LAPORAN PENELITIAN KOMPETITIF

TAHUN ANGGARAN 2017

Aplikasi *Quasigroup Encryption* untuk Mengamankan Soal Ujian

Nomor DIPA	:	DIPA BLU: DIPA-025.04.2.423812/2016
Tanggal	:	7 Desember 2017
Satker	:	(423812) UIN Maulana Malik Ibrahim Malang
Kode Kegiatan	:	(2132) Peningkatan Akses, Mutu, Kesejahteraan dan Subsidi Pendidikan Tinggi Islam
Kode Sub Kegiatan	:	(008) Penelitian Bermutu
Kegiatan	:	(004) Dukungan Operasional Penyelenggaraan Pendidikan

OLEH :

Muhammad Khudzaifah, M.Si

(19900511 20160801 1 057)

KEMENTERIAN AGAMA

LEMBAGA PENELITIAN DAN PENGABDIAN KEPADA MASYARAKAT

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG

2017

PERNYATAAN KESANGGUPAN MENYELESAIKAN PENELITIAN

Kami yang bertanda tangan di bawah ini:

Nama	: Muhammad Khudzaifah, M.Si
NIDT	: 19900511 20160801 1 057
Pangkat /Gol.Ruang	: Penata Muda Tk.I / III B
Fakultas/Jurusan	: Sains dan Teknologi/ Matematika
Jabatan dalam Penelitian	: Ketua Peneliti

Dengan ini menyatakan bahwa:

1. Saya sanggup menyelesaikan dan menyerahkan laporan hasil penelitian maksimal pada tanggal 17 Juli 2017.
2. Apabila sampai batas waktu yang ditentukan saya/kami belum menyerahkan laporan hasil, maka saya sanggup mengembalikan dana penelitian yang telah saya terima.

Malang,14 Juli 2017

Ketua Peneliti

Muhammad Khudzaifah, M.Si

NIDT.19900511 20160801 1 057

HALAMAN PENGESAHAN

Laporan Penelitian ini disahkan oleh Lembaga Penelitian dan Pengabdian kepada Masyarakat Universitas Islam Negeri Maulana Malik Ibrahim Malang Pada tanggal 14 Juli 2017

Peneliti

Ketua:
Nama : Muhammad Khudzaifah, M.Si
NIDT : 19900511 20160801 1 057
Tanda Tangan …….

Ketua LP2M UIN Maulana Malik Ibrahim Malang

Dr. Hj. Mufidah Ch., M.Ag.
NIP. 196009101989032001

PERNYATAAN ORISINALITAS PENELITIAN

Kami yang bertanda tangan di bawah ini:

Nama	: Muhammad Khudzaifah, M.Si
NIDT	: 19900511 20160801 1 057
Pangkat /Gol.Ruang	: Penata Muda Tk.I / III B
Fakultas/Jurusan	: Sains dan Teknologi/ Matematika
Jabatan dalam Penelitian	: Ketua Peneliti

Menyatakan dengan sebenar-benarnya bahwa dalam penelitian ini tidak terdapat unsur-unsur penjiplakan karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat oleh orang lain, kecuali yang secara tertulis disebutkan dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka. Apabila dikemudian hari ternyata dalam penelitian ini terbukti terdapat unsur-unsur penjiplakan dan pelanggaran etika akademik, maka kami bersedia mengembalikan dana penelitian yang telah kami terima dan diproses sesuai dengan peraturan perundang-undangan yang berlaku.

Malang,14 Juli 2017

Ketua Peneliti

Muhammad Khudzaifah, M.Si
NIDT.19900511 20160801 1 057

PERNYATAN TIDAK SEDANG TUGAS BELAJAR

Yang bertanda tangan di bawah ini, Saya:

Nama	: Muhammad Khudzaifah, M.Si
NIDT	: 19900511 20160801 1 057
Pangkat /Gol.Ruang	: Penata Muda Tk.I / III B
Tempat, Tanggal Lahir	: Sidoarjo, 11 Mei 1990
Judul Penelitian	: Aplikasi *Quasigroup Encryption* untuk Mengamankan Soal Ujian

dengan ini menyatakan bahwa:

1. Saya TIDAK SEDANG TUGAS BELAJAR
2. Apabila dikemudian hari terbukti bahwa saya sedang tugas belajar, maka secara langsung saya menyatakan mengundurkan diri dan mengembalikan dana yang telah saya terima dari Program Penelitian Kompetitif tahun 2017.

Demikian surat pernyataan ini, Saya buat sebagaimana mestinya.

Malang, 14 Juli 2017

Yang membuat pernyataan,

Muhammad Khudzaifah, M.Si

NIDT.19900511 20160801 1 057

ABSTRAK

Pada penelitian ini dibahas penerapan quasigrup di bidang kriptografi. Suatu operasi quasigrup didefinisikan sehingga bisa membentuk suatu algoritma kriptografi yang disebut sebagai *quasigroup cipher*. *Quasigroup cipher* merupakan algoritma kriptografi simetris. Algoritma kriptografi simetris memiliki sistem keamanan lemah karena kunci yang digunakan untuk proses *enciphering* sama dengan kunci yang digunakan untuk proses *deciphering*, sehingga pada penelitian ini algoritma *quasigroup cipher* dimodifikasi dengan menggabungkannya dengan algoritma RSA menjadi suatu algoritma hibrida yang memiki dua tingkatan kunci untuk mengamankan naskah soal ujian. Maka dihasilkan algoritma yang memiliki tingkat keamanan yang baik dan proses *enciphering* serta *deciphering* membutuhkan waktu yang singkat.

Kata kunci : *quasigrup, kriptografi, algoritma hibrida*.

ABSTRACT

This research discusses application of quasigroup in the field of cryptography. A quasigroup operation is defined so that it can form a cryptographic algorithm called a cipher quasigroup. quasigroup cipher is a symmetric cryptographic algorithms. Symmetric cryptographic algorithms have weak security systems which are used as the key for enciphering process same as the key used for deciphering process, so at this research quasigroup cipher algorithm is modified by combining the RSA algorithm into a hybrid algorithm that is thinking about the two key levels to secure manuscript of exam. So the resulting algorithm that has a good level of security and the process of enciphering and deciphering takes a short time.

Keywords: *quasigroup, cryptography, hybrid algorithm.*

DAFTAR ISI

BAB I PENDAHULUAN 1
BAB II TINJAUAN PUSTAKA 3
2.1 Pemetaan, Permutasi, Operasi Uner dan Operasi Biner 3
2.2 Latin Rectangle dan *Latin Square* 4
2.3 Groupoid 5
2.4 Quasigroup 7
2.5 Kriptografi 9
BAB III HASIL DAN PEMBAHASAN 14
BAB IV KESIMPULAN 26
DAFTAR PUSTAKA 27

BAB I
PENDAHULUAN

1.1 Latar Belakang

Kriptografi memegang peranan penting seiring dengan perkembangan teknologi informasi dan komunikasi. Masalah keamanan merupakan salah satu aspek penting dari sebuah sistem informasi, salah satunya keamanan atau kerahasiaan soal ujian atau dokumen negara yang lainnya yang bersifat sangat rahasia. Apalagi saat ini sedang marak kasus penyadapan terjadi di Indonesia maupun negara lainnya.

Seiring dengan perkembangan teknologi informasi, ancaman peretasan keamanan data pun semakin besar. Hampir semua kegiatan dilakukan dengan serba digital, terutama kegiatan akademik mulai kegiatan belajar mengajar (e-learning) hingga ujian yang saat ini pemerintah menerapkan ujian dalam bentuk UNBK(Ujian Nasional Berbasis Komputer). Ujian bersifat sangat rahasia, oleh karena itu diperlukan pengamanan yang sangat ketat dari segala tindak bentuk kecurangan, terutama peretasan soal ujian yang bisa mengakibatkan soal ujian bisa bocor.

Sudah sangat banyak algoritma yang dikembangkan untuk mengamankan data, ada algoritma simetris dan asimetris, yang membedakan keduanya adalah kunci untuk mensandikan pesan dan kunci untuk menterjemahkan pesan sama untuk algoritma simetris, sedangkan pada algoritma asimetris mempunyai kunci yang berbeda.

Setiap algoritma memiliki keunggulan dan kelemahan, pada algoritma simetris keunggulannya proses penyandian pesan dan menterjemahkan pesan butuh waktu yang singkat tetapi tingkat keamanannya kurang baik karena kunci untuk mensandikan pesan dan menterjemahkan pesan sama, pada algoritma asimetris keunggulannya tingkat keamanannya baik karena kunci untuk mensandikan pesan dan menterjemahkan pesan berbeda tetapi proses penyandian pesan dan menterjemahkan pesan butuh waktu yang lama.

Jika dianalisis, algoritma simetri apabila diterapkan untuk mengamankan soal ujian maka proses penyandian soal ujian dan menterjemahkan soal ujian membutuhkan waktu yang singkat, akan tetapi tingkat keamanannya kurang

baik sehingga rawan peretasan. Jika kita menggunakan algoritma asimetri untuk diterapkan untuk mengamankan soal ujian maka tingkat keamanannya baik, akan tetapi proses penyandian soal ujian dan menterjemahkan soal ujian membutuhkan waktu yang lama sehingga akan memakan waktu ujian yang cukup banyak.

Oleh karena itu dalam penelitian ini saya ingin mengkombinasikan algoritma simetris dan asimetris, agar menghasilkan algoritma yang efektif dan efisien maka algoritma simetris digunakan untuk menyandikan soal ujian dan algoritma asimetris digunakan untuk mengamankan kunci dari algoritma simetris. Sehingga dihasilkan algoritma hibrida yang memiliki tingkat keamanan yang baik dan proses penyandian soal ujian dan penerjemahan soal ujian membutuhkan waktu yang singkat.

Pada tahun 2004 Gligoroski mengembangkan *quasigroup cipher* dengan mendefinisikan *quasigroup order p-1*, dan dikombinasikan dengan Algoritma Elgamal. Dalam penelitian ini akan dibahas penggunaan *quasigroup* pada kriptografi, yang dikombinasikan dengan algoritma RSA untuk membuat suatu program, yang akan mengamankan soal ujian yang dikirim dosen kepada admin untuk dicetak.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang diatas, dapat disusun beberapa rumusan masalah sebagai berikut.

1. Bagaimana proses penyandian menggunakan Algoritma *Quasigroup Encryption*?
2. Bagaimana program komputer sederhana yang digunakan untuk mengamankan soal ujian dengan Algoritma *Quasigroup Encryption*?

1.3 Tujuan Penulisan

Adapun tujuan pengajuan penelitian ini adalah sebagai berikut :

1. Menjelaskan proses penyandian menggunakan Algoritma *Quasigroup Encryption*
2. Menjelaskan program komputer sederhana yang digunakan untuk mengamankan ujian dengan Algoritma *Quasigroup Encryption*

BAB II
TINJAUAN PUSTAKA

Pada Bab ini dibahas materi-materi yang mendukung bagian hasil dan pembahasan. Materi – materi yang dimaksud meliputi pemetaan, permutasi, operasi biner dan uner, *latin rectangle dan latin square, groupoid, quasi-group,* dan kriptografi.

