & 117 & 104 & 97\\109 & 97 & 100 & 32\\119 & 97 & 105 & 115\\32 & 97 & 108 & 32\end{bmatrix} \cdot \begin{bmatrix}83\\82\\83\\0\end{bmatrix} + \begin{bmatrix}87\\97\\105\\115\end{bmatrix} \pmod{96} \equiv \begin{bmatrix}32\\54\\59\\9\end{bmatrix} \pmod{96}$$

$$\boldsymbol{C}_4 \equiv \begin{bmatrix}77 & 117 & 104 & 97\\109 & 97 & 100 & 32\\119 & 97 & 105 & 115\\32 & 97 & 108 & 32\end{bmatrix} \cdot \begin{bmatrix}65\\65\\73\\0\end{bmatrix} + \begin{bmatrix}87\\97\\105\\115\end{bmatrix} \pmod{96} \equiv \begin{bmatrix}33\\51\\18\\64\end{bmatrix} \pmod{96}$$

Sehingga diperoleh $\boldsymbol{C} = \begin{bmatrix}14 & 64 & 32 & 33\\95 & 6 & 54 & 51\\2 & 69 & 59 & 18\\65 & 31 & 9 & 64\end{bmatrix}$. Setelah diperoleh hasil enkripsi $\boldsymbol{C}$ kemudian ditambah dengan 32 supaya memenuhi karakter 32 sampai 127 menjadi $C = 14 + 32 = 46,\ 64 + 32 = 96,\ 32 + 32 = 64,\ 33 + 32 = 65,\ 95 + 32 = 127,\ 6 + 32 = 38,\ 54 + 32 = 86,\ 51 + 32 = 83,\ 2 + 32 = 34,\ 69 + 32 = 101,\ 59 + 32 = 91,\ 18 + 32 = 50, 65 + 32 = 97, 31 + 32 = 63, 9 + 32 = 41$, dan $64 + 32 = 96$. Dengan kode ASCII diperoleh karakter aslinya menjadi $C =$.`@A□&VS"e[2a?)`. Hasil akhir diperoleh pesan terenkripsi.

Setelah mengenkripsi pesan selanjutnya mengenkripsi kunci sesi dengan kunci publik yang sudah diterima. Dengan kode ASCII kunci a dan kunci b diperoleh kunci $a_1 = 77, a_2 = 117, a_3 = 104, a_4 = 97, a_5 = 109, a_6 = 97, a_7 =$

$100, a_8 = 32,\ a_9 = 119, a_{10} = 97, a_{11} = 105,\ a_{12} = 115, a_{13} = 32,\ a_{11} = 97,$ $a_{15} = 108$, dan $a_{16} = 32$ dan kunci $b_1 = 87, b_2 = 97, b_3 = 105$, dan $b_4 = 115$. Kemudian dilakukan proses enkripsi kunci yaitu $a_{ci} \equiv {a_i}^5 \pmod{221}$ dan $b_{ci} \equiv {b_i}^5 \pmod{221}$, sehingga diperoleh:

$a_{c1} \equiv 77^5 \pmod{221} \equiv 25 \pmod{221}$

$a_{c2} \equiv 117^5 \pmod{221} \equiv 104 \pmod{221}$

$a_{c3} \equiv 104^5 \pmod{221} \equiv 117 \pmod{221}$

$a_{c4} \equiv 97^5 \pmod{221} \equiv 54 \pmod{221}$

$a_{c5} \equiv 109^5 \pmod{221} \equiv 96 \pmod{221}$

$a_{c6} \equiv 97^5 \pmod{221} \equiv 54 \pmod{221}$

$a_{c7} \equiv 100^5 \pmod{221} \equiv 172 \pmod{221}$

$a_{c8} \equiv 32^5 \pmod{221} \equiv 2 \pmod{221}$

$a_{c9} \equiv 119^5 \pmod{221} \equiv 136 \pmod{221}$

$a_{c10} \equiv 979^5 \pmod{221} \equiv 54 \pmod{221}$

$a_{c11} \equiv 105^5 \pmod{221} \equiv 209 \pmod{221}$

$a_{c12} \equiv 115^5 \pmod{221} \equiv 98 \pmod{221}$

$a_{c13} \equiv 32^5 \pmod{221} \equiv 2 \pmod{221}$

$a_{c14} \equiv 97^5 \pmod{221} \equiv 54 \pmod{221}$

$a_{c15} \equiv 108^5 \pmod{221} \equiv 75 \pmod{221}$

$a_{c16} \equiv 32^5 \pmod{221} \equiv 2 \pmod{221}$

dan

$b_{c1} \equiv 87^5 \pmod{221} \equiv 185 \pmod{221}$

$b_{c2} \equiv 97^5 \pmod{221} \equiv 54 \pmod{221}$

$b_{c3} \equiv 105^5 \pmod{221} \equiv 209 \pmod{221}$

$b_{c4} \equiv 115^5 \pmod{221} \equiv 98 \pmod{221}$

Hasil akhir diperoleh kunci a dan kunci b terenkripsi yaitu $a_{c1} = 25, a_{c2} = 104, a_{c3} = 117, a_{c4} = 54, a_{c5} = 96, a_{c6} = 54, a_{c7} = 172, a_{c8} = 2, a_{c9} = 136, a_{c10} = 54, a_{c11} = 209, a_{c12} = 98, a_{c13} = 2, a_{c14} = 54, a_{c15} = 75,$ dan $a_{c16} = 2$ dan $b_{c1} = 185, b_{c2} = 54, b_{c3} = 209,$ dan $b_{c4} = 98$. Setelah diperoleh pesan dan kunci terenkripsi kemudian dikirim kepada penerima pesan.

3.1.5.7 Proses Dekripsi Pesan dan Kunci Menggunakan Algoritma RSA dan *Affine Cipher* pada Matriks $4 \times l$

Sebelum melakukan proses dekripsi pesan terlebih dahulu mendekripsikan Kunci sesi terenkripsi menggunakan kunci privat yaitu $a_i \equiv {a_{ci}}^{77} \pmod{221}$ dan $b_i \equiv {b_{ci}}^{77} \pmod{221}$ sehingga diperoleh:

