IMPLEMENTASI METODE SUPER ENKRIPSI *ATBASH CIPHER* DAN *RAIL FENCE CIPHER* PADA PENYANDIAN PESAN TEKS MENGGUNAKAN *PYTHON*

SKRIPSI

OLEH:
ANISA RAHMA FADHILA
NIM. 17610047

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024

IMPLEMENTASI METODE SUPER ENKRIPSI *ATBASH CIPHER* DAN *RAIL FENCE CIPHER* PADA PENYANDIAN PESAN TEKS MENGGUNAKAN *PYTHON*

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
ANISA RAHMA FADHILA
NIM. 17610047

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2024

IMPLEMENTASI METODE SUPER ENKRIPSI *ATBASH CIPHER* DAN *RAIL FENCE* CIPHER PADA PENYANDIAN PESAN TEKS MENGGUNAKAN *PYTHON*

SKRIPSI

Oleh
Anisa Rahma Fadhila
NIM. 17610047

Telah Disetujui Untuk Diuji
Malang, 26 Juni 2024

Dosen Pembimbing I	Dosen Pembimbing II
Prof. Dr. H. Turmudi, M.Si., Ph.D	Ach. Nashichuddin, M.A
NIP. 19571005 198203 1 006	NIP. 19730705 200003 1 002

Mengetahui,
Ketua Program Studi Matematika

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

IMPLEMENTASI METODE SUPER ENKRIPSI *ATBASH CIPHER* DAN *RAIL FENCE* CIPHER PADA PENYANDIAN PESAN TEKS MENGGUNAKAN *PYTHON*

SKRIPSI

Oleh
Anisa Rahma Fadhila
NIM. 17610047

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S.Mat)

Tanggal 28 Juni 2024

Ketua Penguji	: Dr. Hairur Rahman, M.Si	.....
Anggota Penguji 1	: Muhammad Khudzaifah, M.Si	.....
Anggota Penguji 2	: Prof. Dr. H. Turmudi, M.Si., Ph.D	.....
Anggota Penguji 3	: Ach. Nashichuddin, M.A	.....

Mengetahui,
Ketua Program Studi Matematika

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

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Anisa Rahma Fadhila

NIM : 17610047

Judul Skripsi : Implementasi Metode Super Enkripsi *Atbash Cipher* dan *Rail Fence Cipher* pada Penyandian Pesan Teks Menggunakan *Pyhton.*

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

Malang, 28 Juni 2024

Yang membuat pernyataan,

Anisa Rahma Fadhila

NIM. 17610047

MOTTO

"*Menjalani hidup seperti senja, kehadirannya selalu dinanti dan dinikmati serta kepergiannya akan selalu dikenang dan diceritakan*"

PERSEMBAHAN

Skripsi ini penulis persembahkan untuk:
Ayah Sukaryadi dan Ibu Siti Rodliyah serta adik penulis Pinot dan Zahro yang tak pernah bosan memberi semangat dan mendoakan penulis serta keluarga penulis khususnya keluarga Panti Asuhan Sunan Ampel yang telah menjadi motivasi penulis dalam menuntut ilmu dan mengabdi untuk menjadi pribadi yang lebih baik.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Segala puji bagi Allah SWT. yang telah melimpahkan rahmat, taufik serta hidayah-Nya. Sehingga penulis dapat menyelesaikan penyusunan skripsi yang berjudul "Implementasi Metode Super Enkripsi *Atbash Cipher* dan *Rail Fence Cipher* pada Penyandian Pesan Teks Menggunakan *Python*", sebagai syarat untuk memperoleh gelar pada bidang matematika di Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Shalawat serta salam selalu terlimpahkan kepada Nabi Muhammad SAW. yang telah menuntun kita kejalan yang benar.

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

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim.
2. Prof. Dr. Hj. Sri Harini, M.Si, selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim.
3. Dr. Elly Susanti, M.Sc, selaku ketua Program Studi Matematika, Universitas Islam negeri Maulana Malik Ibrahim.
4. Prof. Dr. H.Turmudi, M.Si., Ph.D, selaku dosen pembimbing I yang telah memberikan bimbingan, nasihat, do'a serta semangat dalam menyelesaikan skripsi ini.
5. Ach. Nasihchuddin, M.A., selaku dosen pembimbing II yang telah memberikan arahan, bimbingan, serta do'a dalam menyelesaikan skripsi ini.
6. Segenap sivitas akademika Program Stusi Matematika, Fakultas Sains dan Teknologi, dan Universitas Islam Negeri Maulana Malik Ibrahim Malang, terutama seluruh dosen yang telah memberikan arahan, ilmu serta do'a nya.
7. Ayah Sukaryadi dan ibu Siti Rodliyah yang tidak bosan selalu memberikan do'a, semangat serta motivasi kepada panulis hingga saat ini.
8. Kedua adik saya Alpinot dan Azzaro yang selalu menjadi semangat penulis.

