PENGAMANAN PESAN MENGGUNAKAN ALGORITMA *ONE TIME PAD* (OTP) DENGAN *LINEAR CONGRUENTIAL GENERATOR* (LCG) SEBAGAI PEMBANGKIT KUNCI

SKRIPSI

OLEH:
JAMILATUL MAGHFIROH
NIM. 17610096

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

PENGAMANAN PESAN MENGGUNAKAN ALGORITMA *ONE TIME PAD* (OTP) DENGAN *LINEAR CONGRUENTIAL GENERATOR* (LCG) SEBAGAI PEMBANGKIT KUNCI

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
Jamilatul Maghfiroh
NIM. 17610096

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

PENGAMANAN PESAN MENGGUNAKAN ALGORITMA *ONE TIME PAD* (OTP) DENGAN *LINEAR CONGRUENTIAL GENERATOR* (LCG) SEBAGAI PEMBANGKIT KUNCI

SKRIPSI

Oleh
Jamilatul Maghfiroh
NIM. 17610096

Telah Diperiksa dan Disetujui untuk Diuji
Tanggal 16 Juni 2022

Pembimbing I	Pembimbing II
Prof. Dr. H. Turmudi, M.Si., Ph.D	Dr. Elly Susanti, S.Pd., M.Sc
NIP. 19571005 198203 1 006	NIP. 19741129 200012 2 005

Mengetahui
Ketua Program Studi Matematika

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

PENGAMANAN PESAN MENGGUNAKAN ALGORITMA *ONE TIME PAD* (OTP) DENGAN *LINEAR CONGRUENTIAL GENERATOR* (LCG) SEBAGAI PEMBANGKIT KUNCI

SKRIPSI

Oleh
Jamilatul Maghfiroh
NIM. 17610096

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S. Mat)
Tanggal 22 Juni 2022

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

Mengetahui
Ketua Program Studi Matematika

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

PERNYATAAN KEASLIAN TULISAN

Saya yang bertandatangan di bawah ini:

Nama : Jamilatul Maghfiroh

NIM : 17610096

Jurusan : Matematika

Fakultas : Sains dan Teknologi

Judul Skripsi : Pengamanan Pesan Menggunakan Algoritma *One Time Pad* (OTP) dengan *Linear Congruential Generator* (LCG) sebagai Pembangkit Kunci

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

Malang, 26 Mei 2022
Yang membuat pernyataan,

Jamilatul Maghfiroh
NIM 17610096

MOTO DAN PERSEMBAHAN

MOTO

لاَيُكَلِّفُ اللهُ نَفْسًا اِلَّا وُسْعَهَا

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya”

(QS. Al-Baqarah: 286)

Kemampuan seseorang terbatas namun rasa semangat tak terbatas maka lakukankan segala sesuatu bukan sampai batas kemampuanmu tetapi sampai habis semangatmu.

PERSEMBAHAN

Skripsi ini penulis persembahkan untuk:

Ayah Syamsuddin dan ibu Tarlin yang selalu memanjatkan doa-doa untuk penulis, dan juga untuk mas Suwartono selaku suami dan Abrisam selaku anak kesayangan penulis yang senantiasa memberikan semangat dan dorongan untuk terus maju.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Segala puji bagi Allah Swt. atas rahmat, taufik serta hidayah-Nya, sehinga penulis mampu menyelesaikan penyusunan skripsi ini sebagai salah satu syarat untuk memperoleh gelar sarjana dalam bidang matematika di Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Sri Harini, M.Si, selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Elly Susanti, S.Pd., M.Sc, selaku ketua Program Studi Matematika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang dan dosen pembimbing II yang telah banyak memberikan arahan dan berbagi ilmunya kepada penulis..
4. Prof. Dr. H. Turmudi, M.Si., Ph.D, selaku dosen pembimbing I yang telah banyak memberikan arahan, nasihat, motivasi, dan berbagi pengalaman yang berharga kepada penulis.
5. Seluruh dosen Program Studi Matematika, Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
6. Ayah, suami, dan ibu di surga yang turut mendoakan, memberikan semangat, serta motivasi kepada penulis sampai saat ini.

Semoga Allah SWT melimpahkan rahmat dan karunia-Nya kepada kita semua dan skripsi ini bisa bermanfaat bagi penulis dan pembaca. *Amiin.*

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Malang, 26 Mei 2022

Penulis

DAFTAR ISI

DAFTAR GAMBAR

ABSTRAK

Maghfiroh, Jamilatul. 2022. **Pengamanan Pesan Menggunakan Algoritma *One Time Pad* (OTP) dengan *Linear Congruential Generator* (LCG) sebagai Pembangkit Kunci**. Skripsi. Jurusan Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing (1) : Prof. Dr. H. Turmudi M.Si., Ph.D, Pembimbing (2) : Dr. Elly Susanti S.Pd., M.Sc.

Kata Kunci : Bilangan Acak, Enkripsi, Dekripsi, Plainteks, Chiperteks, Algoritma LCG, Algoritma OTP

Kriptografi merupakan salah satu metode yang dapat digunakan untuk mengamankan suatu pesan agar pesan tersebut tidak dapat dibaca oleh sembarang orang. Salah satu algoritma klasik yang ada dalam kriptografi adalah algoritma *One Time Pad* (OTP). OTP adalah algoritma yang menggunakan kunci sepanjang *plaintext*nya dan kunci yang digunakan harus acak oleh karena itu dibutuhkan suatu pembangkit bilangan acak agar dapat digunakan sebagai kunci. Salah satu algoritma yang dapat menghasilkan bilangan acak adalah *Linear Congruential Generator* (LCG).

Tujuan utama dari penelitian ini adalah untuk mendeskripsikan proses pembangkitan kunci menggunakan algoritma LCG, proses enkripsi pesan dengan menggunakan algoritma OTP, dan proses dekripsi pesan dengan menggunakan algoritma OTP. Metode penelitian yang digunakan adalah penelitian literatur. Adapun tahapan mengolah data dalam penelitian ada 3 yaitu proses pembangkitan kunci menggunakan algoritma LCG, proses enkripsi pesan dengan menggunakan algoritma OTP, dan proses dekripsi pesan dengan menggunakan algoritma OTP.

Hasil yang diperoleh dari penelitian ini adalah algoritma LCG dapat memenuhi kebutuhan sebagai kunci karena kunci yang dihasilkan adalah kunci yang dinamis dengan syarat panjang periodenya harus lebih besar atau sama dengan panjang *plaintex*tnya. Proses penyandian pesan menggunakan algoritma OTP memiliki tingkat keamanan yang tinggi sebab jumlah karakter yang digunakan lebih banyak dan proses pengiriman pesan lebih mudah. *Chipertext* yang dihasilkan merupakan pesan yang sangat acak dan tidak terbaca sehingga sulit dipecahkan namun dalam proses dekripsi kunci yang digunakan harus sama dengan milik pengirim.

ABSTRACT

Maghfiroh, Jamilatul. 2022. **Message Security Using One Time Pad (OTP) Algorithm with Linear Congruential Generator (LCG) as Key Generator.** Thesis. Mathematics Study Program, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Advisor (1) : Prof. Dr. H. Turmudi M.Si., Ph.D, Advisor (2) : Dr. Elly Susanti S.Pd., M.Sc.

Keywords : Random Numbers, Encryption, Decryption, Plaintext, Ciphertext, LCG Algorithm, OTP Algorithm

Cryptography is one method that can be used to secure a message so that the message cannot be read by unauthorized person. One of the classic algorithms in cryptography is the One Time Pad (OTP) algorithm. OTP is an algorithm that uses a key along the plaintext and the key used must be random, therefore a random number generator is needed to be used as a key. One of the algorithms that can generate random numbers is the Linear Congruential Generator (LCG).

The main purpose of this study is to describe the key generation process using the LCG algorithm, the message encryption process using the OTP algorithm, and the message decryption process using the OTP algorithm. The research method used is literature research. There are three stages of data processing in this research, namely the key generation process using the LCG algorithm, the message encryption process using the OTP algorithm, and the message decryption process using the OTP algorithm.

The results obtained from this study are the LCG algorithm can meet the needs as a key because the key generated is a dynamic key with the condition that the length of the period must be greater than or equal to the length of the plaintext. The process of encoding messages using the OTP algorithm has a high level of security because the number of characters used is more and the process of sending messages is easier. The resulting ciphertext is a very random and unreadable message that is difficult to decipher, but in the decryption process the key used must be the same as the sender's.

مستخلص البحث

مغفرة، جميلة. ٢٠٢٢. أمان الرسائل باستخدام خوارزمية لوحة زمنية واحدة مع المولد التطابق الخطي كمولد مفتاح. البحث

العلمي. قسم الرياضيات، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج.

المشرف (١): البروفيسور الدكتور تورمودي الحاج، المشرفة (٢): الدكتور إيلي سوسانتي الماجستير.

الكلمات الرئيسية: الأرقام العشوائية ، التشفير ، فك التشفير ، النص العادي ، النص المشفر ، خوارزمية المولد التطابق الخطي ، خوارزمية لوحة زمنية واحدة.

التشفير هو إحدى الطرق التي يمكن استخدامها لتأمين رسالة بحيث لا يمكن لأي شخص قراءة الرسالة. إحدى الخوارزميات الكلاسيكية في التشفير هي خوارزمية لوحة زمنية واحدة. لوحة زمنية واحدة عبارة عن خوارزمية تستخدم مفتاحًا على طول النص العادي والمفتاح المستخدم يجب أن يكونا عشوائيًا ، لذلك يلزم استخدام مولد رقم عشوائي كمفتاح. أحد الخوارزميات التي يمكن أن تولد أرقامًا عشوائية هو المولد التطابق الخطي.

الغرض الرئيسي من هذه الدراسة هو وصف عملية توليد المفاتيح باستخدام خوارزمية المولد التطابق الخطي، وعملية تشفير الرسائل باستخدام خوارزمية لوحة زمنية واحدة ، وعملية فك تشفير الرسائل باستخدام خوارزمية لوحة زمنية واحدة .طريقة البحث المستخدمة هي البحث الأدبي. هناك ٣ مراحل لمعالجة البيانات في هذا البحث ، وهي عملية إنشاء المفتاح باستخدام خوارزمية المولد التطابق الخطي، وعملية تشفير الرسائل باستخدام خوارزمية لوحة زمنية واحدة ، وعملية فك تشفير الرسائل باستخدام خوارزمية لوحة زمنية واحدة.

النتائج التي تم الحصول عليها من هذه الدراسة هي أن خوارزمية المولد التطابق الخطي يمكنها تلبية الاحتياجات كمفتاح لأن المفتاح الذي تم إنشاؤه هو مفتاح ديناميكي بشرط أن يكون طول الفترة أكبر من أو يساوي طول النص العادي. تتمتع عملية تشفير الرسائل باستخدام خوارزمية لوحة زمنية واحدة بمستوى عالٍ من الأمان لأن عدد الأحرف المستخدمة أكثر وتكون عملية إرسال الرسائل أسهل. النص المشفر الناتج عبارة عن رسالة عشوائية جدًا وغير قابلة للقراءة ويصعب فك تشفيرها ، ولكن في عملية فك التشفير ، يجب أن يكون المفتاح المستخدم هو نفسه مفتاح المرسل.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Informasi merupakan salah satu aspek penting dalam kehidupan manusia. Informasi sudah mendarah daging di era modern saat ini, banyak sekali hal yang dilakukan dengan melibatkan penukaran, pengumpulan, pengaksesan atau pembuatan informasi. Informasi dapat berupa data diri seseorang, data diri perusahaan, pengumuman, berita atau dapat berupa sebuah pesan penting. Informasi bisa bersifat rahasia atau umum. Adapun informasi yang bersifat rahasia sangatlah penting untuk dijaga keamanan kerahasiaannya. Pada era modern saat ini, perkembangan teknologi kian meningkat, yang mengakibatkan kemampuan dalam mengakses suatu informasi menjadi sangatlah mudah. Oleh karena itu, keamanan merupakan hal yang sangat penting untuk diperhatikan, terutama dalam hal pengiriman informasi atau pesan-pesan rahasia agar pesan bisa sampai kepada pihak yang dituju dan tidak sampai jatuh kepada pihak ketiga.

Sebagaimana yang telah difirmankan oleh Allah SWT dalam surat An – Nisa' ayat 58 :

اِنَّ الله يَأْمُرُكُمْ اَنْ تُؤَدُّوا ٱلاَمٰنٰتِ اِلىٰ اَهْلِهَا ۙ وَاِذَا حَكَمْتُمْ بَيْنَ النَّاسِ اَنْ تَحْكُمُوْا بِالعَدْلِ ۗ اِنَّ اللهَ كَانَ سَمِيْعًا بَصِيْرًا

Artinya: *"Sesungguhnya Allah menyuruh kamu menyampaikan amanat kepada yang berhak menerimanya, dan apabila kamu menetapkan hukum di antara manusia hendaknya kamu menetapkannya dengan adil. Sesungguhnya Allah sebaik-baiknya yang memberi pengajaran kepadamu. Sesungguhnya Allah Maha Mendengar lagi Maha Melihat"(Q.S. An-Nisa':58).*

Dalam surat An – Nisa' ayat 58 telah dijelaskan bahwa Allah SWT memerintahkan kita semua untuk menyampaikan amanah kepada seseorang yang berhak menerimanya. Korelasi dari ayat tersebut dengan penjelasan sebelumnya adalah pada kata "Amanah". Pesan rahasia bisa juga dikatakan "Amanah", oleh karenanya kita harus tetap menjaga kerahasiaan suatu pesan tersebut hingga pesan itu sampai ke pihak yang berhak menerimanya. Dalam menjaga keamanan dan mempertahankan kerahasiaan suatu pesan diperlukan metode untuk penyandian pesan sehingga pesan terhindar dari penyadapan dan pencurian data oleh pihak yang tidak bertanggungjawab. Kriptografi merupakan salah satu metode yang dapat digunakan untuk mengamankan suatu pesan agar pesan tersebut tidak dapat dibaca oleh sembarang orang.

Kriptografi dapat diartikan sebagai proses mengubah teks biasa menjadi teks yang tidak dapat dipahami dan sebaliknya. Ini adalah metode penyembunyian pesan dalam bentuk tersandi sehingga hanya mereka yang dimaksudkan yang dapat membaca dan memprosesnya. Pesan atau teks asli sebelum dilakukan proses apapun disebut *plaintext* atau *cleartext*. Proses mengubah *plaintext* menjadi bentuk rahasia disebut enkripsi. Setelah teks asli dienkripsi, teks yang dihasilkan dikenal sebagai *ciphertext* atau *cryptogram*. Proses pengubahan *ciphertext* menjadi *plaintext* dikenal sebagai dekripsi. Dalam proses enkripsi, sebagian besar algoritma matematika digunakan (Munir, 2005).

Salah satu algoritma klasik yang ada dalam kriptografi adalah algoritma *One Time Pad* (OTP). Algoritma kriptografi klasik memiliki bentuk yang sederhana. Algoritma OTP juga tergolong kedalam algoritma kriptografi simetris yang artinya algoritma OTP menggunakan kunci yang sama selama proses enkripsi dan

dekripsinya. Alasan pengambilan nama "*One Time Pad*" dari algoritma ini karena pada saat proses enkripsi dan dekripsi kunci yang digunakan dari algoritma ini hanya boleh digunakan satu kali (*one time*) dan setelah digunakan untuk proses enkripsi dan dekripsi maka kertas *blocknote* (*pad*) harus dihanguskan agar tidak dapat dipakai kembali untuk penyandian pesan yang lain. OTP adalah algoritma yang menggunakan kunci sepanjang *plaintext*nya dan kunci yang digunakan benar-benar acak sehingga menghasilkan *ciphertext* yang juga benar-benar acak (Mollin, 2007), oleh karena itu dibutuhkan suatu pembangkit bilangan acak agar kunci yang dihasilkan berupa barisan bilangan yang benar-benar acak.