$a_1 \equiv 25^{77} \pmod{221} \equiv 77 \pmod{221}$

$a_2 \equiv 104^{77} \pmod{221} \equiv 117 \pmod{221}$

$a_3 \equiv 117^{77} \pmod{221} \equiv 104 \pmod{221}$

$a_4 \equiv 54^{77} \pmod{221} \equiv 97 \pmod{221}$

$a_5 \equiv 96^{77} \pmod{221} \equiv 109 \pmod{221}$

$a_6 \equiv 54^{77} \pmod{221} \equiv 97 \pmod{221}$

$a_7 \equiv 172^{77} \pmod{221} \equiv 100 \pmod{221}$

$a_8 \equiv 2^{77} \pmod{221} \equiv 32 \pmod{221}$

$a_9 \equiv 136^{77} \pmod{221} \equiv 119 \pmod{221}$

$a_{10} \equiv 54^{77} \pmod{221} \equiv 97 \pmod{221}$

$a_{11} \equiv 209^{77} \pmod{221} \equiv 105 \pmod{221}$

$a_{12} \equiv 98^{77} \pmod{221} \equiv 115 \pmod{221}$

$a_{13} \equiv 2^{77} \pmod{221} \equiv 32 \pmod{221}$

$a_{14} \equiv 54^{77} \pmod{221} \equiv 97 \pmod{221}$

$a_{15} \equiv 75^{77} \pmod{221} \equiv 108 \pmod{221}$

$a_{16} \equiv 2^{77} \pmod{221} \equiv 32 \pmod{221}$

dan

$b_1 \equiv 185^{77} \pmod{221} \equiv 87 \pmod{221}$

$b_2 \equiv 54^{77} \pmod{221} \equiv 97 \pmod{221}$

$b_3 \equiv 209^{77} \pmod{221} \equiv 105 \pmod{221}$

$b_4 \equiv 98^{77} \pmod{221} \equiv 115 \pmod{221}$

Setelah diperoleh kunci a dan kunci b masukkan pada matriks $\boldsymbol{A}$ yang berordo 4×4 dan matriks $\boldsymbol{B}$ yang berordo 4×1 kemudian mencari invers dari $\boldsymbol{A} \pmod{96}$ atau $\boldsymbol{A}^{-1}$ untuk melakukan dekripsi. Terlebih dahulu mencari invers determinan $\boldsymbol{A} \pmod{96}$ atau $\det(\boldsymbol{A})^{-1}$ kemudian adjoin dari matriks $\boldsymbol{A}$ atau $\text{adj}(\boldsymbol{A})$ lalu diperoleh $\boldsymbol{A}^{-1} \equiv \det(\boldsymbol{A})^{-1} \cdot \text{adj}(\boldsymbol{A}) \pmod{96}$.

$$\boldsymbol{A} = \begin{bmatrix} 77 & 117 & 104 & 97 \\ 109 & 97 & 100 & 32 \\ 119 & 97 & 105 & 115 \\ 32 & 97 & 108 & 32 \end{bmatrix}.$$

Dengan menggunakan kofaktor untuk menghitung determinan maka $|\boldsymbol{A}| = a_{11}C_{11} + a_{21}C_{21} + a_{31}C_{31} + a_{41}C_{41}$.

Menentukan kofaktor C_{11}:

$$C_{11} = (-1)^{1+1} \cdot \begin{vmatrix} 97 & 100 & 32 \\ 97 & 105 & 115 \\ 97 & 108 & 32 \end{vmatrix} = (-1^2) \cdot -64408 = -64408$$

Menentukan kofaktor C_{21}:

$$C_{21} = (-1)^{2+1} \cdot \begin{vmatrix} 117 & 104 & 97 \\ 97 & 105 & 115 \\ 97 & 108 & 32 \end{vmatrix} = (-1^3) \cdot -194489 = 194489$$

Menentukan kofaktor C_{31}:

$$C_{31} = (-1)^{3+1} \cdot \begin{vmatrix} 117 & 104 & 97 \\ 97 & 100 & 32 \\ 97 & 108 & 32 \end{vmatrix} = (-1^4) \cdot 45320 = 45320$$

Menentukan kofaktor C_{41}:

$$C_{41} = (-1)^{4+1} \cdot \begin{vmatrix} 117 & 104 & 97 \\ 97 & 100 & 32 \\ 97 & 105 & 115 \end{vmatrix} = (-1^5) \cdot 162121 = -162121$$

$$\begin{aligned} \det(\boldsymbol{A}) &= a_{11}C_{11} + a_{21}C_{21} + a_{31}C_{31} + a_{41}C_{41} \\ &= 77 \cdot (-64408) + 109 \cdot 194489 + 119 \cdot 45320 + 32 \cdot (-162121) \\ &= -4959416 + 21199301 + 5393080 + (-5187872) \\ &= 16445093 \equiv 5 \pmod{96}. \end{aligned}$$

Maka invers determinan $\boldsymbol{A}$ adalah $\boldsymbol{A}^{-1} \equiv 77 \pmod{96}$. Adjoin untuk matriks $(\boldsymbol{A})$ adalah

$$\text{adj}(\boldsymbol{A}) \equiv \begin{bmatrix} 8 & 89 & 8 & 23 \\ 20 & 32 & 36 & 93 \\ 41 & 64 & 77 & 54 \\ 81 & 27 & 28 & 68 \end{bmatrix} \pmod{96}.$$

Setelah diperoleh invers determinan matriks $\boldsymbol{A}$ dan adjoin matriks $\boldsymbol{A}$kemudian mencari invers matriks $\boldsymbol{A}$ (mod 96) yaitu $\boldsymbol{A}^{-1} \equiv \det(\boldsymbol{A})^{-1} \cdot \text{adj}(\boldsymbol{A}) \pmod{96}$.

$$\boldsymbol{A}^{-1} \equiv 77 \cdot \begin{bmatrix} 8 & 89 & 8 & 23 \\ 20 & 32 & 36 & 93 \\ 41 & 64 & 77 & 54 \\ 81 & 27 & 28 & 68 \end{bmatrix} \pmod{96} \equiv \begin{bmatrix} 40 & 37 & 40 & 43 \\ 4 & 64 & 84 & 57 \\ 85 & 32 & 73 & 30 \\ 93 & 63 & 44 & 52 \end{bmatrix} \pmod{96}.$$

Sebelum proses dekripsi mengubah pesan ke dalam kode ASCII sehingga diperoleh . = 46,` = 96, @ = 64, A = 65, DEL = 127, & = 38, V = 86, S = 83, " = 34, e = 101, [ = 91, 2 = 50, a 97, ? = 63, ) 41, dan ` = 96. Masukkan pesan yang sudah diubah ke dalam kode ASCII pada matriks *ciphertext* $\boldsymbol{C}$ yang berordo 4 × 4 supaya sesuai modulo 96 maka setiap entri dikurangi 32 sehingga diperoleh:

$$\boldsymbol{C} = \begin{bmatrix} 14 & 64 & 32 & 33 \\ 95 & 6 & 54 & 51 \\ 2 & 69 & 59 & 18 \\ 65 & 31 & 9 & 64 \end{bmatrix}.$$

Selanjutnya proses dekripsi *ciphertext* yaitu $\boldsymbol{P}_i \equiv \boldsymbol{A}^{-1}(\boldsymbol{C}_i - \boldsymbol{B}) \pmod{m}$. Untuk dekripsi, *ciphertext* dipotong menjadi matriks-matriks berordo 4×1 yaitu $\boldsymbol{C}_1 = \begin{bmatrix} 14 \\ 95 \\ 2 \\ 65 \end{bmatrix}, \boldsymbol{C}_2 = \begin{bmatrix} 64 \\ 6 \\ 69 \\ 31 \end{bmatrix}, \boldsymbol{C}_3 = \begin{bmatrix} 32 \\ 54 \\ 59 \\ 9 \end{bmatrix}$, dan $\boldsymbol{C}_4 = \begin{bmatrix} 33 \\ 51 \\ 18 \\ 65 \end{bmatrix}$ sehingga diperoleh:

$$\boldsymbol{P}_1 \equiv \begin{bmatrix} 40 & 37 & 40 & 43 \\ 4 & 64 & 84 & 57 \\ 85 & 32 & 73 & 30 \\ 93 & 63 & 44 & 52 \end{bmatrix} \cdot \left(\begin{bmatrix} 14 \\ 95 \\ 2 \\ 65 \end{bmatrix} - \begin{bmatrix} 87 \\ 97 \\ 105 \\ 115 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 48 \\ 78 \\ 72 \\ 65 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_2 \equiv \begin{bmatrix} 40 & 37 & 40 & 43 \\ 4 & 64 & 84 & 57 \\ 85 & 32 & 73 & 30 \\ 93 & 63 & 44 & 52 \end{bmatrix} \cdot \left(\begin{bmatrix} 64 \\ 6 \\ 69 \\ 31 \end{bmatrix} - \begin{bmatrix} 87 \\ 97 \\ 105 \\ 115 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 69 \\ 0 \\ 65 \\ 0 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_3 \equiv \begin{bmatrix} 40 & 37 & 40 & 43 \\ 4 & 64 & 84 & 57 \\ 85 & 32 & 73 & 30 \\ 93 & 63 & 44 & 52 \end{bmatrix} \cdot \left(\begin{bmatrix} 32 \\ 54 \\ 59 \\ 9 \end{bmatrix} - \begin{bmatrix} 87 \\ 97 \\ 105 \\ 115 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 83 \\ 82 \\ 83 \\ 0 \end{bmatrix} \pmod{96}$$