2.1 Pemetaan, Permutasi, Operasi Uner dan Operasi Biner

Suatu sistem aljabar terdiri dari himpunan obyek dengan satu atau lebih operasi yang didefinisikan di dalamnya. Berdasarkan banyaknya elemen sebarang yang terlibat dalam himpunan, operasi pada aljabar dibagi menjadi beberapa bagian salah satunya adalah operasi uner dan biner. Operasi uner merupakan operasi yang hanya melibatkan satu elemen sebarang dalam himpunan, sedangkan operasi biner adalah operasi yang melibatkan dua elemen sebarang dalam himpunan. Berikut definisi lain dari operasi uner dan biner. (Khudzaifah, 2014)

Definisi 2.1.1 (Pemetaan)

Misalkan *A* dan *B* adalah himpunan tidak kosong. Pemetaan f dari *A* ke *B* adalah suatu relasi sedemikian sehingga untuk setiap $a \in A$ tedapat satu $b \in B$ dengan $(a, b) \in f$. Selanjutnya dalam pemetaan dapat dituliskan sebagai $f(a) = b$.

Pada pemetaan f dari *A* ke *B,* himpunan A disebut daerah asal (domain) dari *f* dan himpunan B disebut daerah kawan (kodomain) dari f. Secara umum dikenal dua macam pemetaan yaitu:

(i). f disebut pemetaan satu-satu (injektif) jika untuk setiap $a, b \in A$ dengan $a \neq b$ maka $f(a) \neq f(b)$.

(ii). f disebut pemetaan onto (surjektif) jika untuk setiap $b \in B$ terdapat $a \in A$ sedemikian sehingga $b = f(a)$.

Jika f merupakan pemetaan injektif dan surjektif, maka f disebut sebagai pemetaan bijektif.

(Bhattacharya, dkk., 1994)

Definisi 2.1.2 (Permutasi)

Permutasi dari himpunan tak kosong S adalah pemetaan satu-satu dan onto dari S ke S (Durbin, 2009)

Definisi 2.1.3

Misalkan L adalah sebarang himpunan tak kosong. Suatu fungsi θ dari L disebut operasi uner pada L. Jika θ adalah operasi uner pada himpunan L, maka $\theta\ (l)$ adalah elemen tunggal di dalam L untuk semua $l \in L$.

(Gupta, 2012)

Contoh 2.1.4

Diketahui $\mathbb{N}$ adalah himpunan semua bilangan asli. Misalkan θ adalah fungsi dari $\mathbb{N}$ ke $\mathbb{N}$ yang didefinisikan oleh $\theta\ (n) = -n$ untuk semua $n \in \mathbb{N}$. Maka θ adalah operasi uner pada $\mathbb{N}$.

Definisi 2.1.5

Misalkan *S* adalah himpunan tidak kosong. Operasi biner $*$ pada himpunan *S* adalah pemetaan dari $S \times S$ ke *S*. Untuk setiap $(a, b) \in S \times S$, atau dinotasikan sebagai berikut :

$$*: S \times S \to S$$
$$(a, b) \mapsto * (a, b)\ \epsilon\ S$$
$$= a * b\ \epsilon\ S$$

(Bhattacharya, dkk., 1994)

Contoh 2.1.6

Diketahui $\mathbb{Z}$ adalah himpunan semua bilangan bulat. Didefinisikan operasi $*$ pada $\mathbb{Z}$ dengan syarat untuk setiap $a, b \in \mathbb{Z}$ maka $a * b = a + b$. dapat dilihat dari sifat bilangan bulat bahwa penjumlahan dua bilangan bulat akan selalu menghasilkan bilangan bulat dan tunggal, sehingga dapat dipastikan $a * b = a + b \in \mathbb{Z}$. Dari sini disimpulkan bahwa operasi $*$ merupakan operasi biner pada $\mathbb{Z}$. (Khudzaifah, 2014)

2.2 Latin Rectangle dan *Latin Square*

Nama "*latin square*" terinspirasi dari makalah matematika yang ditulis oleh Leonhard Euler yang menggunakan karakter latin sebagai simbol. Tentu saja simbol dengan karakter latin ini dapat diubah dengan karakter lain seperti angka. *Latin square* memiliki hubungan yang erat dengan *latin rectangle*, yang mana *latin rectangle* dapat dilengkapi menjadi *Latin square.* dan *Latin square* dapat direduksi barisnya menjadi *latin rectangle.* Berikut diberikan definisi mengenai *latin rectangle* dan *Latin square.* (Khudzaifah, 2014)

Definisi 2.2.1

$W_{k\times n}$ adalah notasi untuk *latin rectangle* berukuran $k \times n$ dengan $k, n \in \mathbb{N}$ dan $k < n$. *latin rectangle* adalah matriks dengan k buah baris dan n buah kolom yang berisi elemen-elemen $w_1, \dots, w_n$, sedemikian sehingga tiap-tiap elemen disebutkan sekali dalam tiap baris dan kolom (Al-Turky, 2007)

Contoh 2.2.2

Misal diberikan W adalah himpunan bilangan asli kurang dari 6. Dinotasikan W = {1,2,3,4,5}.

$$W_{2\times 5} = \begin{bmatrix} 3 & 2 & 5 & 1 & 4 \\ 1 & 4 & 3 & 2 & 5 \end{bmatrix}$$

$W_{2\times 5}$ adalah sebuah *latin rectangle* berukuran 2×5 atas himpunan W = {1,2,3,4,5}.■

Definisi 2.2.3

L*atin rectangle* berukuran $k \times n$ disebut *latin square* orde n yang dinotasikan $W_{n\times n}$ jika $k = n$ dengan $k, n \in \mathbb{N}$ (Markovski, dkk., 1997). *latin square* Orde n adalah matriks berukuran $n \times n$ dengan n^2 elemen yang diambil dari suatu himpunan tak kosong W. Elemen – elemen tersebut ditata sedemikian rupa sehingga masing – masing disebutkan sekali dalam tiap kolom dan baris

(Koscienly, 2002)

Contoh 2.2.4

Misal diberikan W adalah himpunan bilangan asli kurang dari 6. Dinotasikan W = {1,2,3,4,5}.

$$W_{5\times 5} = \begin{bmatrix} 3 & 2 & 5 & 1 & 4 \\ 1 & 4 & 3 & 2 & 5 \\ 4 & 1 & 2 & 5 & 3 \\ 5 & 3 & 1 & 4 & 2 \\ 2 & 5 & 4 & 3 & 1 \end{bmatrix}$$

$W_{5\times 5}$ adalah sebuah *Latin square* berukuran 5×5 atas himpunan W = {1,2,3,4,5}.■

2.3 Groupoid

Groupoid merupakan salah satu bagian dari sistem aljabar yang paling sederhana dan menjadi dasar bagi sistem aljabar lain yang lebih kompleks seperti *quasigroup*,

group, ring dan sebagainya. Berikut diberikan definisi yang lebih jelas mengenai *groupoid.* (Khudzaifah, 2014)

Definisi 2.3.1

Groupoid adalah himpunan berhingga P yang didalamnya didefinisikan sebuah operasi biner $*$ yang memenuhi $a * b \in P$ untuk semua $a, b \in P$. Hasil operasi biner pada *groupoid* yang memiliki n anggota dapat dinyatakan dalam sebuah matriks berukuran $n \times n$ yang seluruh anggotanya merupakan anggotanya *groupoid* tersebut baik sebagian maupun keseluruhan

Contoh 2.3.2

Berikut adalah *groupoid* P = {1,2,3,4} yang hasil operasi biner anggota-anggotanya dinyatakan sebagai matriks M,

$$M = \begin{bmatrix} 1 & 3 & 2 & 4 \\ 2 & 1 & 3 & 4 \\ 3 & 4 & 1 & 2 \\ 1 & 2 & 3 & 4 \end{bmatrix}$$

Dari matriks M dapat diketahui bahwa *groupoid* diatas memenuhi operasi biner $*$ sebagai berikut:

$$\begin{array}{llll} 1 * 1 = 1, & 1 * 2 = 3, & 1 * 3 = 2, & 1 * 4 = 4, \\ 2 * 1 = 2, & 2 * 2 = 1, & 2 * 3 = 3, & 2 * 4 = 4, \\ 3 * 1 = 3, & 3 * 2 = 4, & 3 * 3 = 1, & 3 * 4 = 2, \\ 4 * 1 = 1, & 4 * 2 = 2, & 4 * 3 = 3, & 4 * 4 = 4, \end{array}$$

Contoh 2.3.3

Seperti halnya pada Contoh 2.3.2, berikut adalah *groupoid* $P = \{1,2,3,4\}$ yang hasil operasi binernya dinyatakan sebagai matriks N,

$$N = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 1 & 2 & 1 & 1 \end{bmatrix}$$

Dari matriks N dapat diketahui bahwa *groupoid* di atas memenuhi operasi biner $*$ sebagai berikut:

$$1 * 1 = 1, \quad 1 * 2 = 3, \quad 1 * 3 = 2, \quad 1 * 4 = 1,$$
$$2 * 1 = 2, \quad 2 * 2 = 1, \quad 2 * 3 = 1, \quad 2 * 4 = 1,$$
$$3 * 1 = 2, \quad 3 * 2 = 2, \quad 3 * 3 = 2, \quad 3 * 4 = 2,$$
$$4 * 1 = 1, \quad 4 * 2 = 2, \quad 4 * 3 = 1, \quad 4 * 4 = 1.$$

Matriks N menyatakan *groupoid* dengan anggota 1,2,3, dan 4 meskipun matriks tersebut tidak memuat elemen 3 dan 4.■ (Khudzaifah, 2014)

2.4 Quasigroup

Definisi 2.4.1

Menurut Ochodcova dan Snasel (2001), sebuah *groupoid* $(Q,*)$ disebut *quasigroup* jika untuk setiap $u, v \in Q$ terdapat dengan tunggal $x, y \in Q$ sedemikian sehingga berlaku $u * x = v$ dan $y * u = v$.

Contoh 2.4.2

Diberikan suatu *quasigroup* $(Q,*)$ dengan $Q = \{3,4,5,6,7\}$ dan operasi biner $*$. Operasi biner pada *quasigroup* $(Q,*)$ didefinisikan sebagai berikut:

$$3 * 3 = 7, \quad 3 * 4 = 6, \quad 3 * 5 = 5, \quad 3 * 6 = 4,$$
$$3 * 7 = 3, \quad 4 * 3 = 4, \quad 4 * 4 = 3, \quad 4 * 5 = 7,$$
$$4 * 6 = 6, \quad 4 * 7 = 5, \quad 5 * 3 = 6, \quad 5 * 4 = 5,$$
$$5 * 5 = 4, \quad 5 * 6 = 3, \quad 5 * 7 = 7, \quad 6 * 3 = 3,$$
$$6 * 4 = 7, \quad 6 * 5 = 6, \quad 6 * 6 = 5, \quad 6 * 7 = 4,$$
$$7 * 3 = 5, \quad 7 * 4 = 4 \quad 7 * 5 = 3, \quad 7 * 6 = 7,$$
$$7 * 7 = 6. \blacksquare$$

Tabel Cayley adalah teknik untuk mempresentasikan operasi biner pada suatu himpunan yang jumlah elemennya berhingga dengan menempatkan semua hasil operasinya ke dalam sebuah *array* berbentuk persegi. Tabel Cayley ini disebut juga sebagai tabel operasi. Dengan menghilangkan *headline* dan *sideline* pada table operasi, diperoleh sebuah *latin square* berukuran $n \times n$ dengan n adalah banyaknya elemen pada suatu himpunan dalam sistem aljabar. Mengingat *quasigroup* adalah bagian dari sistem aljabar, maka tabel Cayley ini dapat di aplikasikan pada

quasigroup. oleh karena itu *quasigroup* bisa dikonstruksi dengan menggunakan *latin square*. berikut adalah teorema yang membahas tentang keterkaitan *groupoid*, *quasigroup* dan *latin square.* (Khudzaifah, 2014)

Teorema 2.4.3

Suatu *groupoid* $Q = \{q_1, \dots, q_n\}$ adalah sebuah *quasi-group* jika dan hanya jika tabel operasi berupa *latin square.*

(Bell, 2005)

Bukti:

Misal diberikan sebuah *quasigroup* $Q = \{q_1, \dots, q_n\}$ dengan n buah elemen. Berdasarkan sifat *quasigroup*, untuk setiap pasangan q_i dan q_j di dalam Q, dapat ditemukan dua elemen tunggal yakni q_k dan q_l di dalam Q sedemikian sehingga berlaku $q_i q_k = q_j$ dan $q_l q_i = q_j$ dengan $i, j, k, l \in \{1, \dots, n\}$. Dari sini dapat diketahui bahwa untuk tiap-tiap i dan j terdapat k dan l yang tunggal sedemikian sehingga $ik = j$, dan $li = j$. Hal ini ekuivalen dengan pernyataan bahwa untuk setiap baris dan kolom yang saling berpotongan diperoleh suatu entri yang tunggal, sehingga bisa dikonstruksikan sebuah *latin square* dari tabel operasi *quasi group* Q.