9. Seluruh teman-teman Matematika angkatan 2017, terutama Kakak Wahyu Nurlaili, Fika ‘Banana’ Wahyuni, Icha Zakiyya, Ester ‘Bambang’ Meyliana yang telah berjuang bersama dan memberi penulis kenangan luar biasa pada masa perkuliahan serta tak lupa teman penulis Excel dan Riyan yang telah banyak membantu penulis menghadapi kerasnya dunia perkuliahan.
10. Keluarga Panti Asuhan Sunan Ampel yang telah menjadi motivasi penulis untuk mejadi pribadi yang kuat dan sabar, terutama adik Ntim dan adik-adik lainnya.
11. Mbah Mail dan Bu Sol yang telah membatu mengutkan mental dan batin penulis.
12. Semua pihak yang tidak disebutkan satu-persatu yang telah terdampak atas pengerjaan skripsi ini dan telah memberikan bantuan secara langsung maupun tidak langsung kepada penulis dalam menyelesaikan skripsi ini.
13. Serta untuk Sarah yang telah berjuang dan bertahan hingga akhir.

Semoga Allah SWT. melimpahkan seluruh rahmat serta karunia-Nya kepada kita semua. Akhirnya penulis mengharapkan semoga skripsi ini dapat memberikan manfaat dan wawasan kepada para pembaca. Aamiin.

Wassalamu’alaikum Warahmatullahi Wabarakatuh

Malang, 28 Juni 2024

Penulis

DAFTAR ISI

HALAMAN JUDUL ..... **i**
HALAMAN PENGAJUAN ..... **ii**
HALAMAN PERSETUJUAN ..... **iii**
HALAMAN PENGESAHAN ..... **iv**
PERNYATAAN KEASLIAN TULISAN ..... **v**
MOTTO ..... **vi**
PERSEMBAHAN ..... **vii**
KATA PENGANTAR ..... **viii**
DAFTAR ISI ..... **x**
DAFTAR TABEL ..... **xii**
DAFTAR GAMBAR ..... **xiii**
DAFTAR LAMPIRAN ..... **xiv**
ABSTRAK ..... **xv**
ABSTRACT ..... **xvi**
مستخلص البحث ..... **xvii**
BAB I PENDAHULUAN **1**
1.1 Latar Belakang 1
1.2 Rumusan Masalah 4
1.3 Tujuan Penelitian 4
1.4 Manfaat Penelitian 4
1.5 Batasan Masalah 5
1.6 Definisi Istilah 5
BAB II KAJIAN TEORI **7**
2.1 Kriptografi 7
2.1.1 Sejarah Kriptografi 7
2.1.2 Klasifikasi Algoritma Kriptografi 8
2.2 Metode *Atbash Cipher* Kunci Geser 12
2.3 Metode Rail Fence Cipher 16
2.3.1 Enkripsi Algoritma *Rail Fence Cipher* 17
2.3.2 Dekripsi Algoritma *Rail Fence Cipher* 18
2.4 Super Enkripsi 18
2.5 Python 19
2.6 Kewajiban Menjaga Amanah 20
BAB III METODE PENELITIAN **23**
3.1 Jenis Penelitian 23
3.2 Pra Penelitian 23
3.3 Tahapan Penelitian 23
BAB IV PEMBAHASAN **28**
4.1 Proses Penyandian Menggunakan Super Enkripsi *Atbash Cipher* dan *Rail Fence Cipher* 28
4.1.1 Proses Penyandian Metode *Atbash Cipher* Kunci Geser 28
4.1.2 Proses Penyandian Metode *Rail Fence Cipher* 34
4.1.3 Proses Penyandian Metode Super Enkripsi *Atbash Cipher* Kunci Geser dan *Rail Fence Cipher* 37