$$\boldsymbol{P}_4 \equiv \begin{bmatrix} 40 & 37 & 40 & 43 \\ 4 & 64 & 84 & 57 \\ 85 & 32 & 73 & 30 \\ 93 & 63 & 44 & 52 \end{bmatrix} \cdot \left(\begin{bmatrix} 33 \\ 51 \\ 18 \\ 64 \end{bmatrix} - \begin{bmatrix} 87 \\ 97 \\ 105 \\ 115 \end{bmatrix} \right) \pmod{96} \equiv \begin{bmatrix} 65 \\ 65 \\ 73 \\ 0 \end{bmatrix} \pmod{96}$$

Sehingga diperoleh $\boldsymbol{P} = \begin{bmatrix} 48 & 69 & 83 & 65 \\ 78 & 0 & 82 & 65 \\ 72 & 65 & 83 & 73 \\ 65 & 0 & 0 & 0 \end{bmatrix}$. Setelah diperoleh hasil dekripsi $\boldsymbol{P}$ kemudian ditambah dengan 32 supaya memenuhi karakter 32 sampai 127 menjadi $P = 48 + 32 = 80, 69 + 32 = 101, 83 + 32 = 115, 65 + 32 = 97, 78 + 32 = 110,\ 0 + 32 = 32,\ 82 + 32 = 114,\ 65 + 32 = 97, 72 + 32 = 104, 65 + 32 = 97,\ 83 + 32 = 115, 73 + 32 = 105, 65 + 32 = 97, 0 + 32 = 32,\ 0 + 32 = 32,$ dan $0 + 32 = 32$. Dengan kode ASCII diperoleh karakter aslinya menjadi C =Pesan rahasia .

Pada proses enkripsi dari matriks $2 \times l$, $3 \times l$ dan $4 \times l$ diperoleh bahwa setelah dienkripsi semakin besar ukuran matriks maka perubahan karakter akan semakin besar. Setiap karakter mempunyai lebih dari satu perubahan karakter setelah dienkripsi. Ini menunjukkan bahwa semakin besar ukuran matriks maka tingkat keamanannya semakin kuat. Dari sini dapat disimpulkan bahwa tingkat keamanan pesan bergantung pada seberapa besar ukuran matriks yang digunakan.

3.1.6 Analisis Hasil Implementasi

3.1.6.1 Keunggulan Algoritma RSA dan *Affine Cipher* Menggunakan Metode Matriks

Pada algoritma *affine cipher* perubahan setiap karakter setelah enkripsi hanya mempunyai satu perubahan karakter. Karena setiap karakter hanya mempunyai satu perubahan karakter maka kriptaanalis akan mudah menebak kunci yang digunakan dengan cara analisis frekuensi, yaitu menebak karakter yang sering muncul dengan statistik penggunaan karakter terbesar untuk menebak kunci enkripsi. Dengan menggunakan metode matriks maka setiap karakter mempunyai lebih dari satu perubahan karakter setelah dienkripsi. Hal ini mengakibatkan analisis frekuensi sulit untuk digunakan atau bahkan tidak dapat digunakan untuk menebak kunci enkripsi. Meskipun kriptaanalis menggetahui beberapa potongan pesan asli tetapi tidak mudah menentukan dua pasangan kunci yang digunakan untuk menghasilkan perameter yang sesuai. Terkadang harus mencoba pasangan lain untuk mendapatkan perameter yang sesuai.

Selain pada perubahan karakter, kunci algoritma *affine cipher* terbatas pada banyaknya bilangan yang relatif prima dengan modulo yang digunakan dan bilangan kurang dari modulo yang digunakan. Untuk menentukan parameter yang sesuai maka kriptaanalisis dapat menggunakan pasangan kunci yang diperoleh.

Dengan menggunakan metode matriks maka kunci yang digunakan tidak terbatas hanya bergantung ukuran matriks dan determinannya harus relatif prima dengan modulo yang digunakan.

Sebagai tambahan keamanan digunakan algoritma RSA. Algoritma RSA digunakan untuk mengamankan kunci sesi ketika akan dikirim ke penerima. Dengan mengenkripsi kunci sesi menggunakan kunci publik maka kunci yang dikirim akan terjaga keamanannya. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan bulat besar untuk mendapatkan faktor primanya. Semakin besar bilangan prima yang digunakan maka keamanannya semakin kuat.

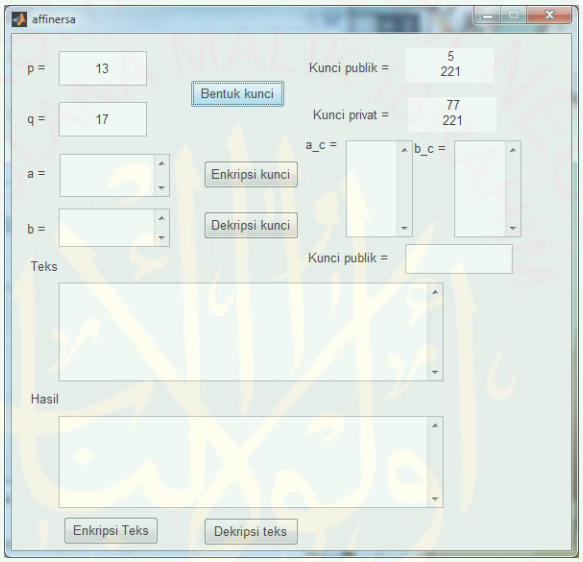
3.1.6.2 Kelemahan Algoritma RSA dan *Affine Cipher* Menggunakan Metode Matriks

Pada algoritma *affine cipher* proses enkripsi dan dekripsi sangat cepat dan tidak membutuhkan waktu lama. Dengan menggunakan metode matriks maka proses enkripsi dan dekripsi akan lebih lama bergantung ukuran matriks. Semakin besar ukuran matriks maka semakin lama proses enkripsi dan dekripsi. Pada determinan kuncinya harus relatif prima dengan modulo yang digunakan, sehingga jika tidak memenuhi maka harus mencari kunci lain sehingga determinannya relatif prima dengan modulo yang digunakan.

Dengan ditambahkannya algoritma RSA maka lama proses enkripsi dan dekripsi akan bertambah. Ini karena pengirim atau penerima tidak hanya mengenkripsi atau mendekripsi pesan saja tetapi termasuk kunci sesi. Sebagai algoritma asimetri, RSA mempunyai kelemahan yaitu lama dalam proses enkripsi maupun dekripsi. Semakin besar bilangan prima yang digunakan maka semakin lama proses enkripsi maupun dekripsi.