Misal diberikan sebuah latin square $B = (b_{ij})$ yang berukuran $n \times n$ dengan $i = 1, \dots, n$ dan $j = 1, \dots, n$, yang mana penomoran pada baris ke-1 sampai ke-n dimulai dari atas ke bawah, dan penomoran pada kolom ke-1 sampai kolom ke-n dimulai dari kiri ke kanan. Didefinisikan $ij = b_{ij.}$ Karena B baris $\alpha \in \{1, \dots, n\}$ dan sebuah entri pada barisan tersebut $\beta \in \{1, \dots, n\}$, selalu dapat di temukan kolom tunggal $i \in \{1, \dots, n\}$ sedemikian sehingga $\alpha i = \beta$. Hal yang sama juga terjadi dengan memandang α sebagai kolom, akan selalu dapat ditemukan baris tunggal j sedemikian sehingga $j\alpha = \beta$. Hal ini sesuai dengan sifat *quasigroup*, yang mana untuk setiap α dan β elemen *quasigroup* dapat ditemukan i dan j sedemikian sehingga berlaku $\alpha i = \beta$ dan $j\alpha = \beta$, oleh karena itu suatu *quasigroup* dapat di konstruksikan dari *latin square.* Karena pembuktian dari kanan dan kiri terpenuhi, maka Teorema 2.4.3 terbukti benar.■

(Bell, 2005)

Contoh 2.4.4

Diberikan *quasigroup* $(Q,*)$ dengan $Q = \{3,4,5,6,7\}$. Operasi biner pada *quasigroup* ini didefinisikan sebagaimana yang ada pada Contoh 2.4.2. Operasi biner pada *quasigroup* $(Q,*)$ dapat disajikan dalam bentuk tabel operasi berikut:

Tabel 1. *Quasigroup* $(Q,*)$

*	3	4	5	6	7
3	7	6	5	4	3
4	4	3	7	6	5
5	6	5	4	3	7
6	3	7	6	5	4
7	5	4	3	7	6

Dengan menghilangkan *headline* dan sideline pada Tabel 1 di atas diperoleh *latin square*,

$$Q_{5\times5} = \begin{bmatrix} 7 & 6 & 5 & 4 & 3 \\ 4 & 3 & 7 & 6 & 5 \\ 6 & 5 & 4 & 3 & 7 \\ 3 & 7 & 6 & 5 & 4 \\ 5 & 4 & 3 & 7 & 6 \end{bmatrix}$$

2.5 Kriptografi

Definisi 2.5.1

Kriptografi (*cryptography*) berasal dari bahasa yunani yang terdiri dari dua suku kata yaitu *kripto* dan *graphia*. *Kripto* berarti menyembunyikan, sedangkan *graphia* memiliki arti tulisan. Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi, seperti kerahasiaan data, integritas data, serta autentikasi data.(Riyanto, 2007) Contoh kriptografi dalam kehidupan sehari-hari diantaranya adalah transaksi melalui ATM, *pay television*, komunikasi dengan telepon selular, *barcode* dan sebagainya. (Munir, 2004)

Ada empat tujuan mendasar dari kriptografi yang juga merupakan aspek keamanan informasi, yaitu :

1. Kerahasiaan, adalah aspek yang digunakan untuk menjaga isi informasi dari siapapun kecuali orang yang memiliki wewenang untuk mengetahuinya. Terdapat banyak sekali pendekatan yang dapat digunakan untuk

merahasiakan data, termasuk membuat suatu algoritma matematis yang mampu mengubah data hingga menjadi sulit dibaca dan dipahami.

2. Integritas data, adalah aspek yang berhubungan untuk penjagaan dari perubahan data secara tidak sah. Untuk menjamin integritas data,seseorang atau sistem harus memiliki kemampuan untuk mendeteksi manipulasi data oleh pihak-pihak yang tidak memiliki wewenang. Bentuk dari manipulasi data antara lain menyisipkan, menghapus, dan mensubtitusikan data lain kedalam data yang sebenarnya.
3. Auntentikasi, adalah aspek yang berhubungan dengan identifikasi, baik autentifikasi pihak-pihak yang terlibat dalam pengiriman data maupun autentifikasi keaslian data. Kedua pihak yang terlibat dalam komunikasi harus mengenalkan diri satu sama lain. Informasi yang dikirim harus terbukti keasliannya meliputi asal usulnya, tanggal asal, isi informasi, tanggal pengiriman dan sebagainya.
4. Non-repudiation, adalah usaha untuk mencegah terjadinya penyangkalan terhadap tanggung jawab atau tindakan pengiriman suatu informasi, dengan kata lain jika pihak pengirim menyangkal telah mengirim suatu pesan, maka harus bisa dibuktikan bahwa pesan yang dikirim berasal dari pengirim tersebut.

Definisi 2.5.2

Enkripsi (*encryption)* adalah proses yang dilakukan untuk mengamankan sebuah pesan (disebut *plaintext*) menjadi pesan yang tersembunyi dan tidak dapat dibaca (disebut *ciphertext*). Menurut ISO 7498-2, terminologi yang lebih tepat digunakan adalah *encipher*.

(Sasongko, 2005)

Definisi 2.5.3

Dekripsi (*decryption*) adalah proses untuk mengubah *ciphertext* menjadi *plaintext*. Menurut IS) 7498-2, terminology yang tepat untuk proses ini adalah *decipher*.

(Sasongko, 2005)

Definisi 2.5.4

Algoritma kriptografi atau yang sering disebut dengan *cipher* adalah suatu fungsi matematis yang digunakan untuk melakukan *encipher* dan *decipher* (Riyanto,

2007). Secara umum algoritma kriptografi dibagi menjadi 2 jenis yaitu algoritma kunci rahasia dan algoritma kunci publik (Markovski, dkk., 1997). Algoritma kriptografi modern tidak lagi mengandalkan keamanannya pada kerahasiaan algoritma tetapi pada kerahasiaan kunci (Budiono, 2004). *Plaintext* yang sama bila disandikan dengan kunci yang berbeda akan menghasilkan *ciphertext* yang berbeda pula.

Definisi 2.5.5

Algoritma kunci rahasia atau biasa disebut algoritma simetris adalah algoritma kriptografi yang menggunakan kunci yang sama untuk proses encipher dan deciphernya. Keamanan algoritma konvensional tergantung pada kunci (Riyanto, 2007). Membocorkan kunci sama artinya dengan memberikan kesempatan bagi pihak tak berwenang untuk melakukan *encipher* dan *decipher* pada *plaintext* (Ochodcova, 2012). Algoritma kriptografi yang termasuk dalam algoritma konvensional diantaranya adalah :

1. *Substitution Cipher*

   Substitution cipher adalah algoritma yang mengganti setiap karakter dari *plaintext* dengan karakter lain dalam susunan abjad tanpa adanya perubahan pada susunan abjad asli. Contoh algoritma ini diantaranya *Caesar cipher* dan *vigenere cipher*.

2. *Transposition Cipher*

   Transposition cipher adalah algoritma yang mengubah susunan karakter dari *plaintext* tanpa mengganti karakter yang ada dengan karakter lain. Contoh algoritma ini adalah *rail fence*.

3. *Block Cipher*

   Block cipher adalah algoritma yang membagi karakter pada *plaintext* menjadi blok dengan ukuran tertentu yang mana setiap blok dikodekan dengan menggunakan kunci yang sama. Empat mode operasi yang lazim diterapkan pada algoritma ini adalah *Electronic Code Book (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFC)* dan *Output Feedback (OFB).*

4. *Stream Cipher*

   Stream cipher adalah algoritma yang mengkodekan karakter persatuan karakter seperti bit, byte, nible, dan sebagainya. Pada tipe pengkodean satu satuan karakter digunakan kunci yang dibangkitkan dari kunci sebelumnya.

Definisi 2.5.6

Algoritma kunci publik atau yang biasa disebut dengan algoritma kunci asimetris adalah algoritma kriptografi dengan menggunakan kunci yang berbeda untuk proses encipher dan deciphernya, yang mana kunci untuk *encipher* dapat diketahui oleh publik tetapi untuk untuk proses *decipher*nya hanya diketahui oleh pihak yang berwenang. Contoh algoritma kunci public adalah RSA, Elgamal, McEliece, LUC, dan DSA

(Riyanto, 2007).

Definisi 2.5.7

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

1. Pilih dua buah bilangan prima sembarang, p dan q.
2. Hitung $n = p \times q$ (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 $\phi(n)$. Sehingga (e, $\phi(n)$)=1.
5. Bangkitkan kunci privat dengan menggunakan $e \times d = 1 \pmod{\phi(n)}$.

   Perhatikan bahwa $e \times d = 1 \pmod{\phi(n)}$ ekivalen dengan

   $e \times d = 1 + k\,\phi(n)$,sehingga 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, n)
- Kunci privat adalah pasangan (d, n)

(Munir, 2004)

Definisi 2.5.8

Algoritma hibrida adalah algoritma yang memanfaatkan dua tingkatan kunci, yaitu kunci rahasia (simetri) yang disebut juga *session key* (kunci sesi) untuk enkripsi data dan pasangan kunci rahasia-kunci publik (asimetris) untuk pemberian tanda tangan digital serta melindungi kunci simetri.