4.2. Implementasi pada *Pyhton*..... 49
4.3 Penyandian Menggunakan Metode Super Enkripsi *Atbash Cipher* dan *Rail Fence Cipher* dalam Pandangan Islam..... 54
BAB V PENUTUP..... 56
5.1 Kesimpulan..... 56
5.2 Saran..... 57
DAFTAR PUSTAKA..... 58
LAMPIRAN..... 60

DAFTAR TABEL

Tabel 2.1	*Atbash Cipher*	13
Tabel 2.2	*Atbash Cipher* dalam Indeks angka	14
Tabel 2.3	*Atbash Cipher* kunci geser (Key=2)	15
Tabel 2.4	Pengacakan Zig-zag	17
Tabel 2.5	Membaca Hasil Pengacakan	17
Tabel 2.6	Dekripsi *Rail Fence*	18
Tabel 4.1	*Konversi Atbash Cipher*	29
Tabel 4.2	Pola pada *rail fence cipher*	35
Tabel 4.3	Urutan memasukkan karakter pada pola	35
Tabel 4.4	Hasil penysusunan pesan sesuai urutan pola	35
Tabel 4.5	Membaca pesan perbaris	36
Tabel 4.6	Pola pada *rail fence cipher*	36
Tabel 4.7	Urutan memasukkan karakter pada pola	37
Tabel 4.8	Hasil penyusunan pesan sesuai urutan pola	37
Tabel 4.9	Membaca pesan secara zig-zag	37
Tabel 4.10	Pola pada *rail fence cipher*	42
Tabel 4.11	Urutan memasukkan karakter pada pola	42
Tabel 4.12	Hasil penysusunan pesan sesuai urutan pola	43
Tabel 4.13	Membaca pesan perbaris	43
Tabel 4.14	Pola pada *rail fence cipher*	44
Tabel 4.15	Urutan pola memasukkan karakter pada pola	44
Tabel 4.16	Hasil penysusunan pesan sesuai urutan pola	45
Tabel 4.17	Membaca pesan secara zig-zag	45
Tabel 4.18	Hasil uji coba pesan teks pada program *python*	52

DAFTAR GAMBAR

Gambar 4.1	Input pesan teks awal.....	50
Gambar 4.2	Input kunci enkripsi.....	50
Gambar 4.3	Hasil enkripsi program *python*	50
Gambar 4.4	Input teks hasil enkripsi.....	51
Gambar 4.5	Input kunci dekripsi.....	51
Gambar 4.6	Hasil dekripsi program *pyhton*	51

DAFTAR LAMPIRAN

Lampiran 1.	Program Enkripsi Python	60
Lampiran 2.	Program Dekripsi Python	61
Lampiran 3.	Hasil Uji Coba Pada Program *Pyhton*	62

ABSTRAK

Fadhila, Anisa Rahma. 2024. **Implementasi Metode Super Enkripsi *Atbash Cipher* dan *Rail Fence Cipher* pada Penyandian Pesan Teks Menggunakan *Pyhton***. Skripsi Jurusan Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (1) Prof. Dr. H. Turmudi, M.Si., Ph.D, (2) Ach. Nashichuddin, M.A.

Kata Kunci: Metode Super Enkripsi *Atbash Cipher*, *Rail Fence* Cipher, *Python.*

Seiring berkembangnya zaman, berkembang pula keamanan informasi yang menimbulkan dampak positif maupun negatif. Kriptografi merupakan salah satu cara untuk mengamankan pesan. Pada kriptografi ada proses enkripsi dan dekripsi. Enkripsi merupakan penyandian pesan awal agar tidak bisa dibaca sedangkan dekripsi merupakan proses penguraian pesan yang telah disandikan menjadi pesan awal agar dapat dibaca. Agar proses penyandian lebih kuat maka dapat menggabungkan dua atau lebih metode penyandian dengan menggunakan metode super enkripsi. Metode super enkripsi merupakan metode untuk menggabungkan dua atau lebih metode enkripsi yaitu subtitusi dan trasformasi. Pada penelitian ini metode penyandian yang digunakan adalah metode penyandian *Atbash cipher* kunci geser dan *Rail Fence cipher*, sehingga nantinya dapat meningkatkan keamanan suatu pesan. Agar proses penyandian dapat berlangsung lebih cepat dan efektif maka proses penyandian dapat dibantu menggunakan program *python*. Hasil penyandian menggunakan metode super enkripsi *Atbash cipher* dan *Rail Fence cipher* pada *python* dengan membuat tabel *atbash cipher* dengan cara membalik urutan karakternya, menggesernya sebanyak kunci yang berupa bilangan bulat kedepan atau kebelakang, dilanjut mentrasnformasi pesan secara zig-zag, hingga diperoleh pesan teks hasil penyandian. Sedangkan, proses dekripsi berkebalikan dengan proses enkripsinya, dengan menginput pesan teks dan kunci maka hasil enkripsi langsung ditampakkan, begitu juga dengan proses dekripsi. Proses penyandian pesan teks dengan super enkripsi *Atbash cipher* dan *Rail Fence cipher* dapat dilakukan dengan menggunakan program *python* karena dapat membantu proses penyandian menjadi lebih cepat dan efektif. Program ini bisa digunakan untuk pesan teks singkat maupun pesan teks berupa paragraf.