Salah satu algoritma yang dapat menghasilkan bilangan acak adalah *Linear Congruential Generator* (LCG). Algoritma ini digunakan untuk menghindari pembuatan kunci berulang. Dalam proses pembangkitan kunci, semakin panjang periodenya maka semakin kecil kemungkinan kunci tersebut diulang sehingga menghasilkan kunci yang dinamis. Algoritma pembangkit bilangan acak memiliki peran penting dalam kriptografi karena memiliki ketahanan terhadap serangan kriptanalisis (Stallings, 2005).

Pada penelitian ini algoritma yang digunakan peneliti untuk mengamankan sebuah pesan rahasia adalah algoritma OTP. OTP digunakan karena merupakan algoritma sederhana namun kuat dengan tingkat keamanan yang tinggi sehingga tidak memungkinkan untuk dipecahkan dengan metode kriptanalitik apapun (Mollin, 2007). Sedangkan LCG, algoritma tersebut digunakan karena memiliki kecepatan, kemudahan implementasi, dan sedikit menggunakan operasi bit. Dengan adanya penelitian ini, diharapkan LCG dapat menjadi pembangkit kunci yang

benar-benar acak pada algoritma OTP sehingga dapat digunakan untuk mengamankan pesan rahasia dengan hasil *ciphertext* yang sulit dipecahkan.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka diperoleh rumusan masalah sebagai berikut:

1. Bagaimana proses membangkitan kunci menggunakan algoritma LCG?
2. Bagaimana proses enkripsi pesan rahasia menggunakan metode OTP dengan kunci yang dibangkitkan menggunakan LCG?
3. Bagaimana proses dekripsi pesan rahasia menggunakan metode OTP dengan kunci yang dibangkitkan menggunakan LCG?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah diatas, maka diperoleh tujuan penelitian sebagai berikut:

1. Untuk mendeskripsikan proses pembangkitan kunci menggunakan algoritma LCG.
2. Untuk mendeskripsikan proses enkripsi pesan rahasia menggunakan metode OTP dengan kunci yang dibangkitkan menggunakan LCG.
3. Untuk mendeskripsikan proses dekripsi pesan rahasia menggunakan metode OTP dengan kunci yang dibangkitkan menggunakan LCG.

1.4 Manfaat Penelitian

Adapun manfaat yang diperoleh dari penelitian ini adalah dapat menambah wawasan dan pengetahuan tentang ruang lingkup kriptografi terutama pada algoritma *One Time Pad* (OTP), *Linear Congruential Generator* (LCG), dan pembangkit bilangan acak.

1.5 Batasan Masalah

Agar penelitian ini lebih terarah dan terfokus pada sasaran maka penulis memberikan batasan masalah sebagai berikut:

1. Dalam proses membangkitkan kunci, nilai parameter yang digunakan adalah $b \neq 0$ dan $a \neq 1$.
2. *Plaintext* atau pesan asli dikonversikan ke nilai karakter menurut tabel ASCII.
3. Nilai modulo (n) pada proses enkripsi pesan menggunakan algoritma *One Time Pad* menggunakan nilai = 256 (disesuaikan dengan jumlah karakter pada tabel ASCII).
4. Hasil enkripsi berupa *ciphertext* dikonversikan ke nilai karakter menurut tabel ASCII.

1.6 Definisi Istilah

Dalam penelitian ini perlu adanya batasan-batasan pengertian untuk menghindari kesalahpahaman. Istilah- istilahnya sebagai berikut:

1. ASCII singkatan dari *American Standard Code for Information Interchange* adalah standar penkodean karakter untuk alat komunikasi.
2. Invers modulo adalah lawan atau kebalikan dari modulo tersebut.
3. Kecil pada kalimat Teorema Kecil Fermat bukan berarti arti kecil pada umumnya, melainkan dimaksudkan untuk membedakan dengan Teorema Fermat Terakhir.
4. Konversi adalah perubahan dari suatu bentuk ke bentuk yang lain.
5. VT adalah *Vertical Tab,* yang mana memiliki nilai Unicode (heksadesimal) 0B dan nilai ASCII (desimal) sebesar 11.

BAB II

KAJIAN PUSTAKA

2.1 Teori Pendukung

Teori pendukung merupakan sekumpulan teori dari berbagai ilmu yang digunakan untuk mendukung dalam mengumpulkan informasi untuk menjawab pertanyaan pada rumusan masalah. Didalam teori pendukung terdapat 3 subbab yakni pembangkit bilangan acak, dasar teori bilangan, dan kriptografi. Pembangkit bilangan acak digunakan sebagai landasan teori untuk pemilihan algoritma yang akan digunakan sebagai kunci dalam proses penyandian pesan. Dasar teori bilangan digunakan sebagai landasan teori untuk membuktikan teorema pada proses membangkit bilangan acak. Dan kriptografi digunakan sebagai landasan teori untuk pemilihan algoritma yang akan digunakan dalam proses penyandian pesan.

2.1.1 Pembangkit Bilangan Acak

a. Definisi Pembangkit Bilangan Acak

Pembangkit bilangan acak merupakan komponen penting dari sistem kripto. Mereka digunakan untuk menghasilkan kunci simetris, menyediakan vektor inisialisasi, dan untuk tujuan lain. Pembangkit bilangan acak yang biasanya digunakan adalah *Pseudo-Random Number Generator* (PRNG). *Pseudo-Random Number Generator* (PRNG) adalah algoritma yang menghasilkan barisan bilangan acak yang baik tetapi pada akhirnya bilangan tersebut berulang. Barisan yang dihasilkan oleh PRNG adalah "pseudo-random" dalam arti bahwa ia dihasilkan oleh fungsi deterministic.

Nilai awal PRNG disebut umpan *(seed)*. Kualitas PRNG diukur dengan keacakan barisan yang dihasilkan. Dalam menentukan kualitas tersebut dapat dilakukan dengan menjalankan beberapa jenis rangkaian uji keacakan bilangan seperti *1-D test*, *Equidistribution*, *Runs test*, *Statistical test*, dan masih banyak lagi (Easttom, 2021).

b. Macam-Macam Pembangkit Bilangan Acak

Macam - macam pembangkit bilangan acak diantaranya :

1) Pembangkit Bilangan Acak Seragam
   a. *Linear Congruential Generators* (LCG),
   b. *Uniform Random Numbers*,
   c. *Mersenne Twiters*,
   d. *Xorshift*,
   e. *Quasirandom Generators*, dll.
2) Pembangkit Bilangan Acak Tak Seragam
   a. *Nonuniform Random Numbers*,
   b. *Normal Distribution*,
   c. *Binomial Distribution*,
   d. *Exponential Distribution*,
   e. *Poissom Distribution*, dll (Kneusel, 2018).

c. Karakteristik Pembangkit Bilangan Acak

Pembangkit bilangan acak yang ‘baik’ harus memenuhi karakteristik berikut ini:

1. Keseragaman: bilangan yang dibangkitkan didistribusikan secara seragam pada $(0,1)$.

2. Independensi: bilangan yang dibangkitkan tidak menunjukkan korelasi satu sama lain.
3. Panjang periode (*long period*): lama waktu yang dibutuhkan sebelum bilangan mulai berulang. Semakin lama waktu yang dibutuhkan sampai bilangan berulang maka barisan acak tersebut akan semakin baik (Easttom, 2021).

2.1.2 Dasar Teori Bilangan

a. Kekongruenan

Definisi

Sistem residu lengkap dari modulo m adalah himpunan bilangan bulat sedemikian sehingga setiap bilangan bulat kongruen modulo m tepat satu bilangan bulat dari himpunan (Rossen, 2000).

Lemma 2.1.1

Jika a, b dan c bilangan bulat positif sedemikian sehingga $gcd(a, b) = 1$ dan $a|bc$, maka $a|c$.

Bukti:

Jika $a|bc$ maka $bc = ak, \exists k \in \mathbb{Z}$, karena $gcd(a, b) = 1$, ada bilangan bulat x dan y sedemikian hingga $ax + by = 1$. Kalikan kedua sisi dengan c, diperoleh

$$acx + bcy = c$$

$$acx + aky = c$$

$$a(cx + ky) = c$$

Dimana $cx + ky \in \mathbb{Z}$. Oleh karenanya $a|c$ (Rossen, 2000).

Proposisi 2.1.1

Misalkan a, b dan c bilangan bulat dengan $gcd(a, b) = d$. Maka

(i) $\left(\frac{a}{d}, \frac{b}{d}\right) = 1$

(ii) $(a + cb, b) = (a, b)$

Bukti:

(i) Misalkan a dan b bilangan bulat dengan $gcd(a, b) = d$. Diketahui bahwa $\frac{a}{d}$ dan $\frac{b}{d}$ tidak memiliki pembagi positif yang sama selain 1. Asumsikan bahwa e adalah bilangan bulat positif sehingga $e|(\frac{a}{d})$ dan $e|(\frac{b}{d})$. Maka, ada bilangan bulat k dan ℓ dengan $\frac{a}{d} = ke$ dan $\frac{b}{d} = \ell e$, sehingga $a = dek$ dan $b = de\ell$. Jadi, de adalah pembagi persekutuan dari a dan b. Karena d adalah pembagi persekutuan terbesar dari a dan b, maka e harus 1. Akibatnya $\left(\frac{a}{d}, \frac{b}{d}\right) = 1$.

(ii) Misalkan a, b dan c bilangan bulat. Akan ditunjukkan bahwa pembagi persekutuan dari a dan b sama persis dengan pembagi persekutuan dari $a + cb$ dan b. Ini akan menunjukkan bahwa $(a + cb, b) = (a, b)$. Misalkan e adalah pembagi persekutuan dari a dan b. Kita melihat bahwa $e|(a + cb)$, sehingga e adalah pembagi persekutuan dari $a + cb$ dan b. Jika f adalah pembagi persekutuan dari $a + cb$ dan b, maka dengan dapat dilihat bahwa $f|(a + cb) - cb = a$, sehingga f adalah pembagi persekutuan dari a dan b. Jadi $(a + cb, b) = (a, b)$ (Rossen, 2000).

Teorema 2.1.1

Jika a, b, c dan m bilangan bulat sedemikian sehingga $m > 0$, $d = gcd(c, m)$, dan $ac \equiv bc \ (mod\ m)$, maka $a \equiv b \ (mod \frac{m}{d})$.

Catatan : Jika $a \equiv b \ (mod \frac{m}{d})$ maka $a - b \equiv 0 \ (mod \frac{m}{d})$ yang berarti bahwa $a - b$ adalah kelipatan dari $\frac{m}{d}$. Karena $a, b \in \mathbb{Z}$ maka $(a - b) \in \mathbb{Z}$ sehingga $\frac{m}{d} \in \mathbb{Z}$.

Bukti:

Jika $ac \equiv bc \ (mod \ m)$, kita tahu bahwa $m|(ac - bc) = c(a - b)$. Dengan membagi kedua sisi dengan d, kita punya $\frac{m}{d} | \frac{c}{d}(a - b)$. Selanjutnya $\left(\frac{m}{d}, \frac{c}{d}\right) = 1$ (menurut proposisi 2.1.1) menunjukkan bahwa $\frac{m}{d} | (a - b)$ (menurut Lemma 2.1.1). Oleh karena itu, $a \equiv b \ (mod \frac{m}{d})$ (Rossen, 2000).

Akibat 2.1.1

Jika a, b, c dan m bilangan bulat sedemikian sehingga $m > 0$, $gcd(c, m) = 1$, dan $ac \equiv bc \ (mod \ m)$, maka $a \equiv b \ (mod \ m)$ (Rossen, 2000).

b. Teorema Fundamental Aritmatika

Teorema 2.1.2 (Teorema Fundamental Aritmatika)

Setiap bilangan bulat positif $n > 1$ selalu dapat disajikan dalam bentuk perkalian bilangan-bilangan prima berpangkat. Representasi ini tunggal terhadap urutan factor-faktornya yaitu

$$n = {p_1}^{\alpha_1} {p_2}^{\alpha_2} \dots {p_k}^{\alpha_k}$$

Dimana $p_1 p_2 \dots p_k$ bilangan prima yang berbeda dan $\alpha_1 \alpha_2 \dots \alpha_k$ bilangan bulat positif.

Bukti:

Akan ditunjukkan bahwa setiap bilangan bulat positif dapat dituliskan sebagai bentuk perkalian bilangan prima. Asumsikan bahwa beberapa

bilangan bulat positif tidak dapat dituliskan sebagai perkalian bilangan prima. Misalkan n menjadi bilangan bulat terkecil. Jika n prima, jelas merupakan perkalian dari himpunan bilangan prima, jadi n haruslah komposit. Misalkan $a, b \in \mathbb{Z}$ sehingga $n = ab$, dimana $1 < a < n$ dan $1 < b < n$, karena a dan b lebih kecil dari n maka keduanya dapat menjadi perkalian bilangan prima. Jadi, karena $n = ab$ maka dapat disimpulkan bahwa n juga merupakan perkalian dari bilangan prima. Hal ini berkontradiksi dengan asumsi, ini menyiratkan bahwa setiap bilangan bulat positif dapat dituliskan sebagai perkalian bilangan prima (Rossen, 2000).

c. Teorema Binomial

Lemma 2.1.2

Misalkan n dan k bilangan bulat positif dengan $n \geq k$. Maka

$$\binom{n}{k} + \binom{n}{k-1} = \binom{n+1}{k}$$

Bukti:

Lakukan penambahan

$$\binom{n}{k} + \binom{n}{k-1} = \frac{n!}{k!\,(n-k)!} + \frac{n!}{(k-1)!\,(n-k+1)!}$$

Dengan menggunakan factor persekutuan $k!\,(n-k+1)!$ Diperoleh

$$\binom{n}{k} + \binom{n}{k-1} = \frac{n!(n-k+1)}{k!(n-k+1)!} + \frac{n!k}{k!(n-k+1)!}$$

$$= \frac{n!\big((n-k+1)+k\big)}{k!(n-k+1)!}$$

$$= \frac{n!(n+1)}{k!(n-k+1)!}$$

$$= \frac{(n+1)!}{k!(n-k+1)!}$$

$$= \binom{n+1}{k}$$

Teorema 2.1.3 (Teorema Binomial)

Misalkan x dan y menjadi variable dan n adalah bilangan bulat positif. Maka,

$$(x+y)^n = \binom{n}{0}x^n + \binom{n}{1}x^{n-1}y + \binom{n}{2}x^{n-2}y^2 + \cdots + \binom{n}{n-2}x^2y^{n-2} + \binom{n}{n-1}xy^{n-1} + \binom{n}{n}y^n$$

Atau menggunakan notasi penjumlahan

$$(x+y)^n = \sum_{j=0}^{n}\binom{n}{j}x^{n-j}y^j.$$

Bukti:

Dengan menggunakan induksi matematika. Ketika $n = 1$, diperoleh

$$(x+y)^1 = \binom{1}{0}x^1y^0 + \binom{1}{1}x^0y^1$$

Karena $\binom{1}{0} = \binom{1}{1} = 1$, ini menyiratkan bahwa $(x+y)^1 = x+y$, jelas bahwa ini benar. Sekarang asumsikan bahwa teorema ini benar untuk bilangan bulat positif n, maka

$$(x+y)^n = \sum_{j=0}^{n}\binom{n}{j}x^{n-j}y^j$$

Sekarang pertimbangkan untuk $n+1$, diperoleh

$$\begin{aligned}(x+y)^{n+1} &= (x+y)^n(x+y)\\ &= \left[\sum_{j=0}^{n}\binom{n}{j}x^{n-j}y^j\right](x+y)\\ &= \sum_{j=0}^{n}\binom{n}{j}x^{n-j+1}y^j + \sum_{j=0}^{n}\binom{n}{j}x^{n-j}y^{j+1}\end{aligned}$$

$$\sum_{j=0}^{n}\binom{n}{j}x^{n-j+1}y^j = x^{n+1} + \sum_{j=1}^{n}\binom{n}{j}x^{n-j+1}y^j$$

$$\sum_{j=0}^{n}\binom{n}{j}x^{n-j}y^{j+1} = \sum_{j=0}^{n-1}\binom{n}{j}x^{n-j}y^{j+1} + y^{n+1}$$

$$= \sum_{j=1}^{n}\binom{n}{j-1}x^{n-j+1}y^{j} + y^{n+1}$$

Didapatkan

$$(x+y)^{n+1} = x^{n+1} + \sum_{j=1}^{n}\left[\binom{n}{j} + \binom{n}{j-1}\right]x^{n-j+1}y^{j} + y^{n+1}$$

Dengan menggunakan Lemma 2.1.2, kita punya

$$\binom{n}{j} + \binom{n}{j-1} = \binom{n+1}{j}$$

Dapat disimpulkan bahwa

$$(x+y)^{n+1} = x^{n+1} + \sum_{j=1}^{n}\binom{n+1}{j}x^{n-j+1}y^{j} + y^{n+1}$$

$$= \sum_{j=0}^{n+1}\binom{n+1}{j}x^{n+1-j}y^{j}.$$

d. Teorema *Fermat's Little*

Teorema 2.1.4 (Teorema *Fermat's Little*)

Jika p prima dan a bilangan bulat positif, nilai $a^p - a$ adalah kelipatan p dan dinotasikan dengan

$$a^p \equiv a \ (mod\ p)$$

Jika a tidak habis dibagi oleh p yang berarti $(a, p) = 1$, maka sama dengan $a^{p-1} - 1$ adalah kelipatan p dan dinotasikan dengan $a^{p-1} \equiv 1 \ (mod\ p)$.