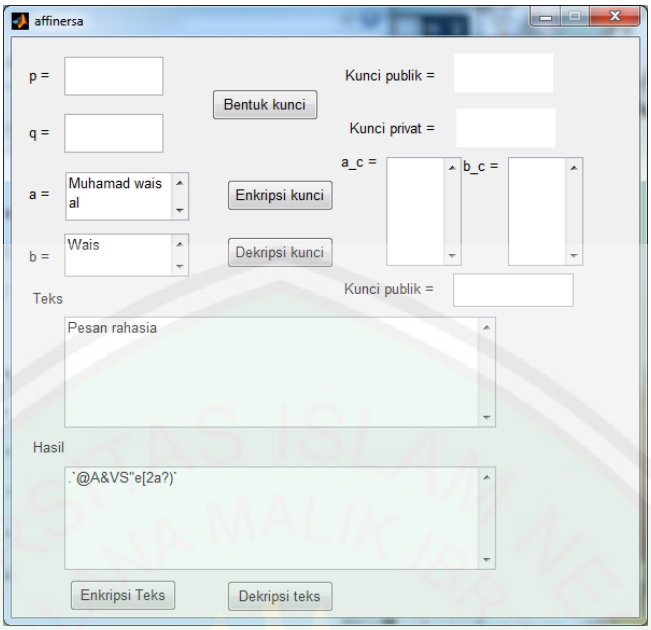
3.1.7 Simulasi Proses Pembentukan Kunci Publik, Kunci Privat, Enkripsi, dan Dekripsi Pesan dengan GUI MATLAB

Pada bab ini simulasi dilakukan dengan menggunakan aplikasi *Graphical User Interface* (GUI) di MATLAB R2013a. Proses pertama yang dilakukan adalah penerima pesan akan membangkitkan kunci publik dan kunci privat. Dengan memasukkan dua bilangan prima p dan q dihasilkan kunci publik dan kunci privat seperti pada Gambar 3.2.



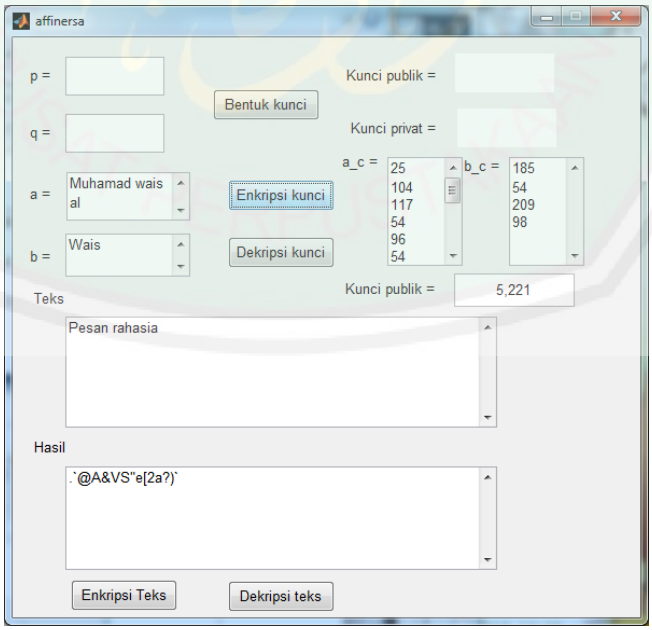
Gambar 3.2 Pembentukan Kunci Publik dan Kunci Privat

Setelah diperoleh kunci publik dan kunci privat, penerima pesan mengirim kunci publik kepada penerima pesan untuk mengenkripsi kunci sesi. Selanjutnya pengirim pesan memilih kunci a dan kunci b dan memasukkan pesan yang akan dienkripsi pada kolom teks. Untuk mendapatkan pesan terenkripsi tekan tombol dekripsi dan menghasilkan *ciphertext* seperti pada Gambar 3.3.



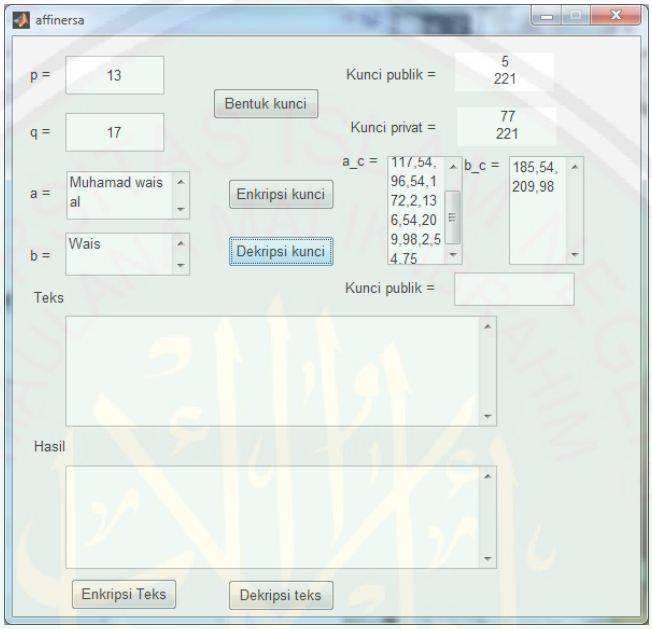
Gambar 3.3 Pengirim Pesan Mengenkripsi Pesan

Selanjutnya proses enkripsi kunci sesi. Pada enkripsi kunci sesi, kunci publik yang diterima dari penerima pesan dimasukkan ke kolom kunci publik. Setelah dimasukkan kemudian tekan tombol enkripsi kunci untuk mendapatkan kunci sesi terenkripsi seperti pada Gambar 3.4.



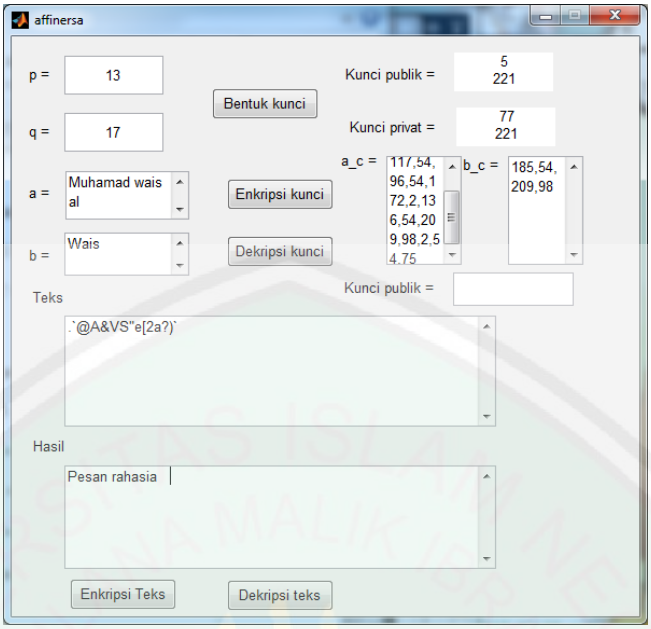
Gambar 3.4 Pengirim Mengenkripsi Kunci Sesi

Hasil akhir diperoleh pesan dan kunci terenkripsi. Pesan dan kunci terenkripsi kemudian dikirim ke penerima pesan. Untuk mendapatkan pesan yang terbaca penerima pesan melakukan proses dekripsi pesan dan kunci. Proses pertama yang dilakukan adalah mendekripsikan kunci sesi terenkripsi seperti pada Gambar 3.5.



Gambar 3.5 Penerima Mendekripsi Kunci Sesi Terenkripsi

Setelah diperoleh kunci sesi maka proses terakhir adalah mendekripsikan pesan. Pesan terenkripsi yang diterima dimasukkan pada kolom teks kemudian tekan tombol dekripsi teks untuk mendapatkan pesan asli seperti pada Gambar 3.6.



Gambar 3.6 Hasil Dekripsi Pesan Menghasilkan Pesan Asli

Hasil akhir penelitian ini dapat membuktikan bahwa perhitungan secara manual dan secara program diperoleh hasil yang sama.