(Doni Ariyus, 2008)

BAB III
HASIL DAN PEMBAHASAN

3.1 Mengkonstruksi Kunci Rahasia dalam Bentuk *Quasigroup Cipher*

Pada subbab ini akan dibahas suatu metode pembentukan kunci rahasia yang menerapkan sifat *quasigroup* untuk proses *encipher* dan *decipher*. Teorema dan definisi berikut dikutip dari Markovski (1997) dan Khudzaifah (2014)

Teorema 3.1.1

Misalkan $(A,*)$ adalah *quasigroup* yang menetapkan sebuah operasi biner \ pada A sedemikian sehingga untuk semua $x, y \in A$ berlaku,

$$x \backslash y = z \leftrightarrow x * z = y,$$

Maka *groupoid* $(A,\backslash)$ adalah *quasigroup.*

Bukti :

Groupoid $(A,\backslash)$ adalah sebuah *quasigroup* jika untuk setiap $x, z \in A$ selalu dapat ditemukan $y, w \in A$ sedemikian sehingga berlaku:

$$w \setminus x = z \text{ dan } x \setminus y = z.$$

$w \setminus x = z$ dan $x \setminus y = z$ akan terpenuhi jika dan hanya jika $w * z = x$ dan $x * z = y$ terdefinisi di $(A,*)$. Karena $(A,*)$ adalah *quasigroup* maka berlaku sifat tertutup sehingga $w * z = x$ dan $x * z = y$ selalu terpenuhi dalam *quasigroup* $(A,*)$. Hal ini mengimplikasikan bahwa $w \setminus x = z$ dan $x \setminus y = z$ berlaku di $(A,\backslash)$, dengan demikian terbukti $(A,\backslash)$ adalah *quasigroup.*■

Contoh 3.1.2

Misalkan $A = (r, s, t)$ dan *quasigroup* $(A,*)$ didefinisikan oleh Tabel 3.1 berikut:

Tabel 3.1. *Quasigroup* $(A,*)$

$*$	r	s	t
r	t	r	s
s	s	t	r
t	r	s	t

Dari Tabel 3.1 diketahui bahwa *quasigroup* $(A,*)$ memenuhi operasi biner berikut:

$r * r = t$; $s * r = s$; $t * r = r$;

$r * s = r;$ $s * s = t;$ $t * s = s;$

$r * t = s;$ $s * t = r;$ $t * t = t.$

Jika z mewakili elemen pada kolom dan x mewakili elemen-elemen pada baris sedemikian sehingga $x * z = y$, maka untuk membentuk *quasigroup* $(A,\backslash)$, harus dipenuhi

$$x \backslash y = z \leftrightarrow x * z = y$$

Sehingga,

$r * r = t \leftrightarrow r \backslash t = r;$ $r * s = r \leftrightarrow r \backslash r = s;$

$r * t = s \leftrightarrow r \backslash s = t;$ $s * r = s \leftrightarrow s \backslash s = r;$

$s * s = t \leftrightarrow s \backslash t = s;$ $s * t = r \leftrightarrow s \backslash r = t;$

$t * r = r \leftrightarrow t \backslash r = r;$ $t * s = s \leftrightarrow t \backslash s = s;$

$t * t = t \leftrightarrow t \backslash t = t.$

Dari hasil operasi biner data, dapat dikonstruksikan Tabel 3.2 *quasigroup* $(A,\backslash)$ sebagai berikut:

Tabel 3.2. *Quasigroup* $(A,\backslash)$

$\backslash$	r	s	t
r	s	t	r
s	t	r	s
t	r	s	t

Definisi 3.1.3

Operasi $\backslash$ adalah *dual* dari $*$ sehingga *quasigroup* $(A,\backslash)$ adalah *dual* dari *quasigroup* $(A,*)$.$(A,*,\backslash)$ adalah *quasigroup* yang diperoleh dari hasil perluasan *quasigroup* $(A,*)$.

Teorema 3.1.4

Quasigroup $(A,*,\backslash)$ memenuhi persamaan identitas :

$x \backslash (x * y) = y,$ $x * (x \backslash y) = y$, Untuk semua $x, y \in A$.

Bukti:

Ambil sebarang $x, y, z \in A$. Misalkan $x * y = z,$ berdasarkan Teorema 3.1.1 berlaku $x \backslash z = y$. subtitusikan nilai $z = x * y$, diperoleh persamaan identitas, $x \backslash (x * y) = y$.

Ambil sebarang $x, y, w \in A$. Misalkan $x \setminus y = w$, berdasarkan Teorema 3.1.1 berlaku $x * w = y$. subtitusikan nilai $w = x \setminus y$, diperoleh persamaan identitas, $x * (x \setminus y) = y$. Dengan demikian Teorema 3.1.4 terbukti. ■

Teorema 3.1.4 di atas menjelaskan tentang sifat *quasigroup* $(A,*)$ dan *quasigroup dual*-nya $(A,\setminus)$, yang mana komposisi dari dua *quasigroup* ini akan menghasilkan persamaan identitas. Sifat ini sangat berguna dalam proses *encipher* dan *decipher*. *Ciphertext* yang diperoleh dari proses enchiper dengan menggunakan *quasigroup* $(A,*)$, akan diubah kembali menjadi *plaintext* dengan menggunakan *quasigroup* $(A,\setminus)$ pada proses *decipher*.

Contoh 3.1.5

Misalkan $A = (r, s, t)$ dan *quasigroup* $(A,*,\setminus)$ didefinisikan oleh Tabel 3.3 dan Tabel 3.4 sebagaimana pada Contoh 3.1.2.

Tabel 3.3. *Quasigroup* $(A,*)$

$*$	r	s	t
r	t	r	s
s	s	t	r
t	r	s	t

Tabel 3.4. *Quasigroup* $(A,\setminus)$

$\setminus$	r	s	t
r	s	t	r
s	t	r	s
t	r	s	t

Dengan menerapkan Teorema 3.1.4 diperoleh persamaan identitas sebagai berikut:

$r \setminus (r * r) = r \setminus t = r$; $\quad r * (r \setminus t) = r * r = t$;

$r \setminus (r * s) = r \setminus r = s$; $\quad r * (r \setminus r) = r * s = r$;

$r \setminus (r * t) = r \setminus s = t$; $\quad r * (r \setminus s) = r * t = s$;

$s \setminus (s * r) = s \setminus s = r$; $\quad s * (s \setminus s) = s * r = s$;

$s \setminus (s * s) = s \setminus t = s$; $\quad s * (s \setminus t) = s * s = t$;

$s \setminus (s * t) = s \setminus r = t$; $\quad s * (s \setminus r) = s * t = r$;

$t \setminus (t * r) = t \setminus r = r$; $\quad t * (t \setminus r) = t * r = r$;

$t \setminus (t * s) = t \setminus s = s; \qquad t * (t \setminus s) = t * s = s;$

$t \setminus (t * t) = t \setminus t = t; \qquad t * (t \setminus t) = t * t = t.$ ■

Definisi 3.1.6

Misalkan $u_i \in A^+, k \in \mathbb{N}, k \geq 1$, dan $a_1 \in A$ maka,

$$\begin{aligned} f_*(u_1u_2 \dots u_k) = v_1v_2 \dots v_k \quad \Leftrightarrow \quad & v_1 = a_1 * u_1, \\ & v_2 = v_2 * u_2, \\ & v_3 = v_3 * u_3, \\ & \vdots \\ & v_{i+1} = v_i * u_{i+1}, \qquad i = 1,2, \dots, k-1, \\ f_\setminus(u_1u_2 \dots u_k) = v_1v_2 \dots v_k \quad \Leftrightarrow & v_1 = a_1 \setminus u_1, \\ & v_2 = v_1 \setminus u_2, \\ & v_3 = v_2 \setminus u_2, \\ & \vdots \\ & v_{i+1} = v_i \setminus u_{i+1}, \qquad i = 1,2, \dots, k-1. \end{aligned}$$

Sixtuple $(A,*,\setminus, a_1, f_*, f_\setminus)$ disebut *quasigroup cipher* atas alfabet A.

Contoh 3.1.7

Misalkan $A = (a, b, c, d)$ dan $a_1 = a$. *Quasigroup* $(A,*)$ dan $(A,\setminus)$ didefinisikan seperti pada Tabel 3.5 dan Tabel 3.6 berikut:

Tabel 3.5. *Quasigroup* $(A,*)$

$*$	a	b	c	d
a	b	c	d	a
b	a	b	c	d
c	d	a	b	c
d	c	d	a	b

Tabel 3.6. *Quasigroup* $(A,\setminus)$

$\setminus$	a	b	c	d
a	d	a	b	c
b	a	b	c	d
c	b	c	d	a
d	c	d	a	b

Misal diberikan $u = adabbaca$, maka diperoleh f_* -nya sebagai berikut:

$v_1 = a_1 * u_1 = a * a = b \quad v_2 = v_1 * u_2 = b * d = d$

$v_3 = v_2 * u_3 = d * a = c \quad v_4 = v_3 * u_4 = c * b = a$

$v_5 = v_4 * u_5 = a * b = c \quad v_6 = v_5 * u_6 = c * a = d$

$v_7 = v_6 * u_7 = d * c = a \quad v_8 = v_7 * u_8 = a * a = b$

Sehingga $f_*(adabbaca) = bdcacdab$. Dengan nilai $u == bdcacdab$ diperoleh $f_\backslash$ -nya sebagai berikut:

$v_1 = a_1 \backslash u_1 = a \backslash b = a \quad v_2 = u_1 \backslash u_2 = b \backslash d = d$

$v_3 = u_2 \backslash u_3 = d \backslash c = a \quad v_4 = u_3 \backslash u_4 = c \backslash a = b$

$v_5 = u_4 \backslash u_5 = a \backslash c = b \quad v_6 = u_5 \backslash u_6 = c \backslash d = a$

$v_7 = u_6 \backslash u_7 = d \backslash a = c \quad v_8 = u_7 \backslash u_8 = a \backslash b = a$

Sehingga $f_\backslash(bdcacdab) = adabbaca$. ■ (Khudzaifah, 2014)

3.2 Mengkonstruksi Kunci Rahasia dalam Bentuk *Quasigroup Cipher* Menggunakan *Quasigroup* atas *Order p-1*

Pada subbab ini akan dibahas suatu metode pembentukan kunci rahasia yang menerapkan sifat *quasigroup order* $p - 1$ untuk proses *encipher* dan *decipher*. Teori mengenai *quasigroup order* $p - 1$ berikut dikutip dari Gligoroski (2004) .

Definisi 3.2.1 (Transformasi String)

Misalkan himpunan huruf alfabet adalah himpunan berhingga Q dan dinotasikan Q^+ adalah himpunan semua kata tak kosong atau string berhingga yang terdiri atas anggota dari Q.

Anggota dari Q^+ akan dinotasikan sebagai $a_1a_2, \dots \ a_n$. $a_i \in Q$. Misalkan $*$ adalah operasi pada *quasigroup* pada himpunan Q dan $(Q,*)$ adalah *quasigroup*, untuk $a \in Q$ didefinisikan dua fungsi $e_a, d_a: Q^+ \longrightarrow Q^+$ sebagai berikut

Misalkan $a_i \in Q, \alpha = a_1a_2 \dots a_n$, maka

$$e_a(\alpha) = b_1b_2 \dots b_n \Leftrightarrow b_1 = a * a_1, b_2 = b_1 * a_2, \dots , b_n = b_{n-1} * a_n.$$

Sehingga $b_{i+1} = b_i * a_{i+1}$ untuk setiap $i = 0,1, \dots , n-1$, dimana $b_0 = a$, dan

$$d_a(\alpha) = c_1c_2 \dots c_n \Leftrightarrow c_1 = a * a_1, c_2 = a_1 * a_2, \dots , c_n = a_{n-1} * a_n.$$

Sehingga $c_{i+1} = a_i * a_{i+1}$ untuk setiap $i = 0,1, \dots , n-1$, dimana $a_0 = a$.

Fungsi e_a dan d_a disebut sebagai e- transformasi string dan d-transformasi string dari Q^+ berdasarkan operasi $*$ dengan *leader* a.

Definisi 3.2.2

Jika dipilih sebanyak k *leaders* $a_1, a_2, \dots, a_k \in Q$ maka didefinisikan pemetaan fungsi komposisi

$$E_k = E_{a_1 \dots a_k} = e_{a_1} \circ e_{a_2} \circ \cdots \circ e_{a_k}$$

dan

$$D_k = D_{a_1 \dots a_k} = d_{a_1} \circ d_{a_2} \circ \cdots \circ d_{a_k}$$

disebut sebagai E- dan D- *quasigroup* transformasi string pada Q^+.

Lemma 3.2.3

Fungsi E_k dan D_k adalah permutasi pada Q^+.