Bukti:

Perhatikan bilangan bulat $a, 2a, \dots, (p-1)a$. Tak satu pun dari bilangan bulat ini yang habis dibagi p, tiap suku memiliki sisa yang berbeda jika dimodulokan p. Akibatnya diperoleh

$$a(2a), \dots, \big((p-1)a\big) \equiv 1.2 \dots . (p-1) \ (mod\ p)$$

Oleh karenanya,

$$(p-1)!\, a^{p-1} \equiv (p-1)!\, (mod\ p)$$

Karena $\big((p-1)!, p\big) = 1$ menggunakan Akibat 2.1.1 kita batalkan $(p-1)!$ Untuk mendapatkan $a^{p-1} \equiv 1\ (mod\ p)$ (Rossen, 2000).

2.1.3 Kriptografi

a. Pengertian Kriptografi

Kriptografi merupakan studi tentang metode untuk mengirim pesan secara rahasia dalam bentuk tersandi atau disamarkan sehingga hanya penerima yang dituju yang dapat menghapus penyamaran dan membaca pesan. Secara etimologi, kriptografi tersusun dari 2 kata bahasa Yunani yakni kryptos, yang berarti tersembunyi, dan graphein, yang berarti menulis. Pesan asli disebut *plaintext*, dan pesan terselubung disebut *ciphertext*. Pesan terakhir, dikemas dan dikirim, disebut kriptogram. Proses mengubah teks biasa menjadi teks sandi disebut enkripsi atau penyandian. Proses mengubah *ciphertext* menjadi *plaintext* yang dilakukan oleh penerima yang memiliki metode untuk menghapus penyamaran disebut dekripsi atau penguraian. Di sisi lain, studi tentang teknik matematika untuk mencoba mengalahkan metode kriptografi disebut kriptanalisis. Mereka yang mempraktikkan kriptanalisis disebut kriptanalis (Mollin, 2007 : 79).

Kriptografi memainkan peran penting dalam komunikasi data rahasia hari ini dan masa depan. Misalnya, komunikasi melalui saluran telepon termasuk faks dan pesan email, transaksi keuangan, riwayat kesehatan, e-banking dan bahkan jenis informasi penting lainnya memerlukan media

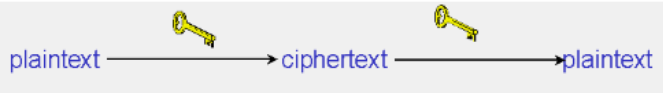
komunikasi yang aman. Terkadang, media diretas oleh penyusup dan mendapatkan semua informasi. Oleh karena itu, aplikasi kriptografi menyediakan media komunikasi yang aman untuk mentransfer data sehingga jika ada yang mencoba meretas data, maka hal itu tidak berguna baginya karena datanya dalam bentuk terenkripsi (Munir, 2005).

b. Macam-Macam Algoritma Kriptografi

Menurut Gary (2010) Algoritma kriptografi terbagi menjadi 3 bagian berdasarkan kunci yang dipakai:

1. Kriptografi Simetris

Metode kriptografi kunci rahasia merupakan metode yang menggunakan satu kunci untuk enkripsi dan dekripsi. Seperti yang ditunjukkan pada Gambar 1, pengirim menggunakan kunci untuk mengenkripsi *plaintext* dan mengirimkan *ciphertext* ke penerima. Penerima menerapkan kunci yang sama untuk mendekripsi pesan dan memulihkan *plaintext*. Karena kunci tunggal digunakan untuk kedua fungsi tersebut, kriptografi kunci rahasia juga disebut kriptografi simetris.



Gambar 2.1.1 Kriptografi Simetris

Dengan bentuk kriptografi ini, jelas bahwa kunci harus diketahui oleh pengirim dan penerima; itulah rahasianya. Beberapa algoritma yang menggunakan kunci simetris adalah:

a. *Data Encryption Standard* (DES),
b. RC2, RC4, RC5, RC6,

c. *International Data Encryption Algorithm* (IDEA),

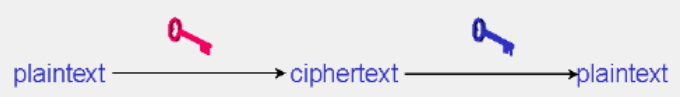
d. *Advanced Encryption Standard* (AES),

e. *One Time Pad* (OTP), dll

2. Kriptografi Asimetris

Kriptografi kunci publik telah dikatakan sebagai perkembangan baru yang paling signifikan dalam kriptografi dalam 300-400 tahun terakhir. Kriptografi kunci publik modern pertama kali dijelaskan secara terbuka oleh profesor Universitas Stanford Martin Hellman dan mahasiswa pascasarjana Whitfield Diffie pada tahun 1976. Makalah mereka menggambarkan sistem kripto dua kunci di mana dua pihak dapat terlibat dalam komunikasi yang aman melalui saluran komunikasi yang tidak aman tanpa harus berbagi kunci rahasia.

Kriptografi kunci publik generik menggunakan dua kunci yang secara matematis terkait meskipun mengetahui satu kunci tidak memungkinkan seseorang untuk dengan mudah menentukan kunci lainnya. Satu kunci digunakan untuk mengenkripsi *plaintext* dan kunci lainnya digunakan untuk mendekripsi *chphertext*. Poin penting di sini adalah bahwa tidak masalah kunci mana yang diterapkan pertama kali, tetapi kedua kunci tersebut diperlukan agar proses dapat berjalan (Gambar 2). Karena diperlukan sepasang kunci, pendekatan ini juga disebut kriptografi asimetris.



Gambar 2.1.2 Kriptografi Asimetris

Beberapa algoritma yang menggunakan kunci asimetris adalah:

a. *Digital Signature Algorithm* (DSA),

b. RSA,

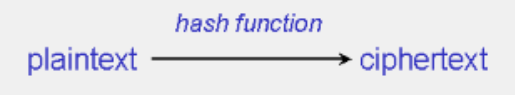
c. *Diffie-Hellman* (DH),

d. *Elliptic Curve Cryptography* (ECC),

e. Kriptografi Quantum, dll

3. Fungsi Hash

Fungsi hash, juga disebut intisari pesan dan enkripsi satu arah, adalah algoritma yang pada dasarnya tidak menggunakan kunci (Gambar 3). Algoritma hash biasanya digunakan untuk memberikan sidik jari digital dari konten file, sering kali digunakan untuk memastikan bahwa file tersebut tidak diubah oleh penyusup atau virus. Fungsi hash juga biasanya digunakan oleh banyak sistem operasi untuk mengenkripsi kata sandi. Fungsi hash juga menyediakan mekanisme untuk memastikan integritas file.



Gambar 2.1.3 Fungsi Hash

2.2 Kajian Integrasi Topik Dengan Al-Qur'an/Hadits

Allah SWT berfirman dalam surat An – Nisa' ayat 58 yang berbunyi:

اِنَّ اللّٰهَ يَأْمُرُكُمْ اَنْ تُؤَدُّوا الْاَمٰنٰتِ اِلٰٓى اَهْلِهَا ۙ وَاِذَا حَكَمْتُمْ بَيْنَ النَّاسِ اَنْ تَحْكُمُوْا بِالْعَدْلِ ۗ اِنَّ اللّٰهَ كَانَ سَمِيْعًا بَصِيْرًا

Artinya: *"Sesungguhnya Allah menyuruh kamu menyampaikan amanat kepada yang berhak menerimanya, dan apabila kamu menetapkan hukum di antara manusia hendaknya kamu menetapkannya dengan adil. Sesungguhnya Allah sebaik-baiknya yang memberi pengajaran kepadamu. Sesungguhnya Allah Maha Mendengar lagi Maha Melihat" (Q.S. An-Nisa':58).*

Dalam surat An – Nisa' ayat 58 telah dijelaskan bahwa Allah SWT memerintahkan kita semua untuk menyampaikan amanah kepada seseorang yang berhak menerimanya. Makna amanah dalam surat ini mencakup semua jenis amanah. Amanah dikategorikan menjadi tiga bagian. Pertama, amanah kepada hak-hak Tuhannya. Dalam hal ini, Allah memerintahkan agar manusia memegang amanah nya sebagai seorang hamba yakni dengan menjalankan perintah-perintahNya serta menjauhi laranganNya seperti melaksanakan sholat, zakat, tidak meminum alkohol, dll. Kedua, amanah kepada hak manusia seperti menyimpan aib, menepati janji, memegang jabatan dan masih banyak lagi. Ketiga, amanah kepada dirinya sendiri seperti menjaga diri agar tidak kalah dengan hawa nafsunya dan menjauhkan diri dari sesuatu yang merugikan diri sendiri (Damasyqi, 2002).

Korelasi dari ayat tersebut dengan kriptografi adalah pada kata "Amanah". Pesan rahasia bisa juga dikatakan "Amanah", oleh karenanya kita harus tetap menjaga kerahasiaan suatu pesan tersebut hingga pesan itu sampai ke pihak yang berhak menerimanya. Kriptografi merupakan salah satu ilmu yang dapat menjaga keamanan dan mempertahankan kerahasiaan suatu pesan dengan cara menyandikan pesan sehingga pesan menjadi tak terbaca dan hanya penerima yang mampu membaca pesannnya.

2.3 Kajian Topik Dengan Teori Pendukung

Pada teori pendukung diketahui bahwa pembangkit bilangan acak digunakan sebagai landasan teori untuk pemilihan algoritma yang akan digunakan sebagai kunci dalam proses penyandian pesan. Dalam hal ini, algoritma yang akan digunakan adalah *Linear Congruential Generators* (LCG) yang mana merupakan pembangkit bilangan acak seragam. Diketahui bahwa kriptografi digunakan sebagai

landasan teori untuk pemilihan algoritma yang akan digunakan dalam proses penyandian pesan. Dalam hal ini, algoritma yang akan digunakan adalah *One Time Pad* (OTP) yang mana merupakan kriptografi simetris. Pada subbab ini akan dijelaskan definisi dari algoritma, dan bagaimana algoritma tersebut bekerja.

2.3.1 *Linear Congruential Generator* (LCG)

a. Definisi *Linear Congruential Generator* (LCG)

Algoritma yang pertama kali diusulkan oleh D.H Lehmer dikenal sebagai metode kongruensial linier. Metode kongruensial linier adalah algoritma yang paling terkenal dan paling banyak digunakan untuk menghasilkan bilangan acak. *Linear Congruential Generators* adalah pembangkit bilangan acak klasik dengan distribusi seragam (Stallings, 2005).

LCG didefinisikan sebagai berikut:

$$X_n = (a * X_{n-i} + c)\ (mod\ m)$$

Dimana m adalah modulo dengan nilai $m > 0$, a adalah faktor pengali dengan nilai $0 < a < m$, b adalah penambah atau *increament* dengan nilai $0 \leq c < m$, dan X_0 sebagai kunci pembangkit atau *seed* dimana $0 \leq X_0 < m$. X_n adalah barisan bilangan acak dengan $n = 1,2, ...$, dan X_{n-i} adalah bilangan acak sebelum X_n dengan $i = 1,2, ...$, (Kneusel, 2018).

Contoh :

Diambil nilai parameter $a = 5, c = 3, m = 26,$ dan $X_0 = 8,$ maka didapatkan hasil perhitungan LCG sebagai berikut:

$$X_1 = (5 \cdot 8 + 3)\ (mod\ 26) = 17$$

$$X_2 = (5 \cdot 17 + 3)\ (mod\ 26) = 10$$

$$X_3 = (5 \cdot 10 + 3)\ (mod\ 26) = 1$$

$$X_4 = (5 \cdot 1 + 3)\ (mod\ 26) = 8$$

$$X_5 = (5 \cdot 8 + 3)\ (mod\ 26) = 17$$

$$X_6 = (5 \cdot 17 + 3)\ (mod\ 26) = 10$$

Pada perhitungan LCG diatas dapat dilihat bahwa dengan pengambilan nilai parameter $a = 5, c = 3, m = 26,$ dan $X_0 = 8$ dihasilkan barisan bilangan acak dengan panjang periodenya adalah 4. Barisan bilangan tersebut dikatakan kurang baik karena periodenya sangatlah pendek, maka diperlukan syarat-syarat dan kondisi tertentu agar memiliki periode penuh.

b. Teorema-Teorema *Linear Congruential Generator* (LCG)

Teorema-teorema yang akan disebutkan merupakan teorema yang dapat memenuhi kondisi yang diperlukan dan cukup untuk barisan *linear congruential* agar mencapai panjang periode penuh.

Lemma 2.3.1

Misalkan $\{x_i\}$ menjadi barisan kongruen linier yang ditentukan oleh $x_0, a, c,$ dan m didefinisikan sebagai berikut

$$x_{i+1} \equiv ax_i + c\ (mod\ m),\ \ 0 \leq x_{i+1} < m \qquad (1.1)$$

Asumsikan $a \geq 2$ maka

$$x_{i+k} \equiv a^k x_i + \frac{(a^k - 1)c}{a - 1}(mod\ m), \forall k \geq 0 \qquad (1.2)$$

Faktanya, suku pada barisan $\{x_i\}$ diberikan sebagai berikut

$$x_k \equiv a^k x_0 + \frac{(a^k - 1)c}{a - 1}(mod\ m), \forall k \geq 0 \qquad (1.3)$$

Bukti:

Dengan menggunakan induksi matematika pada k. Pada kasus $k = 0$, sudah jelas bahwa barisan (1.2) benar karena

$$x_{i+0} \equiv a^0 x_i + \frac{(a^0 - 1)c}{a - 1} (mod\ m)$$

Asumsikan (1.2) benar untuk suku $(i + k)^{th}$ pada barisan $\{x_i\}$, dan pertimbangkan suku $(i + k + 1)^{st}$. Kita punya

$$\begin{aligned} x_{i+(k+1)} &\equiv a x_{i+k} + c\ (mod\ m) \\ &\equiv a\left(a^k x_i + \tfrac{(a^k-1)c}{a-1}\right) + c\ (mod\ m) \\ &\equiv a^{k+1} x_i + \left(a\tfrac{(a^k-1)}{a-1} + 1\right) c\ (mod\ m) \\ &\equiv a^{(k+1)} x_i + \left(\tfrac{(a^{(k+1)}-1)c}{a-1}\right)\ (mod\ m) \end{aligned}$$

Dapat dilihat bahwa (1.2) benar untuk suku $(i + k + 1)^{st}$ pada barisan $\{x_i\}$. Oleh karena itu, dengan menggunakan induksi matematika, (1.2) benar untuk semua $k \geq 0$. Dari lemma ini dapat disimpulkan bahwa subbarisan yang terdiri dari setiap suku k barisan $\{x_i\}$ itu sendiri adalah barisan kongruensi linier dengan pengali $a^k (mod\ m)$ dan penambah $\frac{(a^k-1)c}{a-1} (mod\ m)$ (Glen, 2002).