3.2 Penerapan Tentang Berpesan dan Adil dalam Islam

Dalam menyampaikan amanat atau pesan kita dilarang untuk menghianatinya, seperti yang disampaikan Allah Swt dalam al-Quran surat al-Anfal/8:27 yaitu:

يَٰٓأَيُّهَا ٱلَّذِينَ ءَامَنُواْ لَا تَخُونُواْ ٱللَّهَ وَٱلرَّسُولَ وَتَخُونُوٓاْ أَمَٰنَٰتِكُمۡ وَأَنتُمۡ تَعۡلَمُونَ ٢٧

''Hai orang-orang yang beriman, janganlah kamu mengkhianati Allah dan Rasul (Muhammad) dan (juga) janganlah kamu mengkhianati amanat-amanat yang dipercayakan kepadamu, sedang kamu mengetahui'' (QS. al-Anfal/8:27).

Ali bin Abi Thalib berkata, dari Ibnu Abbas berkenaan dengan firman Allah "*dan (juga) janganlah kamu mengkhianati amanat-amanat yang dipercayakan kepadamu*". Amanat adalah segala macam amal perbuatan yang diamanatkan Allah Swt Kepada hamba-hamba-Nya. Maksudnya adalah kewajiban, Dia

berkata: "*Jangan berkhianat*", maksudnya adalah jangan melanggar amanat itu (Katsir, 2003).

Di dalam al-Quran Allah Swt pernah beramanat kepada langit, bumi, dan gunung-gunung namun mereka tidak sanggup menerima amanat tersebut. Sebagaimana yang tercantum dalam al-Quran surat al-Ahzab/33:72:

إِنَّا عَرَضْنَا ٱلْأَمَانَةَ عَلَى ٱلسَّمَٰوَٰتِ وَٱلْأَرْضِ وَٱلْجِبَالِ فَأَبَيْنَ أَن يَحْمِلْنَهَا وَأَشْفَقْنَ مِنْهَا وَحَمَلَهَا ٱلْإِنسَٰنُ ۖ إِنَّهُۥ كَانَ ظَلُومًا جَهُولًا ﴿٧٢﴾

"*Sesungguhnya Kami telah mengemukakan amanat[1233] kepada langit, bumi dan gunung-gunung, Maka semuanya enggan untuk memikul amanat itu dan mereka khawatir akan mengkhianatinya, dan dipikullah amanat itu oleh manusia. Sesungguhnya manusia itu Amat zalim dan Amat bodoh*" *(QS. al-Ahzab/33:72).*

Al-Aufi telah meriwayatkan dari Ibnu Abbas, bahwa yang dimaksud dengan amanat adalah ketaatan. Allah menawarkan amanat itu kepada mereka sebelum menawarkannya kepada manusia, tetapi ternyata mereka tidak kuat. Lalu Allah berfirman kepada Adam, "*Sesungguhnya Aku telah menawarkan amanat ini kepada langit, bumi, dan gunung-gunung, tetapi mereka tidak mampu memikulnya. Apakah kamu mau memikul amanat ini berikut segala akibatnya?*" Adam bertanya, "*Apa saja konsekuensinya itu, wahai Tuhanku?*" Allah Swt menjawab, "*Jika kamu berbuat baik, maka kamu diberi pahala. Dan jika kamu berbuat buruk, kamu disiksa*." Lalu amanat itu diambil oleh Adam. Yang demikian itu disebutkan oleh firman-Nya: "*dan dipikullah amanat itu oleh manusia. Sesungguhnya manusia itu amat zalim dan amat bodoh."*

Imam Ahmad berkata dari 'Abdullah bin 'Amr, Bahwa Rasulullah Saw bersabda:

أَرْبَعٌ إِذَا كُنَّ فِيكَ فَلَا عَلَيْكَ مَا فَاتَكَ مِنَ الدُّنْيَا: حِفْظُ أَمَانَةٍ، وَصِدْقُ حَدِيثٍ، وَحُسْنُ خَلِيقَةٍ، وَعِفَّةُ طُعْمَةٍ

"*Empat hal, jika ada pada dirimu, maka tidak berbahaya bagimu apa yang hilang dari dunia; menjaga amanat, jujur dalam tutur kata, baik akhlak, dan baik dalam makanan*." (Demikian yang diriwayatkan oleh Imam Ahmad dalam musnadnya) (Katsir, 2003).

Telah ada pula larangan bersumpah dengan amanat. Dalam hal ini terdapat hadits marfu'. Abu Dawud meriwayatkan dari Ibnu Buraidah, bahwa ayahnya berkata Rasulullah Saw bersabda:

من حَلَفَ بِالْأَمَانَةِ فَلَيْسَ مِنَّا

"*Barang siapa yang bersumpah dengan amanat, maka bukan termasuk golongan kami*." (Abu Dawud meriwayatkannya sendiri).

Berkaitan dengan perintah adil Allah Swt telah menyampaikan kepada Rasulullah Saw, yaitu terdapat di dalam al-Quran surat al-Maa-idah /5:42:

سَمَّٰعُونَ لِلْكَذِبِ أَكَّٰلُونَ لِلسُّحْتِ ۚ فَإِن جَآءُوكَ فَٱحْكُم بَيْنَهُمْ أَوْ أَعْرِضْ عَنْهُمْ ۖ وَإِن تُعْرِضْ عَنْهُمْ فَلَن يَضُرُّوكَ شَيْـًٔا ۖ وَإِنْ حَكَمْتَ فَٱحْكُم بَيْنَهُم بِٱلْقِسْطِ ۚ إِنَّ ٱللَّهَ يُحِبُّ ٱلْمُقْسِطِينَ ﴿٤٢﴾

"*Mereka itu adalah orang-orang yang suka mendengar berita bohong, banyak memakan yang haram. jika mereka (orang Yahudi) datang kepadamu (untuk meminta putusan), maka putuskanlah (perkara itu) diantara mereka, atau berpalinglah dari mereka; jika kamu berpaling dari mereka maka mereka tidak akan memberi mudharat kepadamu sedikitpun. dan jika kamu memutuskan perkara mereka, Maka putuskanlah (perkara itu) diantara mereka dengan adil, Sesungguhnya Allah menyukai orang-orang yang adil*." *(QS. al-Maa-idah /5:42).*

Allah Swt berfirman kepada Nabi-Nya "*jika mereka (orang Yahudi) datang kepadamu (untuk meminta putusan), maka putuskanlah (perkara itu) diantara mereka, atau berpalinglah dari mereka; jika kamu berpaling dari mereka maka mereka tidak akan memberi mudharat kepadamu sedikitpun*". Maksudnya, engkau tidak bersalah jika memberikan keputusan di antara, karena tujuan mereka berhukum (meminta keputusan) kepadamu itu bukan untuk

mengikuti kebenaran, tetapi untuk mencari hal yang sesuai dengan hawa nafsu mereka.

Ibnu ‘Abbas, Mujahid, Ikrimah, al-Hasan, Qatadah, as-Suddi, Zaid bin Aslam, ‘Atha’ al-Khurasani, al-Hasan dan beberapa ulama lainnya mengatakan “Ayat tersebut dinaskh dengan firman Allah Swt “*dan jika kamu memutuskan perkara mereka, Maka putuskanlah (perkara itu) di antara mereka dengan adil.*” Yakni dengan hak dan adil, meskipun mereka adalah orang-orang zalim yang keluar dari jalan keadilan. “*Sesungguhnya Allah menyukai orang-orang yang adil.*” (Katsir, 2003).