Bukti :

Berdasarkan Definisi 3.2.2 dikatakan bahwa E_k dan D_k merupakan fungsi komposisi, maka jelas bahwa E_k dan D_k memenuhi pemetaan bijektif, karena E_k dan D_k adalah pemetaan yang bijektif sehingga E_k dan D_k adalah permutasi pada Q^+.■

Lemma 3.2.4

Pada *quasigroup* $(Q,*)$ dengan diberikan himpunan dari k *leaders* $\{a_1, a_2, \dots, a_k\}$ maka invers dari $E_k = E_{a_1 \dots a_k} = e_{a_1} \circ e_{a_2} \circ \cdots \circ e_{a_k}$ adalah

$$E_k^{-1} = D_k = D_{a_k \dots a_1} = d_{a_k} \circ \cdots \circ d_{a_1}.$$

Bukti :

Berdasarkan Definisi 3.2.2 dikatakan bahwa E_k merupakan fungsi komposisi. Karena E_k adalah fungsi komposisi, maka jelas invers dari fungsi komposisi adalah $E_k^{-1} = D_k = D_{a_k \dots a_1} = d_{a_k} \circ \cdots \circ d_{a_1}$. ■

Definisi 3.2.5

Quasigroup $(Q,*)$ dan *k-tuple* $(a_1, a_2, \dots, a_k)$ dari *leader* $a_i \in Q$, sistem $((Q,*), (a_1, a_2, \dots, a_k), E_{a_1 \dots a_k}, D_{a_k \dots a_1})$ terdefinisi sebagai *quasigroup stream cipher* atas string di Q^+.

Lemma 3.2.6

Untuk suatu p bilangan prima dan bilangan K yang memenuhi $1 \leq K \leq p-2$, fungsi

$f_K(j) = \frac{1}{1+(K+j) mod (p-1)} mod\ p$ adalah permutasi dari element di $\mathbb{Z}_p^*$

Bukti :

Ambil sebarang $a, b \in J$, jika $a \neq b$ maka $f_K(a) \neq f_K(b)$ dan $b \in J$ terdapat $a \in J$ sedemikian sehingga $b = f(a)$. Karena fungsi $f_K(j)$ adalah pemetaan yang bijektif, maka terbukti bahwa $f_K(j)$ adalah permutasi. ■

Lemma 3.2.7

Operasi biner $*$ pada himpunan $Q = \{1,2, \dots, p-1\}$ didefinisikan sebagai

$$i * j = i \times f_K(j) mod\ p$$

membentuk *quasigroup* $(Q,*)$.

Bukti :

Ambil sebarang $u, v \in Q$ maka terdapat dengan tunggal $x, y \in Q$ sedemikian sehingga berlaku $u * x = v$ dan $y * u = v$. Maka terbukti operasi * pada himpunan Q membentuk *quasigroup.* ■

Akibat 3.2.8

Jika didefinisikan fungsi

$$g(i,j,K) = \left((i \times j^{-1} mod\ p) - 1 - K\right) mod\ (p-1)$$

yang mengambil parameter i, j, K dari himpunan $Q = \{1,2, \dots, p-1\}$, yang memetakan himpunan $A = \{1,2, \dots, p-1\}^3$ ke himpunan $B = \{1,2, \dots, p-2\}$ maka pembagi kiri $(Q,\backslash)$ dari *quasigroup* $(Q,*)$ yang didefinisikan pada Lemma 3.2.7 didefinisikan sebagai

$$i \backslash j = \begin{cases} g(i,j,K), \text{jika } g(i,j,K) \neq 0 \\ p-1, \text{jika } g(i,j,K) = 0 \end{cases}$$

3.3 Mengkonstruksi Algoritma *Quasigroup Cipher*

Dari teori *quasigroup* diatas bisa dibentuk suatu algoritma kriptografi *quasigroup cipher* sebagai berikut :

Algoritma Enkripsi

1. Pilih sebarang bilangan bulat K, $1 \leq K \leq p-1$ yang mana *quasigroup* $(Q,*)$ terdefinisi untuk elemen $\{1,2, \dots, p-1\}$ dengan persamaan pada Lemma 3.2.7, dengan p adalah sebarang bilangan prima yang dipilih.
2. Pilih $k \geq 3$ bilangan bulat acak $a_i, i = 1,2, \dots, k$, $1 \leq a_i \leq p-2$ untuk menjadi *leader* untuk *quasigroup cipher*.

3. Ubah setiap karakter pada pesan m_μ menjadi bilangan bulat pada range $\{1,2,\dots,p-1\}$, dengan μ adalah indeks setiap karakter dari pesan.
4. Secara berulang hitung $m_\mu^i = a_i * m_\mu^{i-1}$, dimana $m_\mu^0 = m_\mu, i = 1,\dots,k$ dan $*$ adalah operasi *quasigroup* yang terdefinisi pada Lemma 3.2.7.
5. $c_\mu = m_\mu^k$ dan update nilai *leader* dengan $a_i = m_\mu^i, i = 1,\dots,k-1$ dan $a_k = 1 + (\sum_{i=1}^{k} m_\mu^i \ mod\ (p-1))$.
6. Didapatkan pesan yang terenkripsi c_μ (*ciphertext*).

Algoritma Dekripsi

1. Inputkan K untuk membuat $(Q,\backslash)$ dan didapatkan sejumlah k *leader*.
2. Secara berulang hitung $c_\mu^k = a_k \backslash c_\mu$, $c_\mu^i = a_i \backslash c_\mu^{i+1}, i = k-1,\dots,1$ dan $\backslash$ adalah operasi $quasigroup$ yang terdefinisi pada Akibat 3.2.8.
3. $m_\mu = c_\mu^1$ dan update nilai *leader* dengan $a_i = c_\mu^{i+1}, i = 1,\dots,k-1$ dan $a_k = 1 + (c_\mu + \sum_{i=2}^{k} c_\mu^i \ mod\ (p-1))$.
4. Didapatkan *plaintext* m_μ. (Khudzaifah, 2014)

3.4 Mengkonstruksi Algoritma Hibrida *(RSA- Quasigroup Cipher)*

Ilustrasi proses *encipher* dan *decipher*

1. Admin membangkitkan kunci publik dan kunci privat dengan algoritma RSA yang mana kunci publik akan dikirimkan ke Dosen.
2. Dosen mengenkripsi soal ujian dengan algoritma *Quasigroup cipher*, dan mengenkripsi *session key* dari *Quasigroup cipher* dengan kunci publik yang diberikan oleh Admin dengan menggunakan Algoritma RSA. Pesan dan key yang telah terenkripsi dikirim ke Admin.
3. Admin mendekripsi *session key* dari B dengan menggunakan kunci privat algoritma RSA, lalu mendekripsi pesan dari Dosen dengan menggunakan *session key* yang sudah terdekripsi dengan algoritma *Quasigroup cipher* .

Algoritma Hibrida *(RSA- Quasigroup Cipher)*

Algoritma Enkripsi

1. Pilih sebarang bilangan bulat K, $1 \le K \le p-1$ yang mana *quasigroup* $(Q,*)$ terdefinisi untuk elemen $\{1,2,\dots,p-1\}$ dengan persamaan pada Lemma 3.2.7, dengan p adalah sebarang bilangan prima yang dipilih.
2. Enkripsi K dengan algoritma RSA.

3. Pilih $k \geq 3$ bilangan bulat acak $a_i, i = 1,2, \dots, k,\ 1 \leq a_i \leq p-2$ untuk menjadi *leader* untuk *quasigroup cipher* dan enkripsikan dengan algoritma RSA.
4. Ubah setiap karakter pada pesan m_μ menjadi bilangan bulat pada range $\{1,2, \dots, p-1\}$, dengan μ adalah indeks setiap karakter dari pesan.
5. Secara berulang hitung $m_\mu^i = a_i * m_\mu^{i-1}$, dimana $m_\mu^0 = m_\mu, i = 1, \dots, k$ dan $*$ adalah operasi *quasigroup* yang terdefinisi pada Lemma 3.2.7.
6. $c_\mu = m_\mu^k$ dan update nilai *leader* dengan $a_i = m_\mu^i, i = 1, \dots, k-1$ dan $a_k = 1 + (\sum_{i=1}^{k} m_\mu^i \ mod\ (p-1))$.
7. Didapatkan *session key* terenkripsi dan pesan yang terenkripsi c_μ (*ciphertext*).

(Khudzaifah, 2014)

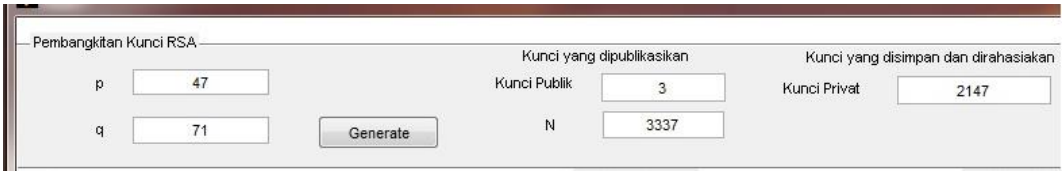
Algoritma Dekripsi

1. Dekripsi *Session key* dengan algoritma RSA, maka didapatkan K untuk membuat $(Q,\backslash)$ dan didapatkan sejumlah k *leader*.
2. Secara berulang hitung $c_\mu^k = a_k \backslash c_\mu$, $c_\mu^i = a_i \backslash c_\mu^{i+1}, i = k-1, \dots, 1$ dan $\backslash$ adalah operasi $quasigroup$ yang terdefinisi pada Akibat 3.2.8.
3. $m_\mu = c_\mu^1$ dan update nilai *leader* dengan $a_i = c_\mu^{i+1}, i = 1, \dots, k-1$ dan $a_k = 1 + (c_\mu + \sum_{i=2}^{k} c_\mu^i \ mod\ (p-1))$.
4. Didapatkan *plaintext* m_μ.

(Khudzaifah, 2014)

Contoh

Misal Admin jurusan matematika UIN butuh soal Ujian Akhir Semester(UAS). Agar soal tidak bocor maka soal uas akan dikirim dari dosen ke Admin dalam pesan terenkripsi, maka Admin membangkitkan kunci algoritma RSA dan memberitahukan kunci publik kepada para dosen yang digunakan untuk mengenkripsi kunci dari algoritma *quasigroup cipher*. Untuk membangkitkan kunci RSA dipilih sebarang bilangan prima p dan q, pada contoh ini dipilih $p = 47, q = 71$, seperti pada Gambar 3.1.



Gambar 3.1

Tampilan program saat pembangkitan kunci algoritma RSA.