Catatan:

$a, a^k \in \mathbb{Z}$ oleh karenanya $a - 1, a^k - 1 \in \mathbb{Z}$ (menurut sifat tertutup operasi pengurangan). Dengan menggunakan definisi dari keterbagian maka $\frac{(a^k-1)}{a-1}$ dapat dinyatakan dengan $(a - 1)|(a^k - 1)$. Hal ini menyiratkan bahwa $\frac{(a^k-1)}{a-1} \in \mathbb{Z}$ jika $(a - 1)|(a^k - 1)$.

Bukti:

Akan ditunjukkan bahwa $(a-1)|(a^k-1)$ dimana $\forall k \in \mathbb{N}$. Untuk setiap $k \geq 1$. Misalkan $S(k)$ didefinisikan sebagai berikut

$$S(k): (a-1)|(a^k-1); \forall k \in \mathbb{N}$$

Perhatikan bahwa $S(k)$ adalah kasus khusus dari pernyataan yang lebih umum yakni:

$$P(k): (p-q)|(p^k-q^k); \forall k \in \mathbb{N}, p \neq q$$

Maka $S(k)$ benar ketika $p = a$ dan $q = 1$.

Akan dibuktikan bahwa $P(k)$ benar dengan menggunakan induksi matematika pada k.

Pada kasus $k = 1$ sudah jelas benar karena

$$P(1): (p-q)|(p^1-q^1)$$

Asumsikan $k = n$ benar sehingga

$$P(n): (p-q)|(p^n-q^n); \forall n \in \mathbb{N}, p \neq q$$

Ini menyiratkan bahwa $p^n - q^n = (p-q)\ell$ dimana ℓ adalah polynomial arbitrer.

Pertimbangkan kasus $k = n+1$, diperoleh

$$P(n+1): (p-q)|(p^{n+1}-q^{n+1})$$

Akan ditunjukkan $(p^{n+1}-q^{n+1})$ habis dibagi $p-q$.

$$\begin{aligned}(p^{n+1}-q^{n+1}) &= p^{n+1} - p^n q + p^n q - q^{n+1} \\ &= p^n(p-q) + q(p^n - q^n) \\ &= p^n(p-q) + q(p-q)\ell \\ &= (p-q)(p^n - q\ell)\end{aligned}$$

Telah ditunjukkan bahwa $(p-q)|(p^{n+1}-q^{n+1})$ sehingga $P(n+1)$ benar. Dengan prinsip induksi matematika untuk $\forall k \in \mathbb{N}$ berlaku untuk $P(k)$ dengan demikian $S(k)$ juga berlaku.

Lemma 2.3.2

Misalkan $m = {p_1}^{\alpha_1}{p_2}^{\alpha_2}$ ... ${p_n}^{\alpha_n}$ menjadi dekomposisi dari modulus m pangkat prima dimana p_1 ... p_n adalah bilangan prima yang berbeda dan semua $\alpha_j \in \mathbb{Z}^+$. Panjang periode terkecil adalah d dari sembarang barisan kongruensi linier $\{x_i\}$ yang ditentukan oleh $x_0, a, c,$ dan m yang mana $d = lcm(d_j)$ (panjang periode terkecil dari barisan kongruensi linier $\{x_i\}$ yang ditentukan oleh x_0, a, c, dan ${p_j}^{\alpha_j}$, $1 \leq j \leq n$).

Bukti:

Dengan menggunakan induksi matematika pada n, kita hanya perlu membuktikan bahwa jika modulus $m = m_1 m_2$ dimana $gcd(m_1 m_2) = 1$, maka panjang periode barisan $\{x_i\}$ adalah d sedemikian hingga $d = lcm(d_1 d_2)$, dimana d_1 dan d_2 menjadi panjang periode barisan $\{y_i\}$ dan $\{z_i\}$ yang ditentukan oleh x_0, a, c, dan m_1 dan x_0, a, c, dan m_2.

Berdasarkan definisi dari ketiga barisan diatas, kita punya $y_i \equiv z_i (mod\ m_1)$ dan $z_i \equiv x_i (mod\ m_2)\ \forall i = 0$. Oleh karena itu, karena $gcd(m_1 m_2) = 1$, maka

$$x_i = x_j \leftrightarrow y_i = y_j\ dan\ z_i = z_j \qquad (1.4)$$

Catatan

Kita telah menggunakan hasil dasar: "jika $gcd(r,s) = 1$, maka $a \equiv b(mod\ rs)$ jika dan hanya jika $a \equiv b(mod\ r)$ dan $a \equiv b(mod\ s)$.

Misalkan $d' = lcm\ (d_1 d_2)$, perhatikan 2 fakta berikut;

(1) Karena barisan $\{y_i\}$ memiliki panjang periode d_1, maka $y_j = y_0$ jika dan hanya jika j adalah kelipatan d_1.

(2) Demikian pula, barisan $\{z_i\}$ memiliki panjang periode d_2, maka $z_j = z_0$ jika dan hanya jika j adalah kelipatan d_2.

Sekarang, karena barisan $\{x_i\}$ memiliki d sebagai panjang periode terkecil maka d adalah bilangan bulat positif terkecil sedemiikian sehingga $x_d = x_0$, oleh karena itu $y_d = y_0$ dan $z_d = z_0$ (menurut (1.4)). ini menyiratkan bahwa d adalah kelipatan dari d_1 dan d_2 (menurut (1) dan (2)), karena d adalah kelipatan dari $d' = lcm(d_1 d_2)$ sehingga $d = d'$. Selain itu, karena $d' = lcm(d_1 d_2)$ adalah kelipatan dari d_1 dan d_2 maka (1) dan (2) menyiratkan $y_{d'} = y_0$ dan $z_{d'} = z_0$. Jadi, menurut (1.4) $x_{d'} = x_0$ yang menyiratkan d' adalah kelipatan dari d, oleh karenanya $d' \geq d$. Jadi, $d = d'$ (Glen, 2002).

Lemma 2.3.3

Misalkan p prima dan $\alpha \in \mathbb{Z}^+$ sedemikian sehingga $p^\alpha > 2$. Jika

$$x \equiv 1 \ (mod\ p^\alpha), x \not\equiv 1 (mod\ p^{\alpha+1})$$

Maka

$$x^p \equiv 1 \ (mod\ p^{\alpha+1}), x^p \not\equiv 1 (mod\ p^{\alpha+2})$$

Bukti:

Misalkan $x \equiv 1 \ (mod\ p^\alpha), x \not\equiv 1 (mod\ p^{\alpha+1})$ maka $x = 1 + qp^\alpha, \exists q \in \mathbb{Z}$, dimana q bukan kelipatan p. Dengan menggunakan Teorema 2.1.3,

$x^p = (1 + qp^\alpha)^p$

$$= 1 + \binom{p}{1} qp^{\alpha} + \cdots + \binom{p}{p-1} q^{p-1} p^{(p-1)\alpha} + q^{p} p^{p\alpha}$$

$$= 1 + qp^{\alpha+1} \left(1 + \frac{1}{p}\binom{p}{2} qp^{\alpha} + \frac{1}{p}\binom{p}{3} q^{2} p^{2\alpha} + \cdots + \frac{1}{p}\binom{p}{p} q^{p-1} p^{(p-1)\alpha}\right)$$

Misalkan $\left(1 + \frac{1}{p}\binom{p}{2} qp^{\alpha} + \frac{1}{p}\binom{p}{3} q^{2} p^{2\alpha} + \cdots + \frac{1}{p}\binom{p}{p} q^{p-1} p^{(p-1)\alpha}\right)$ adalah ω dan $\omega \in \mathbb{Z}$, dapat dibuktikan bahwa $\binom{p}{r}$ habis dibagi oleh p untuk $1 < r < p$. Oleh karena itu, $\frac{1}{p}\binom{p}{r} q^{r-1} p^{(r-1)\alpha}$ habis dibagi oleh $p^{(r-1)\alpha}$ untuk $1 < r < p$, dan suku terakhir dari ω yakni $\frac{1}{p}\binom{p}{p} q^{p-1} p^{(p-1)\alpha} = q^{p-1} p^{(p-1)\alpha}$ habis dibagi oleh p karena $(p-1)\,\alpha > 1$ jika $p^{\alpha} > 2$. Jadi, p membagi setiap suku di ω kecuali suku pertama. Jadi $x^{p} = 1 + q' p^{\alpha+1}$, dimana $q' = q(\omega) \in \mathbb{Z}$ tidak habis dibagi oleh p. Ini mengimplikasikan bahwa $x^{p} \equiv 1\ (mod\ p^{\alpha+1}), x^{p} \not\equiv 1 (mod\ \ p^{\alpha+2})$ (Glen, 2002).

Lemma 2.3.4

Jika $a \equiv 3 (mod\ 4)$ maka $\frac{a^{2^{\alpha-1}}-1}{a-1} \equiv 0 (mod\ 2^{\alpha})$ ketika $\alpha > 1$.

Bukti:

Misalkan $a \equiv 3 (mod\ 4)$ maka $a = 3 + 4t$ untuk beberapa $t \in \mathbb{Z}$, artinya $a = 1 + (2 + 4t)$ jadi $a = 1 + 2(1 + 2t)$ yang mengimplikasikan bahwa $a \equiv 1 (mod\ 2)$. Sekarang, untuk $a^{2} = 9 + 24t + 16t^{2} = 1 + 8(1 + 3t + 2t^{2})$ yang berarti $a^{2} \equiv 1\ (mod\ 8)$, $a^{2} \not\equiv 1\ (mod\ 16)$ (mengikuti Lemma 2.3.3).

$$a^{4} \equiv 1\ (mod\ 16), a^{4} \not\equiv 1\ (mod\ 32)$$

$$a^{8} \equiv 1\ (mod\ 32), a^{8} \not\equiv 1\ (mod\ 64), dst$$

Secara umum, dengan menggunakan induksi matematika pada α, kita punya $a^{2^{\alpha-1}} \equiv 1\ (mod\ 2^{\alpha+1})$, oleh karenanya $a^{2^{\alpha-1}} - 1 \equiv 0\ (mod\ 2^{\alpha+1})$. Artinya, $a^{2^{\alpha-1}} - 1 = t2^{\alpha+1}$ untuk beberapa $t \in \mathbb{Z}^+$, oleh karena itu $\frac{\left(a^{2^{\alpha-1}}-1\right)}{2} = t2^{\alpha}$ sehingga $\frac{\left(a^{2^{\alpha-1}}-1\right)}{2} \equiv 0\ (mod\ 2^{\alpha})$. Jadi, ketika $\alpha = 1$ maka diperoleh $a \equiv 1(mod\ 2)$ (Glen, 2002).

c. Syarat Parameter *Linear Congruential Generator* (LCG)

Parameter a, b, dan m semuanya bilangan bulat dan hanya aritmatika bilangan bulat yang digunakan dalam perhitungan. Pemilihan nilai untuk a, c, dan m sangat penting dalam mengembangkan generator bilangan acak yang baik karena kekuatan dari algoritma kongruensial linier adalah bahwa jika pengali (a), *increament* (c) dan modulo (m) dipilih dengan benar. Pemilihan nilai parameter secara drastis mempengaruhi sifat statistik dan panjang siklus (Kneusel, 2018).

Jika $a = 1$, maka LCG didefinisikan dengan $X_n = (X_{n-i} + c)\ (mod\ m)$, generator tersebut biasanya disebut sebagai *additive congruence method* (ACM). Jika $c = 0$, maka LCG didefinisikan dengan $X_n = (a * X_{n-i})\ (mod\ m)$, generator tersebut biasanya disebut sebagai *multiplicative congruential generator* (MCG), atau Lehmer RNG.

Adapun untuk $c \neq 0$, syarat pemilihan parameter agar memiliki panjang periode sebesar nilai m adalah sebagai berikut:

1) Parameter c dan m relatif prima,
2) $a \equiv 1\ (mod\ p)$ jika p adalah faktor prima dari m,
3) $a \equiv 1\ (mod\ 4)$ jika 4 habis membagi m.

Adapun untuk $c = 0$, syarat pemilihan parameter agar memiliki panjang periode maksimal adalah sebagai berikut:

1) Parameter X_0 dan m relatif prima,
2) a adalah akar primitif untuk p^α, jika p^α adalah faktor dari m, dengan p ganjil dan α sebesar mungkin, atau dengan $p = 2$ dan $\alpha = 1\ atau\ 2$ (Tezuka, 1995).

2.3.2 *One Time Pad* (OTP)

a. Definisi *One Time Pad* (OTP)

Sebuah *cipher* di mana kuncinya memiliki panjang sebanyak *plaintext* itu sendiri, dan kuncinya benar-benar dibuat secara acak dan tidak pernah digunakan lebih dari sekali. Ini adalah deskripsi OTP yang dikembangkan oleh Vernam dan Mauborgne pada tahun 1918. Karena kunci sepanjang *plaintext*, dan kunci yang dipilih benar-benar acak dan hanya digunakan sekali, maka *ciphertext* juga benar-benar acak. Dengan demikian, OTP tidak dapat dipecahkan, artinya tidak mungkin untuk memecahkan dengan metode kriptanalitik apa pun. Sandi OTP bukanlah sandi blok, melainkan merupakan contoh *stream cipher* (Mollin, 2007:101).

b. Algoritma *One Time Pad* (OTP)

Secara matematis proses enkripsi algoritma OTP dirumuskan sebagai berikut:

$$C_i = P_i + K_i (mod\ n)$$

Sedangkan untuk proses dekripsinya dirumuskan sebagai berikut:

$$P_i = C_i - K_i (mod\ n)$$

Dengan kunci

$$k_1 k_2 \dots k_j \in K_i$$

Dimana $i = 1,2, \dots j$

Ket: C_i = Chiperteks (*Chipertext*)

P_i = Plainteks (*Plaintext*)

K_i = Kunci (*key*)

n = Jumlah karakter yang digunakan (Mollin, 2007:113)

Contoh :

Pesan asli (*plaintext*) yang akan dikirimkan adalah "ONEPAD" dengan kunci "TBFRGM" dengan asumsi $A = 0, B = 1, \dots, Z = 25$ yang berarti nilai modulo nya adalah 26. Maka proses enkripsi pesan sebagai berikut:

$$O + T\ (mod\ 26) = 14 + 19(mod 26) = 7 = H$$

$$N + B\ (mod\ 26) = 13 + 1(mod 26) = 14 = O$$

$$E + F\ (mod\ 26) = 4 + 5(mod 26) = 9 = J$$

$$P + R\ (mod\ 26) = 15 + 17(mod 26) = 6 = G$$

$$A + G\ (mod\ 26) = 0 + 6(mod 26) = 6 = G$$

$$D + M\ (mod\ 26) = 3 + 12(mod 26) = 15 = P$$

Sehingga, didapatkan *ciphertext* "HOJGGP". Sedangkan untuk proses dekripsi pesan sebagai berikut:

$$H - T\ (mod\ 26) = 33 - 19(mod 26) = 14 = O$$

$$O - B\ (mod\ 26) = 14 - 1(mod 26) = 13 = N$$

$$J - F\ (mod\ 26) = 9 - 5(mod 26) = 4 = E$$

$$G - R\ (mod\ 26) = 32 - 17(mod 26) = 15 = P$$

$$G - G\ (mod\ 26) = 6 - 6(mod 26) = 0 = A$$

$$P - M\ (mod\ 26) = 15 - 12(mod26) = 3 = D$$

Sehingga, didapatkan *plaintext* “ONEPAD”.