BAB IV

PENUTUP

4.1 Kesimpulan

Berdasarkan pembahasan di atas, dapat disimpulkan bahwa keamanan enkripsi menggunakan algoritma RSA dan *affine cipher* dengan metode matriks terletak pada keamanan kunci simetri sedangkan terletak pada keamanan kunci asimetri. Keamanan kunci simetri terletak pada pasangan kunci berupa matriks persegi dan matriks kolom. Untuk menebak kuncinya membutuhkan dua pasangan kunci yang digunakan untuk menghasilkan perameter yang sesuai. Jika belum sesuai maka menggunakan pasangan lain untuk mendapatkan perameter yang sesuai. Selain itu perubahan setiap karakter setelah proses enkripsi mempunyai lebih dari satu perubahan karakter sehingga tidak mudah untuk menebak karakter sebenarnya. Semakin besar ukuran matriks yang digunakan maka semakin aman. Keamanan kunci asimetri terletak pada sulitnya memfaktorkan bilangan bulat besar untuk menentukan faktor primanya. RSA terkenal sebagai algoritma yang sulit untuk menentukan kunci privat. Semakin besar bilangan prima yang digunakan maka keamanannya semakin kuat.

4.2 Saran

Pada penelitian ini membahas mengenai proses enkripsi dan dekripsi pesan menggunakan algoritma *affine cipher* dengan metode matriks sedangkan kunci sesinya diamankan menggunakan algoritma RSA. Pada penelitian selanjutnya disarankan menggunakan metode lain dalam mengamankan pesan.

DAFTAR RUJUKAN

Andrianto, H. dan Prijono, A. 2006. *Menguasai Matriks dan Vektor*. Bandung: Rekayasa Sains.

Anton, H. dan Rorres, C. 2010. *Elementary Linear Algebra Aplications Version*. New Jersey: John Wiley & Sons, Inc.

Ariyus, D. 2008. *Pengantar Ilmu Kriptografi*. Jogjakarta: Andi.

Hamzah, R. 2011. *Implementasi Algoritma RSA dan Blowfish untuk Enkripsi dan Dekripsi Data Menggunakan Delphi 7*. Jakarta: Universitas Islam Syarif Hidayatullah Jakarta.

Irawan, W.H., Hijriyah, N., dan Habibi, A.R. 2014. *Pengantar Teori Bilangan*. Malang: UIN-MALIKI Press.

Katsir, I. 2003. *Tafsir Ibnu Katsir Jilid 2*. Terjemahan M. Abdul Ghoffar E.M. Bogor: Pustaka Imam as-Syafi'i.

Khudzaifah, M. 2014. Aplikasi QuasiGroup dalam Pembentukan Kunci Rahasia pada Algoritma Hibrida (RSA-QuasiGroup Chiper). *Cauchy* , 3(2): 55-58.

Kromodimoeljo, S. 2010. *Teori dan Aplikasi Kriptografi*. Jakarta: SPK IT Consuling.

Lipschutz, S. dan Lipson, M.L. 2009. *SCHAUM'S Outlines Linear Algebra Fourth Edition*. New York: McGraw-Hill Companies, Inc.

Mollin, R.A. 2007. *An Introduction to Cryptography Second Edition*. London: CRC.

Muhsetyo, G. 1997. *Dasar-Dasar Teori Bilangan*. Jakarta: Departemen Pendidikan dan Kebudayaan Direktorat Jendral Pendidikan Tinggi Proyek Pengembangan Guru Sekolah Menengah.

Munir, R. 2004. *Sistem Kriptografi Kunci-Publik. Diktat Kuliah*. Bandung: Departemen Teknik Informatika Institut Teknologi Bandung.

Purwanto, H., Indriani, G., dan Dayanti, E. 2005. *Aljabar Linier*. Jakarta: PT. Ercontara Rajawali.

Rosen, K.H. 2012. *Discrete Mathematics and Its Applications Seventh Edition*. New York: McGraw-Hill.

Stallings, W. 2005. *Cryptography and Network Security Fourt Edition Principles and Practices.* New Jersey: Prentice Hall.

Wibowo, S., Nilawati, F.E., dan Suharnawi. 2014. Implementasi Enkripsi Dekripsi Algoritma Affine Cipher Berbasis Android. *Techno.COM* , 13(4): 215-221.