Setelah kunci publik diterima, maka para dosen UIN mengenkripsikan soal UAS dengan algoritma *quasigroup cipher* lalu mengenkripsi kan kunci dari *quasigroup cipher* tadi dengan algoritma RSA. Untuk mengenkripsi soal UAS dengan algoritma *quasigroup cipher* pilih sebarang bilangan bulat K, $1 \leq K \leq p-1$, dengan p adalah sebarang bilangan prima yang dipilih (karena *char* yang terdefinisi pada MATLAB sebanyak 127, maka pada program ini menggunakan bilangan prima 127), dan pilih $k \geq 3$ bilangan bulat acak $a_i,\ i = 1,2,\dots,k, 1 \leq a_i \leq p-2$ untuk menjadi *leader* untuk algoritma *quasigroup cipher*. Pada contoh ini dipilih $K = 67, k_1 = 78, k_2 = 89, k_3 = 98$. Misalkan soal UAS yang dienkripsi adalah "Jelaskan definisi Grup, Ring, Field dan Daerah Integral beserta contohnya!".
Karakter "J" yang merupakan huruf awal dari kalimat di atas memiliki nilai ASCII 74, seperti perhitungan pada Lemma 2.3.7

enkrip=$\left(leader(i) * \frac{1}{1+(K+pesan(n))mod(p-1)} mod\ p\right) mod\ p$

- *Leader* ke-1

enkrip=$\left(78 * \frac{1}{1+(67+74)mod(126)} mod\ 127\right) mod\ 127 = 116$

- *Leader* ke-2

enkrip=(89*$\frac{1}{1+(67+116)mod(126)} mod\ 127) mod\ 127 = 30$

- *Leader* ke-3

enkrip=(98*$\frac{1}{1+(67+30)mod(126)} mod\ 127) mod\ 127 = 1$

Nilai 1 pada ASCII adalah karakter " "(spasi). Update *leader* 1=116, *leader* 2=30, dan *leader* 3=$1 + ((116 + 30 + 1)\ mod\ (126)) = 22$
Kemudian huruf setelah "J" adalah huruf "e" yang memiliki nilai ASCII 101

- *Leader* ke-1

enkrip=$\left(116 * \frac{1}{1+(67+101)mod(126)} mod\ 127\right) mod\ 127 = 47$

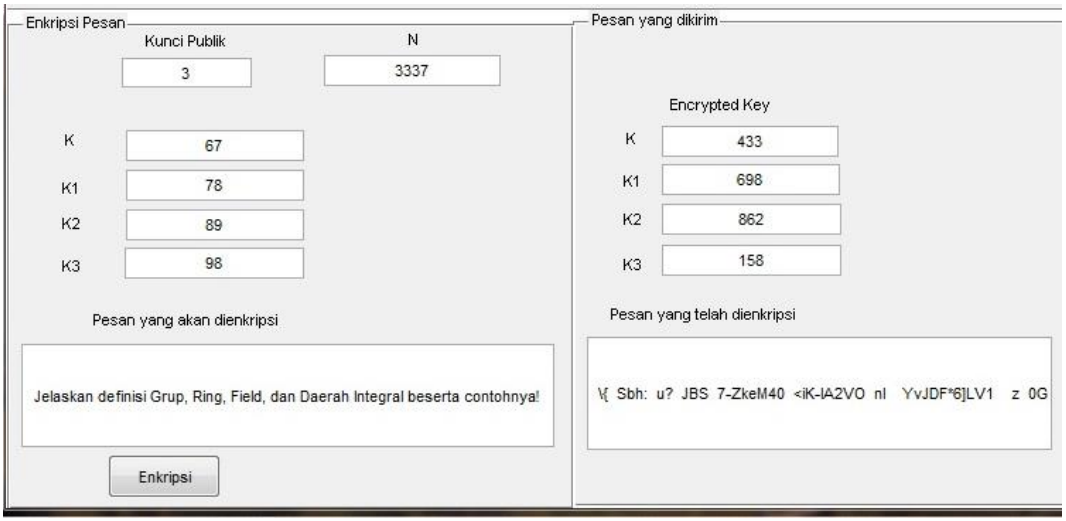
- *Leader* ke-2

enkrip=(30*$\frac{1}{1+(67+47)mod(126)}$ $mod\ 127)mod\ 127 = 61$

- *Leader* ke-3

enkrip=(22*$\frac{1}{1+(67+61)mod(126)}$ $mod\ 127)mod\ 127 = 92$

Nilai 92 pada ASCII adalah karakter "\", proses selanjutnya sama hingga karakter terakhir, sehingga didapatkan *ciphertext* seperti pada Gambar 3.2 dibawah ini.



Gambar 3.2

Tampilan program saat proses enkripsi.

(Khudzaifah, 2014)

Setelah soal UAS terenkripsi diterima Admin, maka kunci *quasigroup cipher* dienkripsi terlebih dahulu, lalu digunakan untuk mengenkripsi soal UAS.

Karakter awal dari *ciphertext* spasi memiliki nilai ASCII 1.

- *Leader* ke-1

dekrip=$\left((78 \times 1^{-1} mod\ p) - 1 - 67\right)\ mod\ (126) = 30$

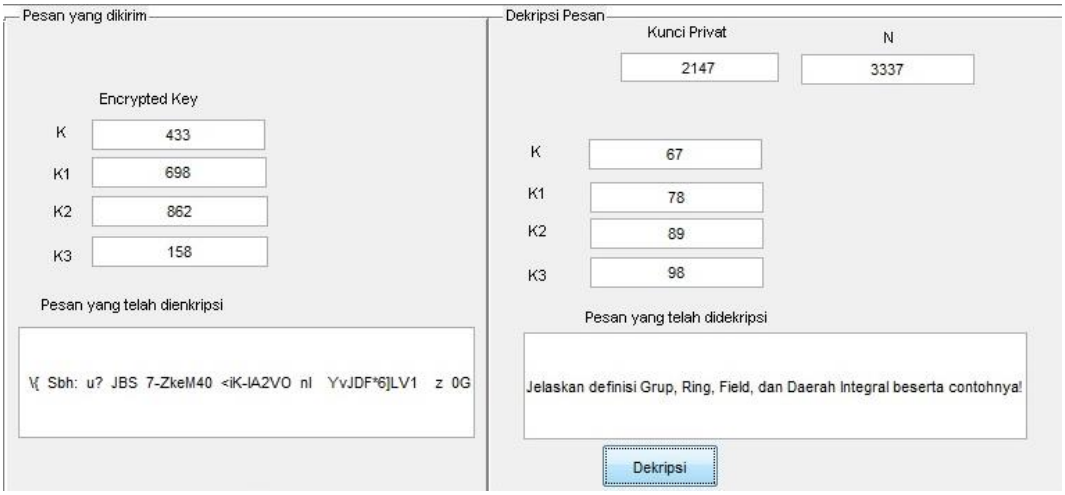
- *Leader* ke-2

dekrip=$\left((87 \times 30^{-1} mod\ p) - 1 - 67\right)\ mod\ (126) = 116$

- *Leader* ke-3

dekrip=$\left((78 \times 1^{-1} mod\ p) - 1 - 67\right)\ mod\ (126) = 74$

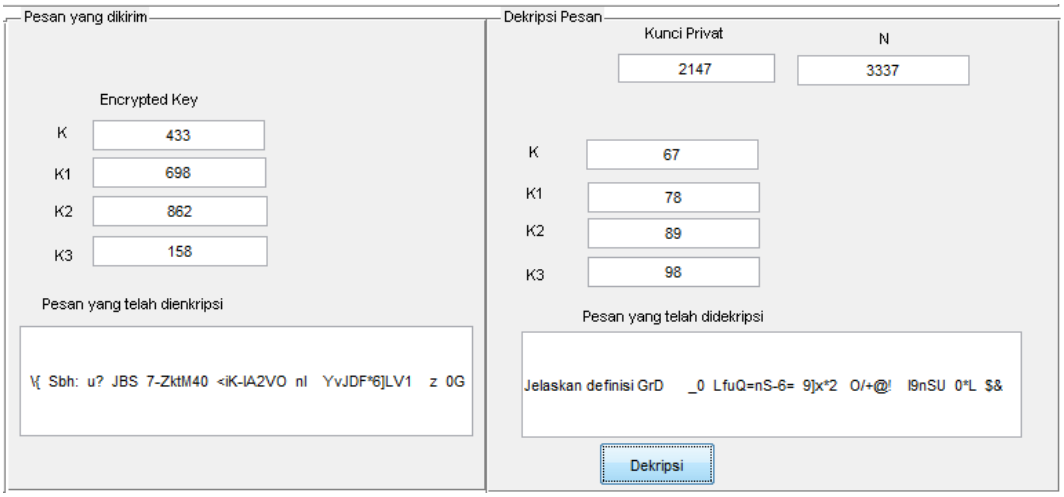
Nilai 74 pada ASCII adalah karakter "J", proses selanjutnya sama hingga karakter terakhir. Sehingga didapatkan *plaintext* seperti pada Gambar 3.3 dibawah ini.



Gambar 3.3

Tampilan program saat proses dekripsi.

Algoritma *quasigroup cipher* ini memiliki sistem keamanan cukup bagus, misalkan software pemecah kode memiliki kesalahan baca pada satu huruf saja. Contohnya huruf “e” pada pesan yang terbaca menjadi huruf “t” maka pesan menjadi tak terbaca, seperti pada Gambar 3.4.



Gambar 3.4

Tampilan program saat pemecah kode gagal mendekripsi pesan.

Maka didapatkan program yang memiliki tingkat keamanan yang baik, dan proses penyandian soal ujian dan penerjemahan soal ujian membutuhkan waktu yang singkat. Sehingga algoritma hibrida ini optimal untuk digunakan untuk mengamankan soal ujian.

BAB IV
KESIMPULAN

Berdasarkan hasil pembahasan pada Bab III, maka dapat diambil kesimpulan sebagai berikut.

1. Algoritma *quasigroup encryption* memiliki keamanan cukup baik. Hal ini dibuktikan pada contoh ketika *software* pemecah kode salah mendekripsi satu huruf saja maka pesan tidak bisa terbaca.
2. Kelemahan algoritma simetris adalah bila kuncinya diketahui oleh orang lain, maka soal ujian bisa bisa bocor. Dengan diperkuat algoritma RSA yang memiliki kunci asimetris, maka algoritma *quasigroup encryption* menjadi algoritma hibrida yang mempunyai tingkat keamanan lebih tinggi, karena memiliki 2 tingkatan kunci.
3. Kombinasi algoritma simetris dan asimetris, menghasilkan algoritma yang efektif dan efisien maka algoritma simetris digunakan untuk menyandikan soal ujian dan algoritma asimetris digunakan untuk mengamankan kunci dari algoritma simetris. Sehingga dihasilkan algoritma *quasigroup encryption* yang memiliki tingkat keamanan yang baik dan proses penyandian soal ujian dan penerjemahan soal ujian membutuhkan waktu yang singkat.

DAFTAR PUSTAKA

Al-Turky, M.A. 2007. On the number and Equivalent Latin Squares. *Journal of Al-Anbar University for pure Science* ISSN: 1991-8941 Vol.1 No.1 Page: 71-75

Ariyus, Dony. 2008. Pengantar Ilmu Kriptografi. Yogyakarta: Penerbit Andi.

Bhattacharya, P. B., dkk. 1990. *Basic Abstract Algebra.* New York: Cambridge University Press.

Bell, J. 2005. An Introduction to SDR's and Latin Squares. *More-head Electonic Journal of Applicable Mathematics* Issue 4 Page: 1-8

Budiyono, A. 2004. *Enkripsi Data Kunci Simetris dengan Algoritma Kriptografi LOKI97.* Bandung : Program Studi Magister Teknologi Informasi Institut Teknologi Bandung.

Glikoroski.D.2004. Stream Cipher Based on Quasigroup String Transformation in Zp*. Skopje: Faculty of Natural Sciences institute of Informatics

Gupta, P. 2012. *Mathematics with Bank of Questions*. New Delhi: Laxmi Publication(P) LTD.

Khudzaifah, M. 2014. Aplikasi Quasigroup Dalam Pembentukan Kunci Rahasia Pada Algoritma Hibrida *(Rsa-Quasigroup Cipher).Tesis*. Malang: Universitas Brawijaya.

Koscielny, C. 2002. Generating Quasigroups for Cryptograpic Applications. *International Journal of Applied Mathematic and Computer Science* Vol. 12 No.4 Page: 559-569.

Markovski,S., D.Glikoroski, dan S. Andonova.1997. Using Quasigroups for One-one Secure Encoding. *Proceeding of VII-th Confrence for Logic and Computing-LIRA'97* page 1-6

Menesez, A J., dkk.1996. *Handbook of Applied Cryptography*. USA: CRC Press, Inc. USA.

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

Ochodkova, E. dan V.Snasel. 2001. Using Quasigroups for Secure Encoding of File Sistem. *Proceedings of The Confrence for Security and Protection of Information* Page 175-181

Riyanto, M. Z 2007. Pengamanan Pesan Rahasia Menggunakan Algoritma Kriptografi Elgamal atas Group Pergandaan Zp*. *Skripsi*. Yogyakarta: Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Gadjah Mada Yogyakarta.