Mendekripsikan suatu *chipertext* dengan menggunakan kunci yang berbeda akan menghasilkan *plaintext* yang berbeda pula, misalnya dari chiperteks diatas kita ambil kunci yang berbeda yakni “POYUSC”, maka proses dekripsinya sebagai berikut:

$$H - P\ (mod\ 26) = 33 - 15(mod26) = 18 = S$$

$$O - O\ (mod\ 26) = 14 - 14(mod26) = 0 = A$$

$$J - Y\ (mod\ 26) = 35 - 24(mod26) = 11 = L$$

$$G - U\ (mod\ 26) = 32 - 20(mod26) = 12 = M$$

$$G - S\ (mod\ 26) = 32 - 18(mod26) = 14 = O$$

$$P - C\ (mod\ 26) = 15 - 2(mod26) = 13 = N$$

Didapatkan *plaintext* “SALMON”, *plaintext* tersebut memiliki makna, sehingga hal ini dapat memicu kebingungan kriptanalis dalam menentukan *plaintext* mana yang benar. Hal inilah yang menjadi salah satu kelebihan dari algoritma OTP.

BAB III

METODE PENELITIAN

3.1 Jenis Penelitian

Dalam menyusun penelitian ini, peneliti menggunakan penelitian literatur atau kepustakaan (*library research*). Penelitian kepustakaan adalah penelitian yang dilakukan tanpa terjun ke lapangan dan hanya memanfaatkan literatur-literatur seperti buku, artikel, atau jurnal penelitian terdahulu sebagai sumber informasi dan data.

3.2 Pra Penelitian

Peneliti mempelajari, mengumpulkan serta mengolah bahan penelitian dari beberapa buku yang berkaitan dengan penelitian ini. Data yang digunakan dalam penelitian ini ada 2, yang pertama nilai parameter yang akan digunakan untuk membangkitkan barisan bilangan acak dan yang kedua berupa pesan biasa (*plaintext*) yang akan dirubah menjadi pesan tersandi (*ciphertext*) dengan menggunakan kunci.

Adapun buku yang menjadi sumber informasi dalam penelitian ini adalah buku *An Overview of Cryptography* karangan Gary C. Kessler, buku *Terjemah Tafsir Ibnu Katsir* karangan Al-Imam Abul Fida Isma'il Ibnu Katsir ad-Damasyqi, buku *An Introduction to Cryptography* karangan Richard Mollin, buku Kriptografi karangan M. Wasim Munir, buku *Modern Cryptography Applied Mathematics for Encryption and Information Security* karangan William Easttom, buku *Random Numbers and Computers* karangan Ronald Kneusel, buku *Elementary Number Theory* karangan Kenneth H. Rossen, buku *On the Period Length of Pseudorandom*

Number Sequences karangan Amy Glen dan buku *Cryptography and Network Security Principles and Practices* karangan William Stallings.

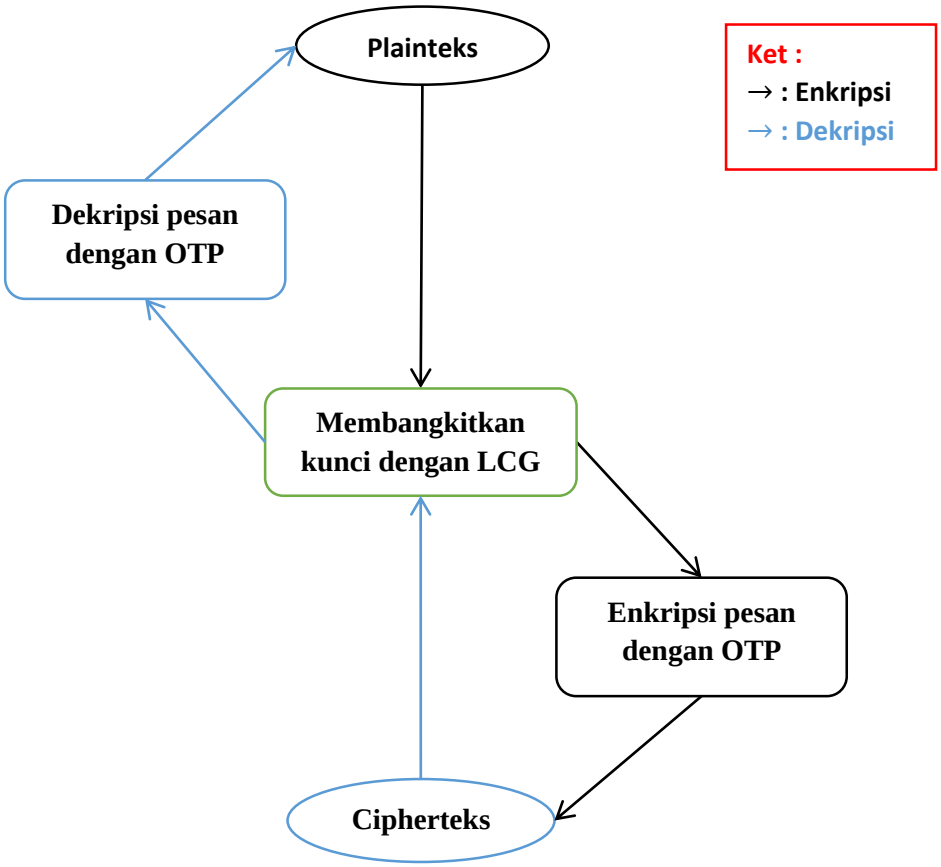
3.3 Tahapan Penelitian

Adapun tahapan-tahapan yang digunakan untuk mengolah data dalam penelitian ini sebagai berikut :

1) Proses pembangkitan kunci menggunakan *Linear Congruential Generator* (LCG):
   a. Menentukan *plaintext* atau pesan asli yang akan disandikan.
   b. Menentukan parameter yang akan digunakan pada LCG. Nilai parameter yang harus ditentukan adalah nilai awal (X_0), nilai faktor pengali (a), nilai *increament* (c), dan nilai modulo (m).
   c. Melakukan proses perhitungan nilai LCG dimulai dari X_1 sampai sepanjang *plaintext*nya.
   d. Mendapatkan nilai LCG yang akan digunakan sebagai pembangkit kunci.
2) Proses enkripsi pesan menggunakan algoritma *One Time Pad* (OTP) dengan LCG sebagai kunci pembangkit:
   a. Menentukan *plaintext* atau pesan asli yang akan disandikan.
   b. Mengkonversikan *plaintext* ke nilai karakter menurut tabel ASCII.
   c. Melakukan perhitungan menggunakan rumus enkripsi algoritma OTP dengan modulo 256 sesuai dengan jumlah karakter yang ada pada tabel ASCII dan menggunakan kunci yang telah dibangkitkan dari perhitungan LCG.
   d. Mengkonversi nilai hasil enkripsi ke nilai karakter menurut tabel ASCII.
   e. Mendapatkan pesan berupa pesan tersandi (*ciphertext*).

3) Proses dekripsi pesan menggunakan algoritma *One Time Pad* (OTP) dengan LCG sebagai kunci pembangkit
    a. Menerima *ciphertext* atau pesan tersandi dan nilai parameter yang digunakan pengirim untuk melakukan perhitungan LCG.
    b. Melakukan perhitungan nilai LCG dengan menggunakan parameter yang sama seperti yang digunakan oleh pengirim.
    c. Mengkonversi pesan yang tersandi (*ciphertext*) ke nilai karakter menurut tabel ASCII.
    d. Melakukan perhitungan menggunakan rumus dekripsi algoritma OTP dengan modulo 256 sesuai dengan jumlah karakter yang ada pada tabel ASCII dan menggunakan kunci yang telah dibangkitkan dari perhitungan LCG.
    e. Mengkonversi nilai hasil dekripsi ke nilai karakter menurut tabel ASCII.
    f. Mendapatkan pesan berupa pesan asli (*plaintext*).

Secara singkat, tahapan-tahapan diatas dapat dilihat pada *flowchart* dibawah ini:



Gambar 3.1.1 Flowchart Tahapan Penelitian

BAB IV

PEMBAHASAN

Pembahasan merupakan bab yang menjelaskan hasil penelitian atau jawaban dari permasalah yang telah dipaparkan pada rumusan masalah di bab 1. Berdasarkan rumusan masalah tersebut maka terdapat 3 subbab pada bab ini. Permasalahan pertama dijawab pada subbab 4.1 yakni proses pembangkitan kunci dengan *Linear Congruential Generator* (LCG), permasalahan kedua dijawab pada subbab 4.2 yakni proses enkripsi menggunakan algoritma *One Time Pad* (OTP) dengan *Linear Congruential Generator* (LCG) sebagai pembangkit kunci, dan permasalahan ketiga dijawab pada subbab 4.3 yakni proses dekripsi menggunakan algoritma *One Time Pad* (OTP) dengan *Linear Congruential Generator* (LCG) sebagai pembangkit kunci.

4.1 Proses Pembangkitan Kunci dengan *Linear Congruential Generator* (LCG)

Proses membangkitkan kunci dengan LCG adalah proses membentuk barisan bilangan acak menggunakan metode LCG yang mana nanti barisan tersebut akan dijadikan kunci pada proses enkripsi dan dekripsi pesan. Adapun proses pembangkitan kunci sebagai berikut:

1. Menentukan *plaintext* atau pesan asli yang akan disandikan.
2. Menentukan nilai parameter yang akan digunakan, nilai a haruslah $0 < a < m$, nilai c haruslah $0 \leq b < m$, nilai X_0 haruslah $0 \leq X_0 < m$, dan nilai m haruslah $m > 0,\ m > a,\ m > c,\ m > X_0$.

3. Pengambilan nilai parameter pada $c \neq 0$ harus mengikuti Teorema 4.1.

Teorema 4.1

Linear congruential generators yang dibangun oleh barisan $x_i \equiv a^i x_0 + \frac{(a^i-1)c}{a-1} (mod\ m), \forall i \geq 0$, memiliki panjang periode sebesar nilai m jika memenuhi kondisi sebagai berikut:

1) Parameter c dan m relatif prima,
2) $a \equiv 1\ (mod\ p)$ jika p adalah faktor prima dari m,
3) $a \equiv 1\ (mod\ 4)$ jika 4 habis membagi m (Knuth, 1981).

Bukti:

Dengan menggunakan Lemma 2.2, kita hanya perlu mempertimbangkan kasus modulus $m = p^{\alpha}$, dimana p prima dan $\alpha \in \mathbb{Z}^+$, karena $p_1{}^{\alpha_1} p_2{}^{\alpha_2} \dots p_n{}^{\alpha_n} = d = lcm(d_1, d_2, \dots, d_n) \leq d_1, d_2, \dots, d_n \leq p_1{}^{\alpha_1} p_2{}^{\alpha_2} \dots p_n{}^{\alpha_n}$ berlaku jika dan hanya jika $d_j = p_j{}^{\alpha_j}$ untuk $1 \leq j \leq n$.

Untuk $a = 1$, misalkan $i = \frac{(a^i-1)}{a-1}$ maka $x_i \equiv x_0 + ic\ (mod\ m), \forall i \geq 0$ dan $x_i = x_0$ ketika $ic\ (mod\ m) = 0$ atau dapat dituliskan $ic \equiv 0\ (mod\ m)$ yaitu jika dan hanya jika $i \equiv 0\ \left(mod \frac{m}{k}\right)$ dimana $k = gcd(c, m)$ (menurut Teorema 2.1.1). Jadi, $x_i = x_0$ jika dan hanya jika i adalah kelipatan dari $\frac{m}{k}$. Hal ini menyiratkan bahwa $\frac{m}{k}$ adalah panjang periode barisan $\{x_i\}$ dan memiliki panjang periode penuh sebesar m jika dan hanya jika $k = gcd(c, m) = 1$. Selanjutnya, karena $a = 1$ kita juga punya $1 \equiv 1\ (mod\ p)$ jika p adalah faktor prima dari m dan $1 \equiv 1\ (mod\ 4)$ jika 4 habis membagi m. Jadi, ketiga kondisi tersebut berlaku untuk $a = 1$.

Untuk $a > 1$, perhatikan bahwa panjang periodenya adalah m jika dan hanya jika setiap bilangan bulat $0,1,2,\dots,(m-1)$ yang mungkin muncul dalam periode, tidak ada nilai yang muncul lebih dari satu kali. Oleh karena itu, panjang periode sama dengan m jika dan hanya jika $x_0 = 0$. Jadi, dapat diasumsikan bahwa panjang periode barisan $\{x_i\}$ adalah $m = p^\alpha$ dengan nilai awal $x_0 = 0$. Dengan mengunakan Lemma 2.3.1 maka barisan $\{x_i\}$ diberikan sebagai berikut

$$x_i \equiv \frac{(a^i - 1)c}{a-1} \,(mod\ m), \forall i \geq 1 \qquad (2.1)$$

Ambil $x_i = 1$ maka diperoleh $\frac{(a^i-1)c}{a-1} \equiv 1\ (mod\ m)$. Selain itu, kongruensi ini berlaku jika dan hanya jika $gcd(c,m) = 1$ karena c memiliki invers modulo m, yaitu $\frac{(a^i-1)}{a-1} = 1 + a + a^2 + \cdots + a^{i-1} \in \mathbb{Z}$ (menurut deret geometri). Jadi, kondisi pertama telah dibuktikan (Glen, 2002).

Untuk membuktikan kondisi kedua dan ketiga, perhatikan fakta berikut: Panjang periode barisan adalah $m = p^\alpha$ jika dan hanya jika n adalah bilangan bulat positif terkecil sehingga $x_n = x_0 = 0$ yakni $n = m$. Ini menyiratkan bahwa $\frac{(a^n-1)c}{a-1} \equiv 0\ (mod\ m)$. Karena $gcd(c,m) = 1$ maka $\frac{(a^n-1)}{a-1} \equiv 0\ (mod\ m)$ (menurut Teorema 2.1.1). Untuk membuktikan bahwa $n = p^\alpha$, fakta diatas direduksi menjadi pembuktian Lemma berikut:

Lemma 4.1 Asumsikan bahwa $1 < a < p^\alpha$, dimana p prima. Jika n adalah bilangan bulat positif terkecil maka

$$n = p^\alpha \text{ jika dan hanya jika } \begin{cases} a \equiv 1\ (mod\ p), & p > 2 \\ a \equiv 1\ (mod\ 4), & p = 2 \end{cases}$$

Bukti:

($\Longrightarrow$)

(1) Asumsikan bahwa $n = p^{\alpha}$. Jika $a \not\equiv 1 \ (mod\ p)$ maka $\frac{(a^n-1)}{a-1} \equiv 0 \ (mod\ p^{\alpha})$ jika dan hanya jika $a^n - 1 \equiv 0 \ (mod\ p^{\alpha})$. Karena $n = p^{\alpha}$ maka $a^{p^{\alpha}} - 1 \equiv 0 \ (mod\ p^{\alpha})$, artinya $a^{p^{\alpha}} - 1 = tp^{\alpha}$ untuk beberapa $t \in \mathbb{Z}^+$ atau dapat dinyatakan $a^{p^{\alpha}} - 1 = (tp^{\alpha-1})p$ dimana $tp^{\alpha-1} \in \mathbb{Z}^+$ dan p habis membagi $(a^{p^{\alpha}} - 1)$ sehingga $a^{p^{\alpha}} \equiv 1 \ (mod\ p)$. Tetapi, menurut Teorema 2.1.4 menyatakan $a^{p^{\alpha}} \equiv a \ (mod\ p)$ jadi $a^n \equiv a \ (mod\ p)$. Hal ini berkontradiksi, oleh karena itu $a \equiv 1 \ (mod\ p)$ (Knuth, 1981).