LAMPIRAN

Lampiran 1. Kode Program Pembentukkan Kunci, Enkripsi, dan Dekripsi

```
function varargout = affinersa(varargin)
% AFFINERSA MATLAB code for affinersa.fig
%      AFFINERSA, by itself, creates a new AFFINERSA or raises the existing
%      singleton*.
%
%      H = AFFINERSA returns the handle to a new AFFINERSA or the handle to
%      the existing singleton*.
%
%      AFFINERSA('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in AFFINERSA.M with the given input
arguments.
%
%      AFFINERSA('Property','Value',...) creates a new AFFINERSA or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before affinersa_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to affinersa_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help affinersa

% Last Modified by GUIDE v2.5 12-Nov-2017 22:12:18

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @affinersa_OpeningFcn, ...
                   'gui_OutputFcn',  @affinersa_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
```

```
end
% End initialization code - DO NOT EDIT

% --- Executes just before affinersa is made visible.
function affinersa_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.%...%

% Choose default command line output for affinersa
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes affinersa wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = affinersa_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT); %...%

% Get default command line output from handles structure
varargout{1} = handles.output;

function p_Callback(hObject, eventdata, handles)
% hObject    handle to p (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of p as text
%        str2double(get(hObject,'String')) returns contents of p as a double
% --- Executes during object creation, after setting all properties.
function p_CreateFcn(hObject, eventdata, handles)
% hObject    handle to p (see GCBO) %...%

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end

function q_Callback(hObject, eventdata, handles)
% hObject    handle to q (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of q as text
%        str2double(get(hObject,'String')) returns contents of q as a double


% --- Executes during object creation, after setting all properties.
function q_CreateFcn(hObject, eventdata, handles)
```

```
% hObject    handle to q (see GCBO) %...%

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end

function en_Callback(hObject, eventdata, handles)
% hObject    handle to en (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of en as text
%        str2double(get(hObject,'String')) returns contents of en as a double

% --- Executes during object creation, after setting all properties.
function en_CreateFcn(hObject, eventdata, handles)
% hObject    handle to en (see GCBO) %...%

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end

function dn_Callback(hObject, eventdata, handles)
% hObject    handle to dn (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of dn as text
%        str2double(get(hObject,'String')) returns contents of dn as a double

% --- Executes during object creation, after setting all properties.
function dn_CreateFcn(hObject, eventdata, handles)
% hObject    handle to dn (see GCBO) %...%

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end

function a_Callback(hObject, eventdata, handles)
% hObject    handle to a (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of a as text
%        str2double(get(hObject,'String')) returns contents of a as a double
```

```
% --- Executes during object creation, after setting all properties.
function a_CreateFcn(hObject, eventdata, handles)
% hObject    handle to a (see GCBO) %...%

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end
function ac_Callback(hObject, eventdata, handles)
% hObject    handle to ac (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of ac as text
%        str2double(get(hObject,'String')) returns contents of ac as a double

% --- Executes during object creation, after setting all properties.
function ac_CreateFcn(hObject, eventdata, handles)
% hObject    handle to ac (see GCBO) %...%

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end
% --- Executes during object creation, after setting all properties.
function bc_CreateFcn(hObject, eventdata, handles)
% hObject    handle to bc (see GCBO) %...%

function b_Callback(hObject, eventdata, handles)
% hObject    handle to b (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of b as text
%        str2double(get(hObject,'String')) returns contents of b as a double

% --- Executes during object creation, after setting all properties.
function b_CreateFcn(hObject, eventdata, handles)
% hObject    handle to b (see GCBO) %...%

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end
```

```
function teks_Callback(hObject, eventdata, handles)
% hObject    handle to teks (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of teks as text
%        str2double(get(hObject,'String')) returns contents of teks as a double

% --- Executes during object creation, after setting all properties.
function teks_CreateFcn(hObject, eventdata, handles)
% hObject    handle to teks (see GCBO) %...%

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end

function hasil_Callback(hObject, eventdata, handles)
% hObject    handle to hasil (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of hasil as text
%        str2double(get(hObject,'String')) returns contents of hasil as a double


% --- Executes during object creation, after setting all properties.
function hasil_CreateFcn(hObject, eventdata, handles)
% hObject    handle to hasil (see GCBO) %...%

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end


% --- Executes on button press in kunci.
function kunci_Callback(hObject, eventdata, handles)
% hObject    handle to kunci (see GCBO) %...%
p=get(figure1.p,'String');
p=str2double(p);
figure2=guidata(gcbo);
q=get(figure2.q,'String');
q=str2double(q);
Pk=p*q;
Phi=(p-1)*(q-1);
x=2;e=1;
while x > 1
```

```
    e=e+1;
    x=gcd(Phi,e);
end
i=1;
r=1;
while r > 0
    k=(Phi*i)+1;
    r=rem(k,e);
    i=i+1;
end
d=k/e;
set(handles.en,'String',{e,Pk});
set(handles.dn,'String',{d,Pk});
clear clear p q Phi
save edn.mat



% --- Executes on button press in enkripsik.
function enkripsik_Callback(hObject, eventdata, handles)
% hObject    handle to enkripsik (see GCBO) %...%
figure1=guidata(gcbo);
a=get(figure1.a,'String');
figure2=guidata(gcbo);
b=get(figure2.b,'String');
figure3=guidata(gcbo);
ene=get(figure3.ene,'String');
ene=str2num(ene);
e=ene(1:1);
Pk=ene(2:2);
x=length(a);
c=0;
for j= 1:x
    for i=0:255
        if strcmp(a(j),char(i))
            c(j)=i;
        end
    end
end
for j= 1:x
  cipher(j)= crypt(c(j),Pk,e);
end
ac=cipher;
x=length(b);
c=0;
for j= 1:x
    for i=0:255
        if strcmp(b(j),char(i))
```

```
            c(j)=i;
        end
    end
end
bc=c;
for j= 1:x
   cipherb(j)= crypt(bc(j),Pk,e);
end
bc=cipherb;
set(handles.ac,'String',(ac));
set(handles.bc,'String',(bc));


% --- Executes on button press in dekripsik.
function dekripsik_Callback(hObject, eventdata, handles)
% hObject    handle to dekripsik (see GCBO) %...%
load edn.mat
figure1=guidata(gcbo);
ac=get(figure1.ac,'String');
ac=str2num(ac);
figure2=guidata(gcbo);
bc=get(figure2.bc,'String');
bc=str2num(bc);
x=length(ac);
for i=1:x
    aa(i)=ac(i:i);
end
ac=aa;
for j= 1:x
   cipher(j)= crypt(ac(j),Pk,d);
end
kuncia=cipher;
x=length(bc);
for i=1:x
    aa(i)=bc(i:i);
end
bc=aa;
for j= 1:x
   cipherb(j)= crypt(bc(j),Pk,d);
end
kuncib=cipherb;
set(handles.a,'String',char(kuncia));
set(handles.b,'String',char(kuncib));

% --- Executes on button press in dekrpsit.
function dekrpsit_Callback(hObject, eventdata, handles)
% hObject    handle to dekrpsit (see GCBO) %...%
figure1=guidata(gcbo);
```

```
a=get(figure1.a,'String');
figure2=guidata(gcbo);
b=get(figure2.b,'String');
figure3=guidata(gcbo);
teks=get(figure3.teks,'String');
x=length(a);
c=0;
for j= 1:x
   for i=0:255
      if strcmp(a(j),char(i))
         c(j)=i;
      end
   end
end
k=sqrt(x);
l=ceil(k);
if rem(k,l)==0
   n=k;
else
   n=l;
   j=x+1:l^2;
   c(j)=32;
end
ak=zeros(n);
for e=1:n;
   for f=1:n;
      ak(e,f)=c(f+(e-1)*n);
   end
end
kuncia=ak;
deta=round(mod((det(kuncia)),96));
invdeta=invmodn(deta,96);
adjkuncia=double(mod(adjoint(sym(kuncia)),96));
invkuncia=mod(invdeta*adjkuncia,96);
t=length(teks);
c=0;
for j= 1:t
   for i=0:255
      if strcmp(teks(j),char(i))
         c(j)=i;
      end
   end
end
r=t/n;
s=ceil(r);
if rem(s,r)==0
   u=r;
else
```

```
    u=s;
    j=t+1:s*n;
    c(j)=32;
end
ac=c;
ak=zeros(n,u);
for e=1:n;
    for f=1:u;
        ak(e,f)=ac(f+(e-1)*s);
    end
end
teks=ak-32;
y=length(b);
if y>n
     msgbox('panjang kunci b harus <= panjang matriks kunci a')
     return
end
c=0;
for j= 1:y
    for i=0:255
        if strcmp(b(j),char(i))
            c(j)=i;
        end
    end
end
if y==n
    n=y;
else
    n=n;
    j=y+1:n;
    c(j)=32;
end
ak=zeros(n,u);
for i=1:u
    ak(:,i)=c(1,:);
end
kuncib=ak;
hasil=mod(invkuncia*(teks-kuncib),96);
hasil=hasil+32;
ka=zeros(1,u*n);
for e=1:n;
    for f=1:u;
        ka(f+(e-1)*s)=hasil(e,f);
    end
end
hasil=ka;
set(handles.hasil,'String',char(hasil));
```

```
% --- Executes on button press in enkripsit.
function enkripsit_Callback(hObject, eventdata, handles)
% hObject    handle to enkripsit (see GCBO) %...%
figure1=guidata(gcbo);
a=get(figure1.a,'String');
figure2=guidata(gcbo);
b=get(figure2.b,'String');
figure3=guidata(gcbo);
teks=get(figure3.teks,'String');
x=length(a);
c=0;
for j= 1:x
   for i=0:255
      if strcmp(a(j),char(i))
         c(j)=i;
      end
   end
end
k=sqrt(x);
l=ceil(k);
if rem(k,l)==0
   n=k;
else
   n=l;
   j=x+1:l^2;
   c(j)=32;
end
ak=zeros(n);
for e=1:n;
   for f=1:n;
      ak(e,f)=c(f+(e-1)*n);
   end
end
kuncia=ak;
ad=round(det(kuncia))
if gcd(ad,96)~=1
   msgbox('determinan a harus relatif prima dengan 96')
   return
end
t=length(teks);
c=0;
for j= 1:t
   for i=0:255
      if strcmp(teks(j),char(i))
         c(j)=i;
      end
   end
end
```

```
r=t/n;
s=ceil(r);
if rem(s,r)==0
   u=r;
else
   u=s;
   j=t+1:s*n;
   c(j)=32;
end
ac=c;
ak=zeros(n,u);
for e=1:n;
   for f=1:u;
      ak(e,f)=ac(f+(e-1)*s);
   end
end
teks=ak-32;
y=length(b);
if y>n
    msgbox('panjang kunci b harus <= panjang matriks kunci a')
    return
end
c=0;
for j= 1:y
   for i=0:255
      if strcmp(b(j),char(i))
         c(j)=i;
      end
   end
end
if y==n
   n=y;
else
   n=n;
   j=y+1:n;
   c(j)=32;
end
ak=zeros(n,u);
for i=1:u
   ak(:,i)=c(1,:);
end
kuncib=ak;
hasil=mod(kuncia*teks+kuncib,96);
hasil=hasil+32;
ka=zeros(1,u*n);
for e=1:n;
   for f=1:u;
      ka(f+(e-1)*s)=hasil(e,f);
```

```
   end
end
hasil=ka;
set(handles.hasil,'String',char(hasil));


function bc_Callback(hObject, eventdata, handles)
% hObject    handle to bc (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of bc as text
%        str2double(get(hObject,'String')) returns contents of bc as a double



function ene_Callback(hObject, eventdata, handles)
% hObject    handle to ene (see GCBO) %...%

% Hints: get(hObject,'String') returns contents of ene as text
%        str2double(get(hObject,'String')) returns contents of ene as a double


% --- Executes during object creation, after setting all properties.
function ene_CreateFcn(hObject, eventdata, handles)
% hObject    handle to ene (see GCBO) %...%

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end


function mc = crypt(M,N,e)
e=dec2bin(e);
k = 65535;
c  = M;
cf = 1;
cf=mod(c*cf,N);
for i=k-1:-1:1
   c = mod(c*c,N);
   j=k-i+1;
    if e(j)==1
       cf=mod(c*cf,N);
    end
end
mc=cf;
```

```
function a = dec2bin(d)
i=1;
a=zeros(1,65535);
while d >= 2
   r=rem(d,2);
   if r==1
      a(i)=1;
   else
      a(i)=0;
   end
   i=i+1;
   d=floor(d/2);
end
if d == 2
   a(i) = 0;
else
   a(i) = 1;
end
x=[a(16) a(15) a(14) a(13) a(12) a(11) a(10) a(9) a(8) a(7) a(6) a(5) a(4) a(3) a(2)
a(1)]
```