ABSTRACT

Fadhila, Anisa Rahma. 2024. **Implementation of Atbash Cipher and Rail Fence Cipher Super Encryption Methods for Encoding Text Messages Using Python.** Thesis. Department of Mathematics, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Supervisor: (1) Prof. Dr. H. Turmudi, M.Si., Ph.D, (2) Ach. Nashichuddin, M.A.

Keywords: Super Encryption Method Atbash Cipher, Rail Fence Cipher, Python

As the times evolve, so does information security, which has both positive and negative impacts. Cryptography is one way to secure messages. In cryptography, there are encryption and decryption processes. Encryption is the encoding of the initial message so that it cannot be read while decryption is the process of decoding the encoded message into the initial message so that it can be read. In order to make the encryption process stronger, it can combine two or more encryption methods by using the super encryption method. The super encryption method can combine two or more encryption methods, namely substitution and transformation. In this study using the encoding method modified Atbash cipher sliding key and Rail Fence cipher. So that later it can increase the security of a message. In order for the encryption process to take place more quickly and effectively, the encryption process can be assisted using a python program. The encryption results use the super encryption method modified Atbash cipher and Rail Fence cipher in python by creating an Atbash cipher table by reversing the order of the characters, shifting it as much as the key in the form of an integer forward or backward, continuing to transform the message in a zigzag manner, until the encoded text message is obtained. Meanwhile, the decryption process is the opposite of the encryption process, by inputting the text message and key, the encryption result is immediately displayed, as well as the decryption process. The process of encrypting text messages with super encryption modified Atbash cipher and Rail Fence cipher can be done using the python program because it can help the encryption process to be faster and more effective. This program can be used for short text messages or text messages in the form of paragraphs.

مستخلص البحث

أنيسة رحمة فضيلة. 2024. *إستخدام طريقة التشفير الفائق Atbash Cipher و Rail Fence لترميز الرسائل النصية بلغة Python. بحث جامعي، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرفان: (1) أ.د.ترمدي الحاج (2) أحمد نشيخ الدين لماجستير.*