(2) Misalkan $p = 2$ sehingga $n = m = 2^{\alpha}, \alpha \geq 2$. Jika $a \equiv 1 \ (mod\ 2)$ tetapi $a \not\equiv 1 \ (mod\ 4),\ a \equiv 3 \ (mod\ 4)$ maka $\frac{a^{2^{\alpha-1}}-1}{a-1} \equiv 0 \ (mod\ 2^{\alpha})$ menurut Lemma 2.4. Hal ini kontradiksi karena $\frac{a^{2^{\alpha}}-1}{a-1} \equiv 0 \ (mod\ 2^{\alpha})$ ketika $n = m = 2^{\alpha}$, dimana $a \equiv 1 \ (mod\ 4)$. Alternatifnya, gunakan $x_i \equiv ax_{i-1} + c \ (mod\ m)$, maka diperoleh

$$x_i \equiv a(ax_{i-2} + c) + c \ (mod\ m)$$
$$\equiv a^2 x_{i-2} + (a+1)c \ (mod\ m)$$

Jadi, karena $x_0 = 0$ maka $x_2, x_4, x_6, \ldots$ adalah kelipatan dari $(a+1)c \equiv 4c \ (mod\ m)$. Akibatnya, barisan $\{x_{2i}\}$ memiliki panjang periode paling banyak $\frac{m}{4}$ sehingga pada $\{x_i\}$ memiliki panjang periode paling banyak $\frac{m}{2}$. Hal ini kontradiksi, oleh karena itu $a \equiv 1 \ (mod\ 4)$ (Glen, 2002).

($\Leftarrow$)

Asumsikan $n = p^{\alpha}$ dan amati bahwa kondisi kedua dan ketiga menyiratkan bahwa $a = 1 + qp^{\beta}$, dimana $p^{\beta} > 2$ dan q bukan kelipatan dari p. Dengan menggunakan Lemma 2.3.3, $a^{p} \equiv 1\ (mod\ p^{\beta+1})$ tetapi $a^{p} \not\equiv 1\ (mod\ p^{\beta+2})$. Dengan menggunakan induksi matematika pada γ, maka $a^{p^{\gamma}} \equiv 1\ (mod\ p^{\beta+\gamma})$ tetapi $a^{p^{\gamma}} \not\equiv 1\ \left(mod\ p^{\beta+\gamma+1}\right), \forall \gamma \geq 0$. Oleh karena itu,

$$\frac{a^{p^{\gamma}} - 1}{a - 1} \equiv 0\ (mod\ p^{\gamma});\ \frac{a^{p^{\gamma}} - 1}{a - 1} \not\equiv 0\ (mod\ p^{\gamma+1}) \qquad (2.2)$$

Persamaan (2.2) mengimplikasikan $\frac{a^{p^{\alpha}}-1}{a-1} \equiv 0\ (mod\ p^{\alpha})$. Dengan menggunakan persamaan (2.1), barisan konruensi linier $\{x_i\}$ yang ditentukan oleh $0, a, c,$ dan p^{α} sedemikian sehingga $x_i \equiv \frac{(a^{i}-1)c}{a-1}(mod\ p^{\alpha}),\ \forall i \geq 0$. Misalkan barisan ini memiliki panjang periode n, maka $x_i = x_0 = 0$ jika dan hanya jika i adalah kelipatan dari n, yaitu $\frac{a^{i}-1}{a-1} \equiv 0\ (mod\ p^{\alpha})$ karena $(c, m) = 1$ (menurut Teorema 2.1.1). Jadi, p^{α} adalah kelipatan n, yang hanya dapat terjadi jika $n = p^{\gamma}$, untuk suatu bilangan bulat $\gamma \geq 0$ dan persamaan (2.2) mengimplikasikan $n = p^{\alpha}$. Jadi, kondisi kedua dan ketiga telah terbukti (Knuth, 1981).

4. Melakukan perhitungan LCG dengan rumus:

$$X_n = (a * X_{n-i} + c)\ (mod\ m)$$

5. Panjang periode harus lebih besar atau sama dengan panjang plainteks.

Adapun implementasinya adalah sebagai berikut:

Si A akan mengirimkan sebuah pesan rahasia kepada si B, si A sangat berhati-hati dalam memilih nilai parameter yang akan digunakan dalam proses pembangkitan bilangan acak, sebab barisan tersebut nantinya akan digunakan sebagai kunci dalam penyandian pesan dan kunci tersebut tidak boleh rekursif. Pada kasus kali ini, agar kunci tak rekursif maka panjang periode barisan bilangan acak tersebut tidak boleh kurang dari panjang *plaintext*nya maka si A melakukan beberapa kali percobaan dalam membangkitkan bilangan acak dengan menggunakan parameter yang berbeda-beda.

Percobaan pertama :

1) *Plaintext* atau pesan asli yang akan disandikan adalah "Satgas COVID-19" dengan panjang *plaintext* sebesar 15.
2) Nilai parameter yang digunakan oleh si A adalah $c \neq 0$ dan $a \neq 1$, dipilih $c = 18$, $m = 38$, $a = 7$, dan $(X_0) = 33$.
3) Nilai $c = 18$ dan $m = 38$ **tidak** relatif prima, faktor prima dari $m = 38$ adalah 2 yang mana 2 dapat membagi $(a - 1) = 6$, dan $m = 38$, $(a - 1) = 6$ keduanya tidak habis dibagi oleh 4.
4) Melakukan proses perhitungan nilai LCG sebagai berikut:

$$X_1 = (7 \cdot X_0 + 18)\ (mod\ 38) = (7 \cdot 33 + 18)(mod\ 38) = 21$$

$$X_2 = (7 \cdot X_1 + 18)\ (mod\ 38) = (7 \cdot 21 + 18)(mod\ 38) = 13$$

$$X_3 = (7 \cdot X_2 + 18)\ (mod\ 38) = (7 \cdot 13 + 18)(mod\ 38) = 33$$

$$X_4 = (7 \cdot X_3 + 18)\ (mod\ 38) = (7 \cdot 33 + 18)(mod\ 38) = 21$$

$$X_5 = (7 \cdot X_4 + 18)\ (mod\ 38) = (7 \cdot 21 + 18)(mod\ 38) = 13$$

5) Setelah dilakukan perhitungan dapat dilihat bahwa dengan pengambilan nilai parameter $X_0 = 33, c = 18, m = 38,$ dan $a = 7,$ yang mana pengambilan nilai parameter tersebut tidak memenuhi syarat pertama yakni m dan c tidak relatif prima maka didapatkan panjang periodenya 3, kurang dari m juga kurang 15.

Percobaan kedua :

1) *Plaintext* atau pesan asli yang akan disandikan adalah "Satgas COVID-19" dengan panjang *plaintext* sebesar 15.
2) Nilai parameter yang digunakan oleh si A adalah $c \neq 0$ dan $a \neq 1$, dipilih $c = 17, m = 60, a = 45,$ dan $(X_0) = 33.$
3) Nilai $c = 17$ dan $m = 60$ relatif prima, faktor prima dari $m = 60$ adalah 2,3 dan 5 namun hanya 2 yang dapat membagi $(a - 1) = 44$ sedangkan 3 dan 5 **tidak** dapat membagi $(a - 1) = 44,$ dan $m = 120,$ $(a - 1) = 44$ keduanya habis dibagi oleh 4.
4) Melakukan proses perhitungan nilai LCG sebagai berikut:

$$X_1 = (45 \cdot X_0 + 17)\ (mod\ 60) = (45 \cdot 33 + 17)(mod\ 60) = 2$$

$$X_2 = (45 \cdot X_1 + 17)\ (mod\ 60) = (45 \cdot 2 + 17)(mod\ 60) = 47$$

$$X_3 = (45 \cdot X_2 + 17)\ (mod\ 60) = (45 \cdot 47 + 17)(mod\ 60) = 32$$

$$X_4 = (45 \cdot X_3 + 17)\ (mod\ 60) = (45 \cdot 32 + 17)(mod\ 60) = 17$$

$$X_5 = (45 \cdot X_4 + 17)\ (mod\ 60) = (45 \cdot 17 + 17)(mod\ 60) = 2$$

$$X_6 = (45 \cdot X_5 + 17)\ (mod\ 60) = (45 \cdot 2 + 17)(mod\ 60) = 47$$

5) Setelah dilakukan perhitungan dapat dilihat bahwa dengan pengambilan nilai parameter $X_0 = 33,\ c = 18,\ m = 60,$ dan $a = 45,$ yang mana pengambilan nilai parameter tersebut tidak memenuhi syarat kedua yakni ada faktor prima

dari m yang tidak dapat membagi $(a-1)$ maka didapatkan panjang periodenya 4, kurang dari m juga kurang dari 15.

Percobaan ketiga :

1) *Plaintext* atau pesan asli yang akan disandikan adalah "Satgas COVID-19" dengan panjang *plaintext* sebesar 15.
2) Nilai parameter yang digunakan oleh si A adalah $c \neq 0$ dan $a \neq 1$, dipilih $c = 17$, $m = 40$, $a = 31$, dan $(X_0) = 33$.
3) Nilai $c = 17$ dan $m = 40$ relatif prima, faktor prima dari $m = 40$ adalah 2 dan 5 sedangkan 2 dan 5 keduanya dapat membagi $(a-1) = 30$, dan $m = 40$ habis dibagi oleh 4 namun $(a-1) = 30$ **tidak** habis dibagi oleh 4.
4) Melakukan proses perhitungan nilai LCG sebagai berikut:

$$X_1 = (31 \cdot X_0 + 17)\ (mod\ 40) = (31 \cdot 33 + 17)(mod\ 40) = 0$$

$$X_2 = (31 \cdot X_1 + 17)\ (mod\ 40) = (31 \cdot 0 + 17)(mod\ 40) = 17$$

$$X_3 = (31 \cdot X_2 + 17)\ (mod\ 40) = (31 \cdot 17 + 17)(mod\ 40) = 24$$

$$X_4 = (31 \cdot X_3 + 17)\ (mod\ 40) = (31 \cdot 24 + 17)(mod\ 40) = 1$$

$$X_5 = (31 \cdot X_4 + 17)\ (mod\ 40) = (31 \cdot 1 + 17)(mod\ 40) = 8$$

$$X_6 = (31 \cdot X_5 + 17)\ (mod\ 40) = (31 \cdot 8 + 17)(mod\ 40) = 25$$

$$X_7 = (31 \cdot X_6 + 17)\ (mod\ 40) = (31 \cdot 25 + 17)(mod\ 40) = 32$$

$$X_8 = (31 \cdot X_7 + 17)\ (mod\ 40) = (31 \cdot 32 + 17)(mod\ 40) = 9$$

$$X_9 = (31 \cdot X_8 + 17)\ (mod\ 40) = (31 \cdot 9 + 17)(mod\ 40) = 16$$

$$X_{10} = (31 \cdot X_9 + 17)\ (mod\ 40) = (31 \cdot 16 + 17)(mod\ 40) = 33$$

$$X_{11} = (31 \cdot X_{10} + 17)\ (mod\ 40) = (31 \cdot 33 + 17)(mod\ 40) = 0$$

$$X_{12} = (31 \cdot X_{11} + 17)\ (mod\ 40) = (31 \cdot 0 + 17)(mod\ 40) = 17$$

5) Setelah dilakukan perhitungan dapat dilihat bahwa dengan pengambilan nilai parameter $X_0 = 33$, $c = 17$, $m = 40$, dan $a = 31$, yang mana pengambilan nilai parameter tersebut tidak memenuhi syarat ketiga yakni $(a - 1)$ tidak habis dibagi oleh 4 ketika m habis dibagi oleh 4 maka didapatkan panjang periodenya 10, kurang dari m juga kurang dari 15.

Percobaan keempat :

1) *Plaintext* atau pesan asli yang akan disandikan adalah "Satgas COVID-19" dengan panjang *plaintext* sebesar 15.
2) Nilai parameter yang digunakan oleh si A adalah $c \neq 0$ dan $a \neq 1$, dipilih $c = 17$, $m = 50$, $a = 11$, dan $(X_0) = 33$.
3) Nilai $c = 17$ dan $m = 50$ relatif prima, faktor prima dari $m = 50$ adalah 2 dan 5 sedangkan 2 dan 5 keduanya dapat membagi $(a - 1) = 10$, dan $m = 50$, $(a - 1) = 10$ keduanya tidak habis dibagi oleh 4.
4) Melakukan proses perhitungan nilai LCG sebagai berikut:

$$X_1 = (11 \cdot X_0 + 17)\ (mod\ 50) = (11 \cdot 33 + 17)(mod\ 50) = 30$$

$$X_2 = (11 \cdot X_1 + 17)\ (mod\ 50) = (11 \cdot 30 + 17)(mod\ 50) = 47$$

$$X_3 = (11 \cdot X_2 + 17)\ (mod\ 50) = (11 \cdot 47 + 17)(mod\ 50) = 34$$

$$X_4 = (11 \cdot X_3 + 17)\ (mod\ 50) = (11 \cdot 34 + 17)(mod\ 50) = 41$$

$$X_5 = (11 \cdot X_4 + 17)\ (mod\ 50) = (11 \cdot 41 + 17)(mod\ 50) = 18$$

$$X_6 = (11 \cdot X_5 + 17)\ (mod\ 50) = (11 \cdot 18 + 17)(mod\ 50) = 15$$

$$X_7 = (11 \cdot X_6 + 17)\ (mod\ 50) = (11 \cdot 15 + 17)(mod\ 50) = 32$$

$$X_8 = (11 \cdot X_7 + 17)\ (mod\ 50) = (11 \cdot 32 + 17)(mod\ 50) = 19$$

$$X_9 = (11 \cdot X_8 + 17)\ (mod\ 50) = (11 \cdot 19 + 17)(mod\ 50) = 26$$

$$X_{10} = (11 \cdot X_9 + 17)\ (mod\ 50) = (11 \cdot 26 + 17)(mod\ 50) = 3$$

$$X_{11} = (11 \cdot X_{10} + 17)\ (mod\ 50) = (11 \cdot 3 + 17)(mod\ 50) = 0$$

$$X_{12} = (11 \cdot X_{11} + 17)\ (mod\ 50) = (11 \cdot 0 + 17)(mod\ 50) = 17$$

$$X_{13} = (11 \cdot X_{12} + 17)\ (mod\ 50) = (11 \cdot 17 + 17)(mod\ 50) = 4$$

$$X_{14} = (11 \cdot X_{13} + 17)\ (mod\ 50) = (11 \cdot 4 + 17)(mod\ 50) = 11$$

$$X_{15} = (11 \cdot X_{14} + 17)\ (mod\ 50) = (11 \cdot 11 + 17)(mod\ 50) = 38$$

$$X_{16} = (11 \cdot X_{15} + 17)\ (mod\ 50) = (11 \cdot 38 + 17)(mod\ 50) = 35$$

$$X_{17} = (11 \cdot X_{16} + 17)\ (mod\ 50) = (11 \cdot 35 + 17)(mod\ 50) = 2$$

$$X_{18} = (11 \cdot X_{17} + 17)\ (mod\ 50) = (11 \cdot 2 + 17)(mod\ 50) = 39$$

$$X_{19} = (11 \cdot X_{18} + 17)\ (mod\ 50) = (11 \cdot 39 + 17)(mod\ 50) = 46$$

$$X_{20} = (11 \cdot X_{19} + 17)\ (mod\ 50) = (11 \cdot 46 + 17)(mod\ 50) = 23$$

$$X_{21} = (11 \cdot X_{20} + 17)\ (mod\ 50) = (11 \cdot 23 + 17)(mod\ 50) = 20$$