Sasongko, J. 2005. Pengamanan Data Informasi Menggunakan Kriptografi Klasik. *Jurnal Teknologi Informasi DINAMIK* Vol.10 No.3 Hal. 160-167.

Lampiran

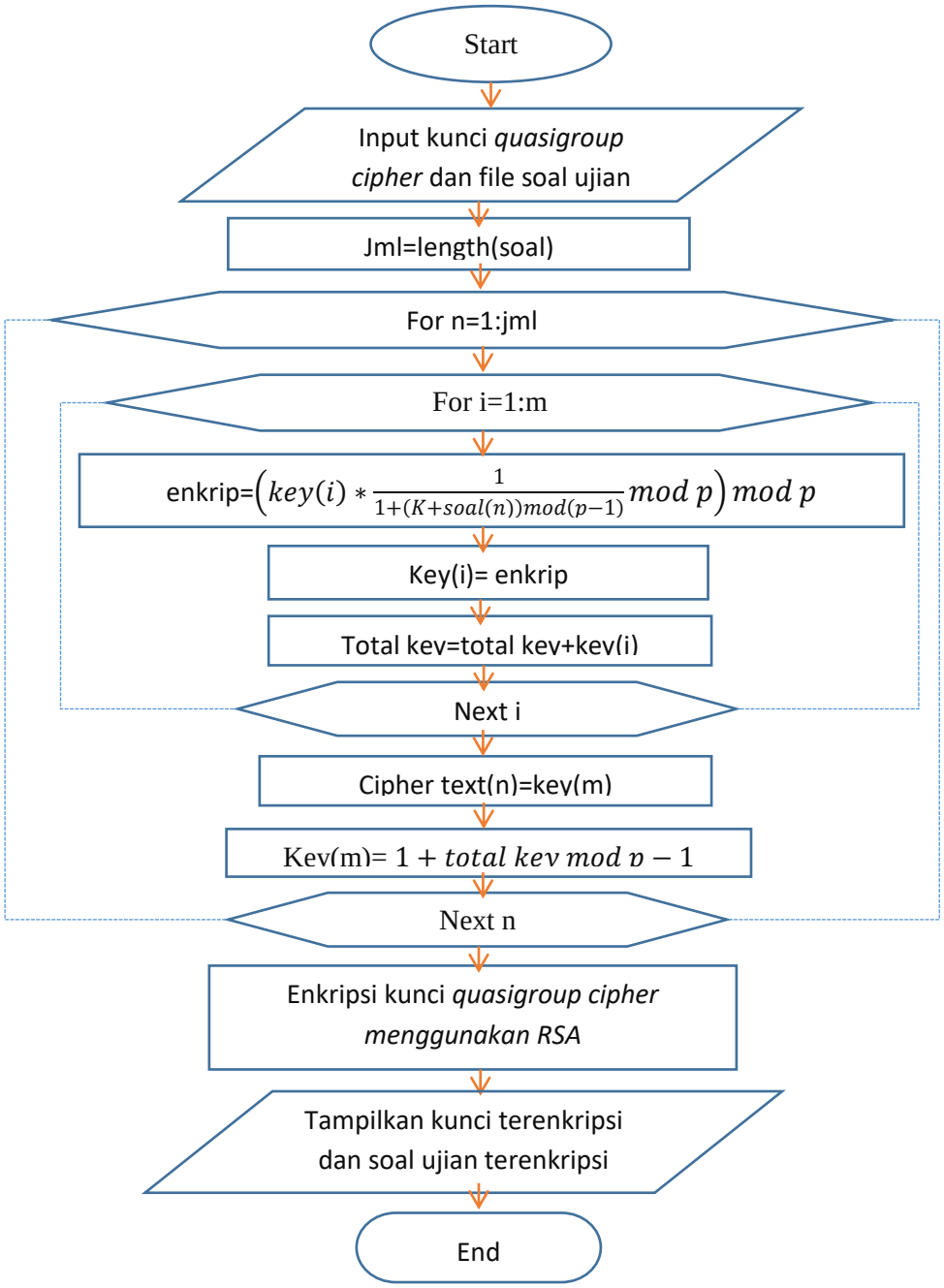
Listing Program Algoritma Hibrida

```
function bangkitRSAKunci_Callback(hObject, eventdata,
handles)
figure1=guidata(gcbo);
q=str2double(get(figure1.q,'string'));
[N,Phi,d,e] = intialize(p,q);
p=str2double(get(figure1.p,'string'));
set(figure1.d,'String',d);
set(figure1.e,'String',e);
set(figure1.N,'String',N);

function enkrip_Callback(hObject, eventdata, handles)
%function untuk mengenkripsi pesan
% input data
figure1=guidata(gcbo);
prima=str2double(get(figure1.prima,'string'));
k=str2double(get(figure1.k,'string'));
ka(1)=str2double(get(figure1.k1,'string'));
ka(2)=str2double(get(figure1.k2,'string'));
ka(3)=str2double(get(figure1.k3,'string'));
key(1)=k;
key(2)=ka(1);
key(3)=ka(2);
key(4)=ka(3);
pesan=fileread('pesan.txt');
set(figure1.pesan,'String',pesan);
jml=length(pesan);
for n=1:jml
   s=(k+pesan(n));
    if n>=2,
        ka(3)=1+mod((ka(1)+ka(2)+sandi(n-1)),prima-1);
    end;
for i=1:3
if i>=2,
    s=(k+ka(i-1));
end;
c=prima-1;
r=mod(s,c);
b=1+r;
y=invmodn(b,prima);
as=ka(i)*y;
ka(i)=mod(as,prima);
end;
sandi(n)=ka(3);
end
set(figure1.cipher,'String',char(sandi));
```

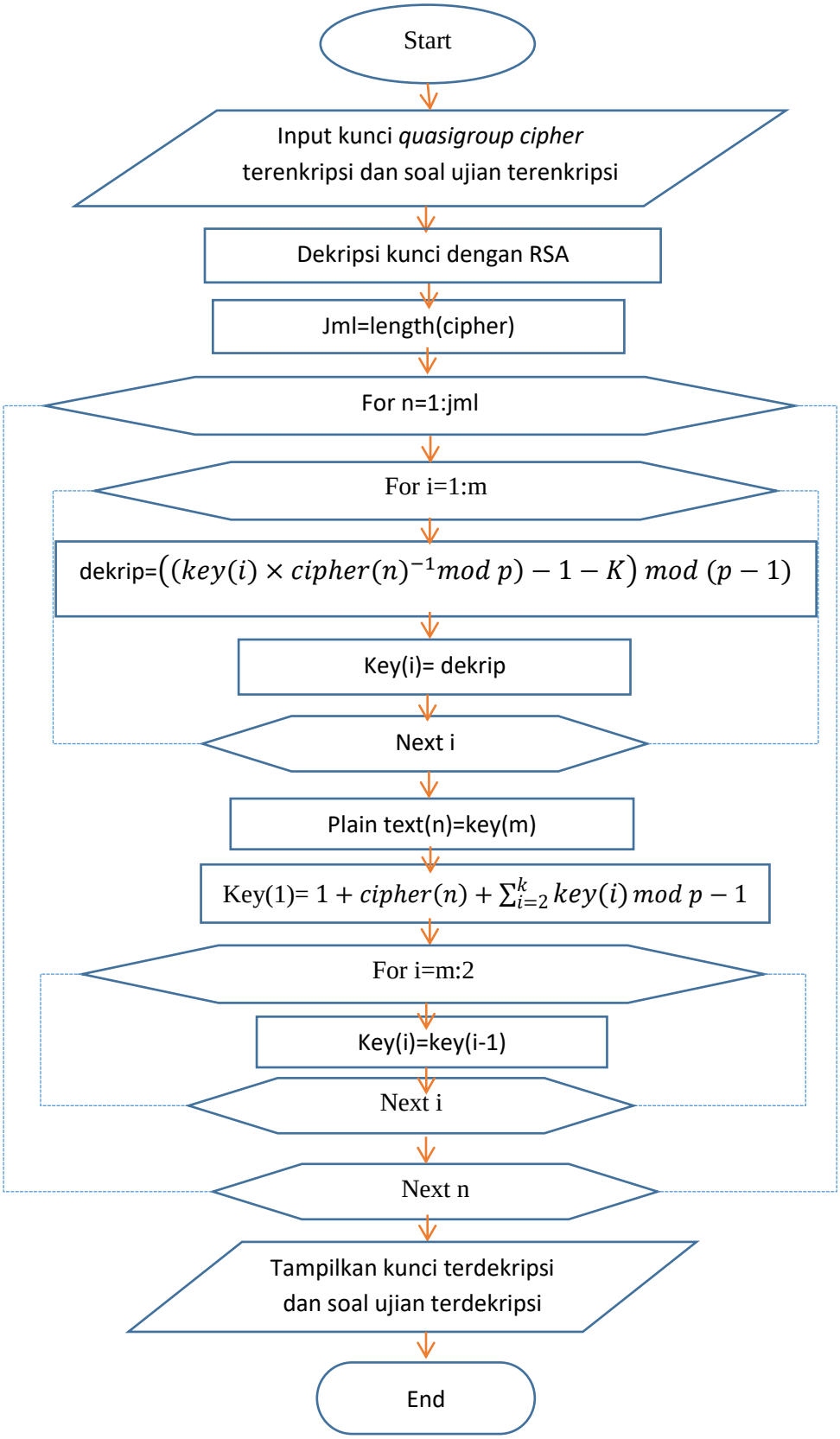
```
dlmwrite('cipher.txt',char(sandi),'delimiter','');
%Enkripsi key quasigroupcipher
e=str2double(get(figure1.e,'string'));
N=str2double(get(figure1.N,'string'));
for j= 1:4
   cipherk(j)= crypt(key(j),N,e);
end
set(figure1.ek,'String',cipherk(1));
set(figure1.ek1,'String',cipherk(2));
set(figure1.ek2,'String',cipherk(3));
set(figure1.ek3,'String',cipherk(4));


function Dekrip_Callback(hObject, eventdata, handles)
%function untuk mendekripsi pesan
% input data
figure1=guidata(gcbo);
prima=str2double(get(figure1.prima,'string'));
cipherk(1)=str2double(get(figure1.ek,'string'));
cipherk(2)=str2double(get(figure1.ek1,'string'));
cipherk(3)=str2double(get(figure1.ek2,'string'));
cipherk(4)=str2double(get(figure1.ek3,'string'));
d=str2double(get(figure1.d,'string'));
N=str2double(get(figure1.N,'string'));

% % %Dekripsi kunci yang terenkripsi
for j= 1:4
   key(j)= crypt(cipherk(j),N,d);
end
set(figure1.dk,'String',key(1));
set(figure1.dk1,'String',key(2));
set(figure1.dk2,'String',key(3));
set(figure1.dk3,'String',key(4));
k=key(1);
ka(1)=key(4);
ka(2)=key(3);
ka(3)=key(2);
cipher=fileread('cipher.txt');
set(figure1.cipher,'String',cipher);
jml=length(cipher)-1;
for n=1:jml
    cc=cipher(n)+127;
   ce=invmodn(cc,prima);
    if n>=2,
        leader(2)=ka(1);
        leader(3)=ka(2);
        ka(1)=1+mod((ka(1)+ka(2)+cipher(n-1)),prima-1);
        ka(2)=leader(2);
        ka(3)=leader(3);
```

```
    end;
for i=1:3
if i>=2,
   ce=invmodn(ka(i-1),prima);
end;
ij=ka(i)*ce;
ijp=mod(ij,prima);
ka(i)=mod([ijp-1-k],[prima-1]);
if (ka(i)==0),
    ka(i)=prima-1;
end;
end;
plainteks(n)=ka(3);
end;
dlmwrite('plain.txt',char(plainteks),'delimiter','');
set(figure1.plain,'String',char(plainteks));

function y=invmodn(b,p) % perhitungan invers modulo
b0=b;
n0=p;
t=1;
t0=0;
q=floor(n0/b0);
r=n0-q*b0;
while r>0,
temp=t0-q*t;
if(temp>=0),
temp=mod(temp,p);
end;
if (temp<=0),
temp=p-(mod(-temp,p));
end;
t0=t;
t=temp;
b0=r;
n0=b0;
q=floor(n0/b0);
r=n0-q*b0;
end;
if b0~=1,
y=[];
else
y=mod(t,p);
end;
```