Lampiran 2. Tabel ASCII

No	Kode	No	Kode
0	NUL (*null*)	29	GS (*group separator*)
1	SOH (*start of hearder*)	30	RS (*request to send*)
2	STX (*start of text*)	31	US (*unite separator*)
3	ETX (*end of text*)	32	SP (*space*)
4	EOT (*end of transmission*)	33	!
5	ENQ (*enquiry*)	34	“
6	ACK (*acknowladgment*)	35	#
7	BEL (*bell*)	36	$
8	BS (*bacjspace*)	37	%
9	HT (*horizontal tab*)	38	&
10	LF (*line feed*)	39	‘
11	VT (*vertical tab*)	40	(
12	FF (*form feed*)	41	)
13	CR (*carriage return*)	42	*
14	SO (*shift out*)	43	+
15	SI (*shift in*)	44	,
16	DLE (*data link escape*)	45	-
17	DC1 (*device control* 1)	46	.
18	DC2 (*device control* 2)	47	/
19	DC3 (*device control* 3)	48	0
20	DC4 (*device control* 4)	49	1
21	NAK (*negative acknowladgment*)	50	2
22	SYN (*syncronous idle*)	51	3
23	ETB (*end of trans, block*)	52	4
24	CAN (*cancel*)	53	5
25	EM (*end of medium*)	59	;
26	SUB (*substitute*)	60	<
27	ESC (*escape*)	61	=
28	FS (*file separator*)	62	>

No	Kode	No	Kode
63	?	93	]
64	@	94	^
65	A	95	_
66	B	96	`
67	C	97	a
68	D	98	b
69	E	99	c
70	F	100	d
71	G	101	E
72	H	102	f
73	I	103	g
74	J	104	h
75	K	105	i
76	L	106	j
77	M	107	k
78	N	108	l
79	O	109	m
80	P	110	n
81	Q	111	o
82	R	112	p
83	S	113	q
84	T	114	r
85	U	115	s
86	V	116	t
87	W	117	u
88	X	118	v
89	Y	119	w
90	Z	120	x
91	[	121	y
92	\	122	z

No	Kode
124	|
125	}
126	~
127	DEL (*delete*)

RIWAYAT HIDUP

Muhamad Wais Al Qorny, lahir di Malang 26 Oktober 1993, tinggal di Desa Lumbangsari, Kecamatan Bululawang, Kabupaten Malang. Anak bungsu dari tiga bersaudara, putra dari pasangan bapak Moch. Toha dan ibu Siti Ismariyah.

Pendidikan dasar ditempuh di SDN Lumbangsari 2 dan lulus pada tahun 2006, kemudian melanjutkan pendidikan menengah pertama di SMP Negeri 1 Bululawang dan lulus pada tahun 2009, kemudian melanjutkan pendidikan menengah atas di SMA Negeri 4 Malang dan lulus pada tahun 2012. Selanjutnya menempuh pendidikan tinggi pada tahun 2013 di Universitas Islam Negeri Maulana Malik Ibrahim Malang mengambil Jurusan Matematika Fakultas Sains dan teknologi.

Selama menjadi mahasiswa, dia aktif pada organisasi intra kampus dalam rangka mengembangkan akademiknya. Organisasi intra yang diikutinya antara lain adalah Himpunan Mahasiswa Jurusan (HMJ) Matematika sebagai anggota pada periode 2014/2015, Mathematics English Club (MEC) sebagai anggota pada periode 2015/2016 dan 2016/2017, dan Serambi Matematika Aktif (SEMATA) sebagai ketua periode 2015/2016 dan 2016/2017.

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

BUKTI KONSULTASI SKRIPSI

Nama : Muhamad Wais Al Qorny
NIM : 13610118
Fakultas/ Jurusan : Sains dan Teknologi/ Matematika
Judul Skripsi : Enkripsi dan Dekripsi Pesan Menggunakan Algoritma RSA dan *Affine Cipher* dengan Metode Matriks
Pembimbing I : Dr. Abdussakir, M.Pd
Pembimbing II : Ari Kusumastuti, M.Pd, M.Si

No	Tanggal	HAL	Tanda Tangan
1.	10 Juli 2017	Konsultasi Bab I & Bab II Kajian Keagamaan	1.
2.	01 Agustus 2017	Konsultasi Bab I & Bab II	2.
3.	15 Agustus 2017	Konsultasi Bab III & Revisi Bab I & Bab II	3.
4.	15 November 2017	Revisi Bab I, Bab II & Bab III	4.
5.	29 November 2017	Revisi Bab I & Bab II Kajian Keagamaan	5.
6.	04 Januari 2018	Konsultasi Bab I, Bab II & Bab III	6.
7.	09 Januari 2018	Revisi Bab I, Bab II & Bab III	7.
8.	09 Januari 2018	Konsultasi Bab I, Bab II & Bab III Kajian Keagamaan	8.
9.	18 Januari 2018	Konsultasi Bab I & Bab IV	9.
10.	24 Januari 2018	Revisi Bab I & Bab IV	10.
11.	25 Januari 2018	Revisi Bab I, Bab II & Bab III Kajian Keagamaan	11.
12.	21 Februari 2018	Revisi Bab III Kajian Keagamaan	12.
13.	16 April 2018	Konsultasi Bab I I & Bab III	13.
14.	27 April 2018	Revisi Bab Bab I I & Bab III	14.

Malang, 03 Mei 2018
Mengetahui,
Ketua Jurusan Matematika

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