$$X_{22} = (11 \cdot X_{21} + 17)\ (mod\ 50) = (11 \cdot 20 + 17)(mod\ 50) = 37$$

$$X_{23} = (11 \cdot X_{22} + 17)\ (mod\ 50) = (11 \cdot 37 + 17)(mod\ 50) = 24$$

$$X_{24} = (11 \cdot X_{23} + 17)\ (mod\ 50) = (11 \cdot 24 + 17)(mod\ 50) = 31$$

$$X_{25} = (11 \cdot X_{24} + 17)\ (mod\ 50) = (11 \cdot 31 + 17)(mod\ 50) = 8$$

$$X_{26} = (11 \cdot X_{25} + 17)\ (mod\ 50) = (11 \cdot 8 + 17)(mod\ 50) = 5$$

$$X_{27} = (11 \cdot X_{26} + 17)\ (mod\ 50) = (11 \cdot 5 + 17)(mod\ 50) = 22$$

$$X_{28} = (11 \cdot X_{27} + 17)\ (mod\ 50) = (11 \cdot 22 + 17)(mod\ 50) = 9$$

$$X_{29} = (11 \cdot X_{28} + 17)\ (mod\ 50) = (11 \cdot 9 + 17)(mod\ 50) = 16$$

$$X_{30} = (11 \cdot X_{29} + 17)\ (mod\ 50) = (11 \cdot 16 + 17)(mod\ 50) = 43$$

$$X_{31} = (11 \cdot X_{30} + 17)\ (mod\ 50) = (11 \cdot 43 + 17)(mod\ 50) = 40$$

$$X_{32} = (11 \cdot X_{31} + 17)\ (mod\ 50) = (11 \cdot 40 + 17)(mod\ 50) = 7$$

$$X_{33} = (11 \cdot X_{32} + 17)\ (mod\ 50) = (11 \cdot 7 + 17)(mod\ 50) = 44$$

$$X_{34} = (11 \cdot X_{33} + 17)\ (mod\ 50) = (11 \cdot 44 + 17)(mod\ 50) = 1$$

$$X_{35} = (11 \cdot X_{34} + 17)\ (mod\ 50) = (11 \cdot 1 + 17)(mod\ 50) = 28$$

$$X_{36} = (11 \cdot X_{35} + 17)\ (mod\ 50) = (11 \cdot 28 + 17)(mod\ 50) = 25$$

$$X_{37} = (11 \cdot X_{36} + 17)\ (mod\ 50) = (11 \cdot 25 + 17)(mod\ 50) = 42$$

$$X_{38} = (11 \cdot X_{37} + 17)\ (mod\ 50) = (11 \cdot 42 + 17)(mod\ 50) = 29$$

$$X_{39} = (11 \cdot X_{38} + 17)\ (mod\ 50) = (11 \cdot 29 + 17)(mod\ 50) = 36$$

$$X_{40} = (11 \cdot X_{39} + 17)\ (mod\ 50) = (11 \cdot 36 + 17)(mod\ 50) = 13$$

$$X_{41} = (11 \cdot X_{40} + 17)\ (mod\ 50) = (11 \cdot 13 + 17)(mod\ 50) = 10$$

$$X_{42} = (11 \cdot X_{41} + 17)\ (mod\ 50) = (11 \cdot 10 + 17)(mod\ 50) = 27$$

$$X_{43} = (11 \cdot X_{42} + 17)\ (mod\ 50) = (11 \cdot 27 + 17)(mod\ 50) = 14$$

$$X_{44} = (11 \cdot X_{43} + 17)\ (mod\ 50) = (11 \cdot 14 + 17)(mod\ 50) = 21$$

$$X_{45} = (11 \cdot X_{44} + 17)\ (mod\ 50) = (11 \cdot 21 + 17)(mod\ 50) = 48$$

$$X_{46} = (11 \cdot X_{45} + 17)\ (mod\ 50) = (11 \cdot 48 + 17)(mod\ 50) = 45$$

$$X_{47} = (11 \cdot X_{46} + 17)\ (mod\ 50) = (11 \cdot 45 + 17)(mod\ 50) = 12$$

$$X_{48} = (11 \cdot X_{47} + 17)\ (mod\ 50) = (11 \cdot 12 + 17)(mod\ 50) = 49$$

$$X_{49} = (11 \cdot X_{48} + 17)\ (mod\ 50) = (11 \cdot 49 + 17)(mod\ 50) = 6$$

$$X_{50} = (11 \cdot X_{49} + 17)\ (mod\ 50) = (11 \cdot 6 + 17)(mod\ 50) = 33$$

$$X_{51} = (11 \cdot X_{50} + 17)\ (mod\ 50) = (11 \cdot 33 + 17)(mod\ 50) = 30$$

$$X_{52} = (11 \cdot X_{51} + 17)\ (mod\ 50) = (11 \cdot 30 + 17)(mod\ 50) = 47$$

5) Setelah dilakukan perhitungan dapat dilihat bahwa dengan pengambilan nilai parameter $X_0 = 33$, $c = 17$, $m = 50$, dan $a = 11$, yang mana pengambilan nilai parameter telah memenuhi ketiga syarat maka didapatkan panjang periodenya sebesar m yaitu 50 dan lebih dari 15.

Setelah melakukan beberapa kali percobaan, periode penuh didapatkan pada percobaan keempat. Pada percobaan pertama, kedua, dan ketiga barisan tersebut

periodenya kurang dari m juga kurang dari panjang *plaintext* nya. Dalam hal ini, si A memutuskan yang akan dijadikan sebagai kunci dalam penyandian pesan adalah barisan bilangan acak yang dihasilkan pada percbaan keempat, karena memiliki periode penuh. Panjang *plaintext* sebesar 15 maka kunci yang akan digunakan adalah 30 47 34 41 18 15 32 19 26 3 0 17 4 11 38.

4.2 Proses Enkripsi Pesan menggunakan Algoritma *One Time Pad* (OTP) dan *Linear Congruential Generator* (LCG) sebagai pembangkit kunci

Proses enkripsi menggunakan algoritma *One Time Pad* (OTP) dengan *Linear Congruential Generator* (LCG) sebagai pembangkit kunci adalah proses penyandian pesan atau mengubah *plaintext* menjadi *ciphertekxt* dengan menggunakan kunci yang telah dibangkitkan menggunakan algortima LCG. Adapun proses enkripsi algoritma OTP sebagai berikut:

1. Menentukan *plaintext* atau pesan asli yang akan disandikan.
2. Mengkonversikan *plaintext* ke nilai karakter menurut tabel ASCII. Tabel ASCII disajikan pada lampiran.
3. Mengambil kunci yang telah dibangkitkan dari perhitungan LCG.
4. Melakukan proses enkripsi (penyandian) pesan dengan menggunakan algoritma OTP dimana nilai modulonya 256 (disesuaikan jumlah karakter pada tabel ASCII),

$$C_i = P_i + K_i (mod\ n)$$

5. Mengkonversi nilai hasil enkripsi ke karakter menurut tabel ASCII.
6. Mendapatkan pesan berupa pesan tersandi (*ciphertext*).

Adapun implementasinya sebagai berikut:

1. Si A akan mengirimkan sebuah pesan rahasia kepada si B, *plaintext* atau pesan asli yang akan disandikan adalah “Satgas COVID-19”.
2. *Plaintext* dikonversikan ke nilai desimal menurut tabel ASCII.

S = 83

a = 97

t = 116

g = 103

a = 97

s = 115

C = 67

O = 79

V = 86

I = 73

D = 68

−= 45

1 = 49

9 = 57

3. Pengambilan kunci yang akan digunakan dalam proses enkripsi pesan adalah barisan bilangan acak yang telah dibangkitkan dari perhitungan LCG yakni 30 47 34 41 18 15 32 19 26 3 0 17 4 11 38.
4. Melakukan proses enkripsi (penyandian) pesan dengan menggunakan algoritma OTP,

$$P_1 + K_1 (mod\ 256) = 83 + 30 (mod\ 256) = 113 = C_1$$

$$P_2 + K_2(mod\ 256) = 97 + 47(mod\ 256) = 144 = C_2$$

$$P_3 + K_3(mod\ 256) = 116 + 34(mod\ 256) = 150 = C_3$$

$$P_4 + K_4(mod\ 256) = 103 + 41(mod\ 256) = 144 = C_4$$

$$P_5 + K_5(mod\ 256) = 97 + 18(mod\ 256) = 115 = C_5$$

$$P_6 + K_6(mod\ 256) = 115 + 15(mod\ 256) = 130 = C_6$$

$$P_7 + K_7(mod\ 256) = 32 + 32(mod\ 256) = 64 = C_7$$

$$P_8 + K_8(mod\ 256) = 57 + 19(mod\ 256) = 76 = C_8$$

$$P_9 + K_9(mod\ 256) = 79 + 26(mod\ 256) = 105 = C_9$$

$$P_{10} + K_{10}(mod\ 256) = 8 + 3(mod\ 256) = 11 = C_{10}$$

$$P_{11} + K_{11}(mod\ 256) = 73 + 0(mod\ 256) = 73 = C_{11}$$

$$P_{12} + K_{12}(mod\ 256) = 68 + 17(mod\ 256) = 85 = C_{12}$$

$$P_{13} + K_{13}(mod\ 256) = 45 + 4(mod\ 256) = 49 = C_{13}$$

$$P_{14} + K_{14}(mod\ 256) = 49 + 11(mod\ 256) = 60 = C_{14}$$

$$P_{15} + K_{15}(mod\ 256) = 57 + 38(mod\ 256) = 95 = C_{15}$$

5. Mengkonversi nilai hasil enkripsi ke karakter menurut tabel ASCII.

$$113 = q$$

$$144 = \text{É}$$

$$150 = \text{û}$$

$$144 = \text{É}$$

$$115 = s$$

$$130 = \text{é}$$

$$64 = @$$

$$76 = L$$

$$105 = i$$

$$11 = VT$$

$$73 = I$$

$$85 = U$$

$$49 = 1$$

$$60 = <$$

$$95 = \_$$

6. *Chipertext* yang dihasilkan adalah qÉûÉsé@LiVTIU1<_.

Pesan qÉûÉsé@LiVTIU1<_ yang akan dikirimkan si A kepada si B beserta nilai parameter yang digunakan pada perhitungan LCG yakni $X_0 = 33$, $c = 17$, $m = 50$, dan $a = 11$.

4.3 Proses Dekripsi Pesan menggunakan Algoritma *One Time Pad* (OTP) dan *Linear Congruential Generator* (LCG) sebagai pembangkit kunci

Proses dekripsi menggunakan algoritma *One Time Pad* (OTP) dengan *Linear Congruential Generator* (LCG) sebagai pembangkit kunci adalah proses mengubah *ciphertext* menjadi *plaintekxt* dengan menggunakan kunci yang dibangkitkan menggunakan algortima LCG. Sebelum melakukan proses dekripsi si penerima harus melakukan perhitungan LCG terlebih dahulu untuk mendapatkan kunci karena kunci yang diberikan oleh pengirim hanya nilai parameternya saja. Adapun proses dekripsi algoritma OTP sebagai berikut:

1. Menerima *ciphertext* atau pesan tersandi dan nilai parameter yang digunakan pengirim untuk melakukan perhitungan LCG.
2. Melakukan perhitungan nilai LCG dengan menggunakan parameter yang sama seperti yang digunakan oleh pengirim.

3. Mengkonversikan *chipertext* ke nilai karakter menurut tabel ASCII. Tabel ASCII disajikan pada lampiran.
4. Melakukan proses dekripsi pesan dengan menggunakan algoritma OTP

$$P_i = C_i - K_i (mod\ n)$$

dimana nilai modulonya 256 (disesuaikan jumlah karakter pada tabel ASCII) dan mengunakan kunci yang telah dibangkitkan dari perhitungan LCG.

5. Mengkonversi nilai hasil dekripsi ke karakter menurut tabel ASCII.
6. Mendapatkan pesan berupa pesan asli (*plaintext*).

Adapun implementasinya sebagai berikut:

1. Si B menerima sebuah pesan tersandi yang dikirimkan oleh si A, *ciphertext* atau pesan tersandi tersebut adalah qÉûÉsé@LiVTIU1<_.
2. Si B menerima nilai parameter yang dikirimkan oleh oleh si A, nilai parameternya adalah $X_0 = 33$, $c = 17$, $m = 50$, dan $a = 11$.
3. Si B melakukan proses perhitungan LCG sampai X_{15} disesuaikan panjang *chipertext*nya.

$$X_1 = (11 \cdot X_0 + 17)\ (mod\ 50) = (11 \cdot 33 + 17)(mod\ 50) = 30$$

$$X_2 = (11 \cdot X_1 + 17)\ (mod\ 50) = (11 \cdot 30 + 17)(mod\ 50) = 47$$

$$X_3 = (11 \cdot X_2 + 17)\ (mod\ 50) = (11 \cdot 47 + 17)(mod\ 50) = 34$$

$$X_4 = (11 \cdot X_3 + 17)\ (mod\ 50) = (11 \cdot 34 + 17)(mod\ 50) = 41$$

$$X_5 = (11 \cdot X_4 + 17)\ (mod\ 50) = (11 \cdot 41 + 17)(mod\ 50) = 18$$

$$X_6 = (11 \cdot X_5 + 17)\ (mod\ 50) = (11 \cdot 18 + 17)(mod\ 50) = 15$$

$$X_7 = (11 \cdot X_6 + 17)\ (mod\ 50) = (11 \cdot 15 + 17)(mod\ 50) = 32$$

$$X_8 = (11 \cdot X_7 + 17)\ (mod\ 50) = (11 \cdot 32 + 17)(mod\ 50) = 19$$

$$X_9 = (11 \cdot X_8 + 17)\ (mod\ 50) = (11 \cdot 19 + 17)(mod\ 50) = 26$$

$$X_{10} = (11 \cdot X_9 + 17)\ (mod\ 50) = (11 \cdot 26 + 17)(mod\ 50) = 3$$

$$X_{11} = (11 \cdot X_{10} + 17)\ (mod\ 50) = (11 \cdot 3 + 17)(mod\ 50) = 0$$

$$X_{12} = (11 \cdot X_{11} + 17)\ (mod\ 50) = (11 \cdot 0 + 17)(mod\ 50) = 17$$

$$X_{13} = (11 \cdot X_{12} + 17)\ (mod\ 50) = (11 \cdot 17 + 17)(mod\ 50) = 4$$

$$X_{14} = (11 \cdot X_{13} + 17)\ (mod\ 50) = (11 \cdot 4 + 17)(mod\ 50) = 11$$

$$X_{15} = (11 \cdot X_{14} + 17)\ (mod\ 50) = (11 \cdot 11 + 17)(mod\ 50) = 38$$

4. Hasil yang didapatkan oleh si B sama seperti hasil yang didapatkan oleh si A, yakni 30 47 34 41 18 15 32 19 26 3 0 17 4 11 38.