***Flowchart* Proses Enkripsi Algoritma Hibrida (RSA-*Quasigroup Cipher*)**



***Flowchart* Proses Dekripsi Algoritma Hibrida (RSA-*Quasigroup Cipher*)**



Jadwal Seminar Progres Report Penelitian

APLIKASI *QUASIGROUP ENCRYPTION* UNTUK MENGAMANKAN SOAL UJIAN

Seminar : Kajian tentang Aplikasi *Quasigroup Encryption* untuk Mengamankan Soal Ujian

Hari/Tanggal : Kamis, 18 Mei 2017

Waktu	Kegiatan	Narasumber
07.30 – 09.30 WIB	Kajian Teori *Quasigroup*	Muhammad Khudzaifah, M.Si
09.30 – 11.30 WIB	Pembahasan *Security Problem On Computer Based Test*	Muhammad Khudzaifah, M.Si
11.30 – 12.30 WIB	Istirahat	
12.30 – 14.30 WIB	Kajian mengenai Aplikasi *Quasigroup Encryption* untuk Mengamankan Soal Ujian	Muhammad Khudzaifah, M.Si

Ketua Peneliti

Muhammad Khudzaifah, M.Si

NIP. 19900511 20160801 1 057

DAFTAR HADIR NARASUMBER

PROGRES REPORT PENELITIAN

APLIKASI *QUASIGROUP ENCRYPTION* UNTUK MENGAMANKAN SOAL UJIAN

Hari : Kamis

Tanggal : 18 Mei 2017

Jam : 07.30 – 14.30

Tempat : Gedung B Ruang 208

No	Nama	Judul Presentasi	Tanda Tangan
1	Muhammad Khudzaifah, M.Si	Aplikasi *Quasigroup Encryption* untuk Mengamankan Soal Ujian	

Malang,

Ketua Tim Peneliti,

Muhammad Khudzaifah, M.Si

NIP. 19900511 20160801 1 057

CATATAN PRESENTASI

Hari/Tanggal : Kamis/18 Mei 2017

Tempat : Gedung B Ruang 208

Acara : Seminar Progres Report Penelitian Aplikasi *Quasigroup Encryption* untuk Mengamankan Soal Ujian

Kegiatan diawali dengan Kajian Teori Tentang *quasigroup* yang merupakan salah satu teori dalam aljabar abstrak, sifat permutasi q*uasigroup* dapat diterapkan dalam banyak hal, terutama bidang kriptografi. Peserta sangat antusias dalam materi tersebut karena banyak peserta berasal dari bidang matematika.

Kegiatan selanjutnya adalah pembahasan permasalahan keamanan pada tes berbasis komputer yang saat ini sangat sering digunakan, terutama untuk UNBK (Ujian Nasional Berbasis Komputer) dan SBMPTN yang sebagian menggunakan CBT(*Computer Based Test*). Peserta sangat antusias mengikuti materi karena sedang hangatnya berita penyadapan dan serangan virus *wannacry.* Dari pembahasan permasalahan keamanan ini, dibutuhkan suatu system keamanan yang kuat dari serangan *hacker*. Namun, system keamanan yang kuat cenderung lama pada proses dekripsi, padahal durasi ujian sangat terbatas.

Kegiatan selanjutnya dilakukan setelah istirahat dan sholat dhuhur, yaitu kajian tentang aplikasi *quasigroup encryption* untuk mengamankan soal ujian. Peneliti mengusulkan suatu system keamanan yang kuat tapi mempunyai durasi proses dekripsi yang singkat, yaitu menggunakan algoritma hibrida yang merupakan algoritma *quasigroup encryption* yang diperkuat dengan algoritma RSA.

Algoritma kriptografi yang didasarkan dari quasigroup yaitu quasigroup encryption memiliki keamanan cukup baik, hal ini dibuktikan ketika kriptanalis salah mendekripsi satu huruf saja maka pesan tidak bisa terbaca. Algoritma q*uasigroup encryption* memiliki sebanyak n-kunci akan tetapi merupakan algoritma kunci simetris sehingga bila kuncinya diketahui orang lain, maka soal ujian bisa bocor. Sehingga diperkuat dengan algoritma RSA yang memiliki kunci asimetris, menjadi algoritma hibrida yang tingkat keamanannya lebih tinggi karena memiliki 2 tingkatan kunci.

Ketua Peneliti

Muhammad Khudzaifah, M.Si
NIP. 19900511 20160801 1 057

BIODATA PENELITI

IDENTITAS DIRI

Nama	: Muhammad Khudzaifah, M.Si.
NIP/NIK	: 19900511 20160801 1 057
Jenis Kelamin	: Laki-laki
Tempat dan Tanggal Lahir	: Sidoarjo, 11 Mei 1990
Status Perkawinan	: Kawin
Agama	: Islam
Golongan / Pangkat	: III b / Penata Muda Tk. I
Jabatan Fungsional Akademik	: Asisten Ahli
Perguruan Tinggi	: UIN Maulana Malik Ibrahim Malang
Alamat	: Jl. Gajayana 50 Malang
Telp./Faks.	: (0341) 551354 / (0341) 572533
Alamat Rumah	: Bukit Cemara Tidar A-54, Sukun-Malang
Telp./Faks.	: 0857 3000 9061
Alamat E-mail	: m_khudzaifah@yahoo.com

RIWAYAT PENDIDIKAN PERGURUAN TINGGI

Tahun Lulus	Jenjang	Perguruan Tinggi	Jurusan/ Bidang Studi
2012	S-1	Universitas Brawijaya Malang	Matematika
2014	S-2	Universitas Brawijaya Malang	Matematika

PELATIHAN PROFESIONAL

Tahun	Pelatihan	Penyelenggara
2013	Peserta Workshop "*Pemodelan Matematika*"	Jurusan Matematika Fakultas MIPA Universitas Brawijaya
2015	Peserta Pelatihan Pengembangan Keterampilan Dasar Teknik Instruksional	Kopertis Wilayah VII
2015	Peserta Workshop "*Analisis Real dan Geometri*"	STKIP PGRI Sidoarjo dan *IndoMS*
2016	Peserta Workshop "*Penyusunan Kurikulum KKNI*"	STKIP PGRI Pasuruan
2016	Peserta Workshop "*Integrated Learning Model of Ulul Albab Competence for Lecturer Pedagogy Enchancement*"	Lembaga Penjaminan Mutu (LPM) UIN Maulana Malik Ibrahim Malang

PENGALAMAN MENGAJAR

Mata Kuliah	Jenjang	Institusi/Jurusan/Program	Tahun
Pemrograman Komputer	S1	Jurusan Pendidikan Matematika STKIP PGRI Pasuruan	2015
Aljabar Linier	S1	Jurusan Pendidikan Matematika STKIP PGRI Pasuruan	2016
Komputer II	S1	Jurusan Pendidikan Matematika STKIP PGRI Pasuruan	2016
Teori Bilangan	S1	Jurusan Matematika FSAINTEK UIN Maulana Malik Ibrahim Malang	2016
Matematika Dasar	S1	Jurusan Biologi FSAINTEK UIN Maulana Malik Ibrahim Malang	2016
Pemrograman Komputer I	S1	Jurusan Matematika FSAINTEK UIN Maulana Malik Ibrahim Malang	2017
Praktikum Pemrograman Komputer I	S1	Jurusan Matematika FSAINTEK UIN Maulana Malik Ibrahim Malang	2017

KEGIATAN PROFESIONAL/PENGABDIAN KEPADA MASYARAKAT

Tahun	Jenis/Nama Kegiatan	Tempat
2016	Workshop Pendidikan Bagi Kepala dan Guru Madrasah Ibtidaiyah Ma'arif NU	Kecamatan Watulimo Trenggalek

Makalah/Poster Konferensi/Seminar/Workshop/Lokakarya/Simposium

Tahun	Judul Kegiatan	Penyelenggara	Panitia/ Peserta/ Pembicara
2011	*School of Programming*	Himamaster Universitas Brawijaya	Pembicara
2017	Pengembangan Kompetensi Penelitian Sains dan Teknologi	Pusat Studi Saintek Universitas Islam Malang	Panitia
2017	Seminar Integrasi Matematika dan Islam	Fakultas Sains dan Teknologi UIN Malang	Panitia
2017	*International Seminar on Green Technology 8*	Fakultas Sains dan Teknologi UIN Malang	Panitia

Malang, 12 Mei 2017

Perihal : **Undangan Seminar Progres Penelitian**

Kepada Yth.

Bapak/Ibu……

Di tempat

Assalamu'alaikum Wr. Wb.

Mengharap kehadiran bapak/ibu pada:

Hari, tanggal : Kamis, 18 Mei 2017

Pukul : 07.30-14.30 WIB

Tempat : Gedung B Ruang 208

Acara : Seminar Progres Report Penelitian Aplikasi *Quasigroup Encryption* untuk Mengamankan Soal Ujian

Demikian atas perhatian dan kehadiran Bapak/Ibu, disampaikan terima kasih.

Wassalamu'alaikum Wr. Wb.

Ketua Peneliti

Muhammad Khudzaifah, M.Si
NIP. 19900511 20160801 1 057

Daftar Hadir Seminar Penelitian
Aplikasi *Quasigrup Encryption* untuk Mengamankan Soal Ujian

Acara : Kajian tentang Aplikasi *Quasigroup Encryption* untuk Mengamankan Soal Ujian
Hari/Tanggal : Kamis, 18 Mei 2017

No.	Nama	Tanda Tangan
1	Astri Kumala	1
2	Paulana Meesa Silvana	2
3	Kurnia Shinta	3
4	Anisahur Rizqiyah	4
5	Fidyatus Saqitri	5
6	Nur Aini Amilatus solikah	6
7	Nur Azizah	7
8	Yeni Ayuwarni	8
9	Nadia Walindra.	9
10	Anggi Destiana	10
11	Dwi Noviana	11
12	Jie Yan Kirana E.F.A	12
13	DWY AINURROKHIMAH P.	13
14	Ismatiah Ummu	14
15	Ghina Ayu K.D	15
16	Evana Dyah P	16
17	Rapenda Mundi W.2	17
18	Nurul Hidayati	18
19	Siti Nur Hazimah S.S	19
20	Cici Erisa M.	20
21	Moch. Faisal Habibi	21
22	Ardi Muliawan	22
23	Ahmad Zaini	23
24	Sulaeman Hamdani A	24
25	Moh Mubtis F	25
26	Ahmad Ridho Alwan .T.	26
27	M. Husen Al Farrsy	27
28	M. Arif Abdillah	28
29	Mohammad Firman Bayuraksi	29
30	Dimas Adi P	30

Ketua Peneliti

Muhammad Khudzaifah, M.Si
NIP. 19900511 20160801 1 057