Adapun proses dekripsi algoritma OTP sebagai berikut:

1. *Ciphertext* atau pesan tersandi adalah qÉûÉsé@LiVTIU1<_.
2. Mengkonversikan *ciphertext* ke nilai desimal menurut tabel ASCII

q = 113

É = 144

û = 150

É = 144

s = 115

é = 130

@ = 64

L = 76

i = 105

VT = 11

I = 73

U = 85

1 = 49

< = 60

_ = 95

3. Kunci yang akan digunakan adalah barisan bilangan acak yang dihasilkan dari perhitungan LCG yakni 30 47 34 41 18 15 32 19 26 3 0 17 4 11 38.
4. Melakukan proses dekripsi pesan dengan menggunakan algoritma OTP,

$$C_1 - K_1(mod\ 256) = 113 - 30(mod\ 256) = 83 = P_1$$

$$C_2 - K_2(mod\ 256) = 144 - 47(mod\ 256) = 97 = P_2$$

$$C_3 - K_3(mod\ 256) = 150 - 34(mod\ 256) = 116 = P_3$$

$$C_4 - K_4(mod\ 256) = 144 - 41(mod\ 256) = 103 = P_4$$

$$C_5 - K_5(mod\ 256) = 115 - 18(mod\ 256) = 97 = P_5$$

$$C_6 - K_6(mod\ 256) = 130 - 15(mod\ 256) = 115 = P_6$$

$$C_7 - K_7(mod\ 256) = 64 - 32(mod\ 256) = 32 = P_7$$

$$C_8 - K_8(mod\ 256) = 76 - 19(mod\ 256) = 57 = P_8$$

$$C_9 - K_9(mod\ 256) = 105 - 26(mod\ 256) = 79 = P_9$$

$$C_{10} - K_{10}(mod\ 256) = 11 - 3(mod\ 256) = 8 = P_{10}$$

$$C_{11} - K_{11}(mod\ 256) = 73 - 0(mod\ 256) = 73 = P_{11}$$

$$C_{12} - K_{12}(mod\ 256) = 85 - 17(mod\ 256) = 68 = P_{12}$$

$$C_{13} - K_{13}(mod\ 256) = 49 - 4(mod\ 256) = 45 = P_{13}$$

$$C_{14} - K_{14}(mod\ 256) = 60 - 11(mod\ 256) = 49 = P_{14}$$

$$C_{15} - K_{15}(mod\ 256) = 95 - 38(mod\ 256) = 57 = P_{15}$$

5. Mengkonversi nilai hasil dekripsi ke karakter menurut tabel ASCII

$$83 = S$$

$$97 = a$$

$$116 = t$$

$$103 = g$$

$$97 = a$$

$$115 = s$$

$$32 = spasi$$

$$67 = C$$

$$79 = O$$

$$86 = V$$

$$73 = I$$

$$68 = D$$

$$45 = -$$

$$49 = 1$$

$$57 = 9$$

6) Si B mendapatkan *plaintext* atau teks asli berupa “Satgas COVID-19”. Pesan tersebut sesuai seperti teks asli milik si A.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan pembahasan yang telah dijelaskan pada bab sebelumnya dapat disimpulkan sebagai berikut:

1. Algoritma LCG mampu membantu dalam memenuhi kebutuhan sebagai kunci dalam proses enkripsi dan dekripsi pesan pada agoritma OTP, sebab barisan bilangan yang dihasilkan oleh algoritma ini mampu menghindari penggunaan kunci yang rekursif dengan syarat pemilihan parameternya mengikuti Teorema 4.1 dan panjang periodenya lebih besar atau sama dengan panjang *plaintext*nya.
2. Pada proses enkripsi, pesan dikonversikan kedalam karakter pada tabel ASCII, sehingga tingkat keamanan penyandian pesan menjadi lebih tinggi sebab jumlah karakter yang digunakan lebih banyak. Selain itu, proses pengiriman pesan oleh pengirim menjadi lebih mudah sebab kunci yang harus dikirimkan tidak sepanjang teks aslinya melainkan hanya nilai parameternya saja.
3. Pesan tersandi (*ciphertext*) menjadi sulit terpecahkan sebab barisan kunci yang digunakan benar-benar acak sehingga bentuk pesan tersandipun menjadi acak dan tak terbaca. Namun, jika kunci yang digunakan tidak sesuai dengan milik pengirim maka *plaintext* yang dihasilkan tidak sama dengan milik pengirim.

5.2 Saran

Pada penelitian ini proses membangkitan bilangan acak yang digunakan sebagai kunci menggunakan algoritma LCG, pada penelitian selanjutnya disarankan untuk membuat aplikasi untuk melakukan perhitungan LCG agar lebih menghemat waktu atau dapat menggunakan metode lain yang lebih baik dalam membangkitkan bilangan acak agar kunci lebih rumit dan teracak. Dalam mengamankan pesan, peneliti menggunakan algoritma OTP. Pada penelitian selanjutnya disarankan menggunakan metode lain yang lebih bisa membuat pesan menjadi acak, tak terbaca dan sulit dipecahkan.

DAFTAR PUSTAKA

Al-Quran Terjemahan. 2015. *Departemen Agama RI*. Bandung: CV Darus Sunnah.

Damasyqi, Al-Imam Abul Fida Isma'il Ibnu Katsir. 2002. *Terjemah Tafsir Ibnu Katsir*. Bandung: Sinar Baru al-Gensido.

Easttom, William. 2021. *Modern Cryptography Applied Mathematics for Encryption and Information Security*. Jerman: Springer.

Glen, Amy. 2002. *On the Period Length of Pseudorandom Number Sequences*. Australia: The University of Adelaide.

Kessler, Gary C. 2010. *An Overview of Cryptography*. Amerika: Gary Kessler Associates.

Kneusel, Ronald T. 2018. *Random Numbers and Computers*. Jerman: Springer.

Knuth, Donald E. 1981. *The Art of Computer Programming*. Amerika: Addison-Wesley.

Mollin, Richard A. 2007. *An Introduction to Cryptography*. London: Taylor & Francis Group.

Munir, M. Wasim. 2005. *Cryptography*. Pakistan: SZABIST

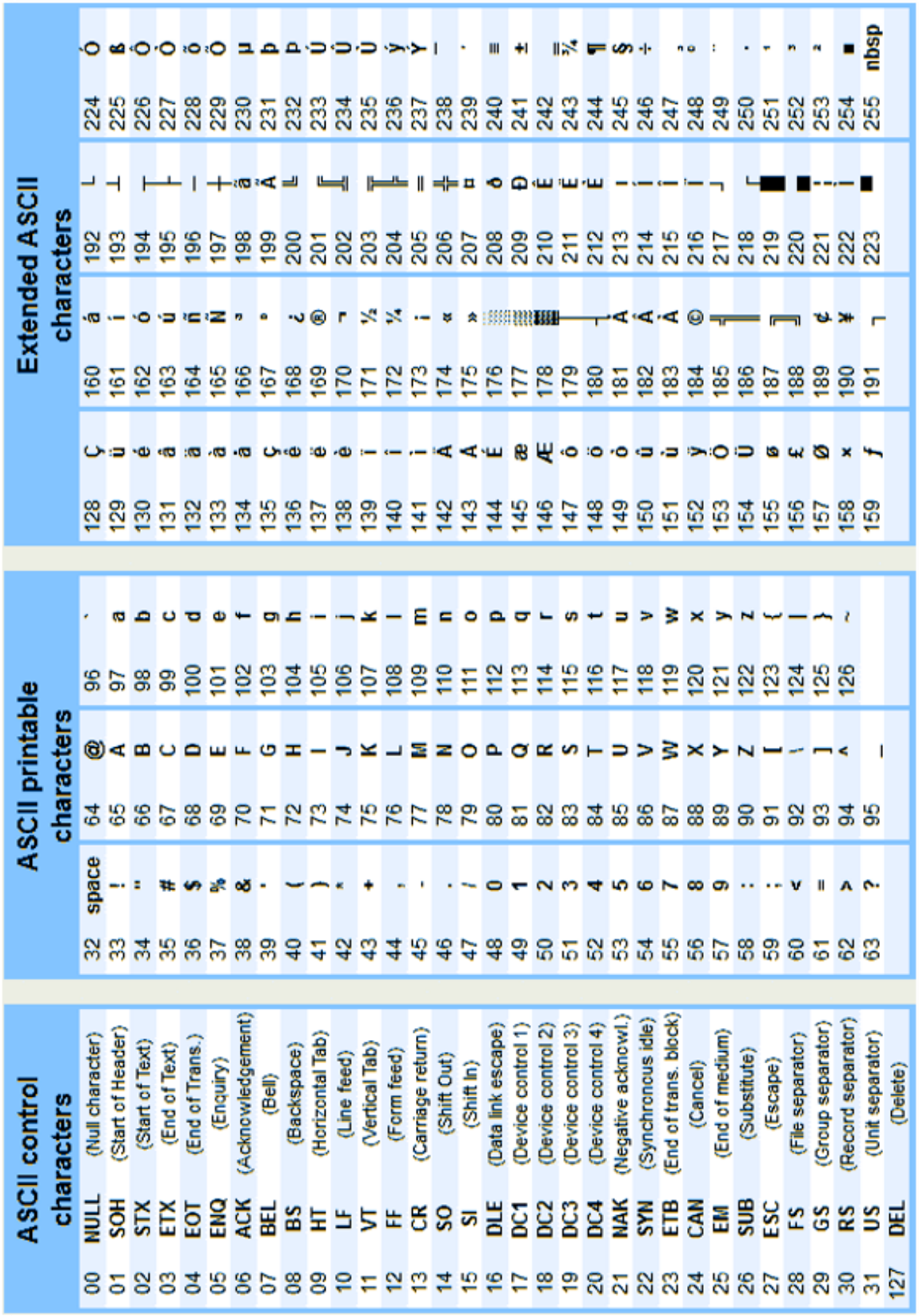
Rossen, Kenneth H. 2000. *Elementary Number Theory*. London: Addison-Wesley.

Stallings, William. 2005. *Cryptography and Network Security Principles and Practices*. Amerika: Pearson Education Inc.

Tezuka, Shu. 1995. *Uniform Random Numbers Theory and Practice*. Jerman: Springer.

LAMPIRAN

Tabel ASCIII



ASCII control characters			ASCII printable characters														Extended ASCII characters															
00	NULL	(Null character)	32	space	64	@	96	`									128	Ç	160	á	192	└	224	Ó
01	SOH	(Start of Header)	33	!	65	A	97	a									129	ü	161	í	193	┴	225	ß
02	STX	(Start of Text)	34	"	66	B	98	b									130	é	162	ó	194	┬	226	Ô
03	ETX	(End of Text)	35	#	67	C	99	c									131	â	163	ú	195	├	227	Ò
04	EOT	(End of Trans.)	36	$	68	D	100	d									132	ä	164	ñ	196	─	228	õ
05	ENQ	(Enquiry)	37	%	69	E	101	e									133	à	165	Ñ	197	┼	229	Õ
06	ACK	(Acknowledgement)	38	&	70	F	102	f									134	å	166	ª	198	ã	230	µ
07	BEL	(Bell)	39	'	71	G	103	g									135	ç	167	º	199	Ã	231	þ
08	BS	(Backspace)	40	(	72	H	104	h									136	ê	168	¿	200	╚	232	Þ
09	HT	(Horizontal Tab)	41	)	73	I	105	i									137	ë	169	®	201	╔	233	Ú
10	LF	(Line feed)	42	*	74	J	106	j									138	è	170	¬	202	╩	234	Û
11	VT	(Vertical Tab)	43	+	75	K	107	k									139	ï	171	½	203	╦	235	Ù
12	FF	(Form feed)	44	,	76	L	108	l									140	î	172	¼	204	╠	236	ý
13	CR	(Carriage return)	45	-	77	M	109	m									141	ì	173	¡	205	═	237	Ý
14	SO	(Shift Out)	46	.	78	N	110	n									142	Ä	174	«	206	╬	238	¯
15	SI	(Shift In)	47	/	79	O	111	o									143	Å	175	»	207	¤	239	´
16	DLE	(Data link escape)	48	0	80	P	112	p									144	É	176	▒	208	ð	240	≡
17	DC1	(Device control 1)	49	1	81	Q	113	q									145	æ	177	▓	209	Ð	241	±
18	DC2	(Device control 2)	50	2	82	R	114	r									146	Æ	178	▓	210	Ê	242	‗
19	DC3	(Device control 3)	51	3	83	S	115	s									147	ô	179	│	211	Ë	243	¾
20	DC4	(Device control 4)	52	4	84	T	116	t									148	ö	180	┤	212	È	244	¶
21	NAK	(Negative acknowl.)	53	5	85	U	117	u									149	ò	181	Á	213	ı	245	§
22	SYN	(Synchroncus idle)	54	6	86	V	118	v									150	û	182	Â	214	Í	246	÷
23	ETB	(End of trans. block)	55	7	87	W	119	w									151	ù	183	À	215	Î	247	¸
24	CAN	(Cancel)	56	8	88	X	120	x									152	ÿ	184	©	216	Ï	248	°
25	EM	(End of medium)	57	9	89	Y	121	y									153	Ö	185	╣	217	┘	249	¨
26	SUB	(Substitute)	58	:	90	Z	122	z									154	Ü	186	║	218	┌	250	·
27	ESC	(Escape)	59	;	91	[	123	{									155	ø	187	╗	219	▉	251	¹
28	FS	(File separator)	60	<	92	\	124	|								156	£	188	╝	220	▄	252	³
29	GS	(Group separator)	61	=	93	]	125	}									157	Ø	189	¢	221	¦	253	²
30	RS	(Record separator)	62	>	94	^	126	~									158	×	190	¥	222	▌	254	■
31	US	(Unit separator)	63	?	95	_											159	ƒ	191	┐	223	▀	255	nbsp
127	DEL	(Delete)																						

RIWAYAT HIDUP

Jamilatul Maghfiroh lahir di Pasuruan pada tanggal 20 Juli 1999. Memiliki nama panggilan Jamila. Alamat rumah di Dusun Sebani Desa Tanjangrono Kecamatan Ngoro Kabupaten Mojokerto. Merupakan anak ketiga dari Bapak Syamsuddin dan Ibu Tarlin Tasliatun Niswah.

Pendidikan yang pernah ditempuh adalah TK Manbaul Hikmah Sumberejo Pandaan pada tahun 2006-2007 kemudian melanjutkan ke jenjang selanjutnya SD Ma'arif Sumberejo Pandaan pada tahun 2007-2012. Melanjutkan jenjang SMP dan SMA di SMP-SMA AVISENA Jabon Sidoarjo pada tahun 2012-2017. Pada tahun 2017 melanjutkan pendidikan starta 1 di Universitas Islam Negeri Maulana Malik Ibrahim Malang dengan mengambil Program Studi Matematika, Fakultas Sains dan Teknologi.

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

BUKTI KONSULTASI SKRIPSI

Nama : Jamilatul Maghfiroh
NIM : 17610096
Fakultas/Jurusan : Sains dan Teknologi/Matematika
Judul Skripsi : Pengamanan Pesan Menggunakan Algoritma *One Time Pad* (OTP) dengan *Linear Congruential Generator* (LCG) Sebagai Pembangkit Kunci
Pembimbing I : Prof. Dr. H. Turmudi, M.Si.,Ph.D
Pembimbing II : Dr. Elly Susanti, M.Sc

No	Tanggal	Hal	Tanda Tangan
1	12 Februari 2021	Konfirmasi Bimbingan Proposal Skripsi	1
2	25 Maret 2021	Konsul Bab I, II, III	2
3	16 Maret 2021	Konsul Bab I, II, III	3
4	6 Maret 2021	Konsul Bab I, II, III dan Kajian Agama	4
5	22 April 2021	Revisi Bab I dan III	5
6	27 April 2021	Revisi Bab III	6
7	7 Mei 2021	Acc Pendaftaran Seminar Proposal	7
8	7 Mei 2021	Acc Pendaftaran Seminar Proposal	8
9	22 Februari 2022	Bimbingan Revisi Seminar Proposal	9
10	24 Februari 2022	Bimbingan Revisi Seminar Proposal	10
11	24 Maret 2022	Acc Pendaftaran Seminar Hasil	11
12	24 Maret 2022	Acc Pendaftaran Seminar Hasil	12

13	18 Mei 2022	Bimbingan Revisi Seminar Hasil	13
14	30 Mei 2022	Bimbingan Revisi Seminar Hasil	14
15	7 Juni 2022	Acc untuk ujian sidang	15
16	7 Juni 2022	Acc untuk ujian sidang	16

Malang, 21 Juni 2022
Mengetahui,
Ketua Program Studi Matematika

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