الكلمات الأساسيّة: طريقة التشفير الفائقة *Atbash Cipher و Rail Fence و Python*

مع تطور العصر، يتطور أمن المعلومات أيضاً، وهو أمر له آثار إيجابية وسلبية على حد سواء. التشفير هو إحدى طرق تأمين الرسائل. في التشفير، هناك عمليات تشفير وفك تشفير. التشفير هو ترميز الرسالة الأولية بحيث لا يمكن قراءتها بينما فك التشفير هو عملية فك تشفير الرسالة المشفرة إلى الرسالة الأولية بحيث يمكن قراءتها. لجعل عملية التشفير أقوى، يمكننا الجمع بين طريقتين أو أكثر من طرق التشفير باستخدام طريقة التشفير الفائق. إن طريقة التشفير الفائق هي طريقة للجمع بين طريقتين أو أكثر من طرق التشفير، وهي الاستبدال والتحويل. في هذا البحث، طريقة التشفير المستخدمة هي طريقة ***atbash cipher***. المفتاح المنزلق ***rail fence cipher***، بحيث يمكن فيما بعد زيادة أمان الرسالة. ولكي تتم عملية التشفير بسرعة وفعالية أكبر، يمكن المساعدة في عملية التشفير باستخدام برنامج ***pyhthon***. تستخدم نتائج التشفير طريقة التشفير الفائق ***atbash cipher*** و***rail fence cipher*** في لغة بايثون عن طريق إنشاء جدول ***atbash cipher*** عن طريق عكس ترتيب الأحرف, وتحويله بقدر المفتاح الذي هو عبارة عن عدد صحيح إلى الأمام أو الخلف الاستمرار في تحويل الرسالة بطريقة متعرجة، حتى يتم الحصول على الرسالة النصية المشفرة. وفي الوقت نفسه، فإن عملية فك التشفير هي عكس عملية التشفير، من خلال إدخال الرسالة النصية والمفتاح، يتم عرض نتيجة التشفير على الفور، وكذلك عملية فك التشفير. يمكن إجراء عملية تشفير الرسائل النصية باستخدام الفائق ***atbash cipher*** و ***rail fence cipher*** باستخدام برنامج ***python*** لأنه يمكن أن يساعد في أن تكون عملية التشفير أسرع وأكثر فعالية. يمكن استخدام هذا البرنامج للرسائل النصية القصيرة أو الرسائل النصية على شكل فقرات.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Keamanan informasi menjadi masalah yang akan selalu ada dari zaman dahulu hingga sekarang. Keamanan informasi meliputi banyak hal, misalnya saja mencegah pihak-pihak yang tidak bertanggungjawab dari pengaksesan informasi, melindungi kerahasiaan informasi, pencegahan perubahan informasi, dan lain sebagainya. Banyak cara yang dapat dilakukan untuk mencegah pihak-pihak yang tidak bertanggungjawab mengakses informasi yang kita miliki, apalagi dizaman yang hampir seluruh informasi bisa diakses pada jaringan internet. Salah satu cara untuk mengatasinya adalah menggunakan kriptografi (Paryati, 2008).

Di zaman yang terus berkembangi ini, teknologi menghadirkan kemudahan untuk semua orang dalam membantu aktifitasnya, termasuk jaringan internet. Jaringan ini sudah diakses hamper seluruh orang didunia, maka dari itu akses mengirim pesan atau file ke berbagai tempat yang ingin dituju jauh lebih mudah dari biasanya. Namun karena dapat mengakses suatu hal dengan mudah akan timbul masalah keamanan, karena semua orang berada pada jaringan internet yang sama. Oleh karena itu sistem keamanan juga harus diperkuat ketika akan mengirim suatu pesan atau file yang penting agar tidak disalah gunakan oleh orang yang tidak bertanggung jawab (Sugiyanti dkk. 2022).

Setiap orang pasti ingin pesan yang mereka kirimkan tetap aman. Dalam Islam menyampaikan suatu pesan murapakan suatu amanah yang harus dilaksanakan. Memenuhi amanah yang diberikan sangatlah penting, salah satunya

bisa lebih mendekatkan diri kepada Allah SWT. juga dapat meningkatkan rasa percaya orang atau masyarakat kepada orang yang telah amanah tersebut. Pentingnya menyampaikan amanah juga tertulis dalam Al Qur'an Surat Al-Mu'minun Ayat 8 (Kementrian Agama RI):

وَالَّذِيْنَ هُمْ لِاَمٰنٰتِهِمْ وَعَهْدِهِمْ رَاعُوْنَۙ

"(Sungguh beruntung pula) orang-orang yang memelihara amanat dan janji mereka."(QS. Al-Mu'minun: 8)

Pada ayat tersebut telah dijelaskan dalam tafsir Ibnu Katsir bahwa menjaga amanah dan menepati janji adalah tanda dari kesempurnaan iman seseorang, dan dua hal tersebut harus dijaga oleh setiap individu agar tercipta masyarakat yang adil dan terpercaya. Ibnu Katsir juga menambahkan dalam bahwa karakterisitk ini adalah bagian dari sifat orang-orang yang nantinya akan mendapatkan kebahagiaan dunia dan akhirat (Abdullah, 2007).

Banyak cara yang bisa dilakukan untuk mengamankan pesan salah satunya kriptografi. Kriptografi adalah ilmu untuk menjaga keamanan pesan dengan cara meyandikan pesan kedalam bentuk yang tidak dipahami maknanya. Kriptografi menjadi salah satu pilihan untuk mengamankan suatu pesan yang akan kita kirim. Seiring dengan berjalannya waktu banyak orang yang telah mempelajari kriptografi ini, baik untuk mengamankan pesannya atau orang yang menguraikan pesan tersebut (kripanalis). Akan menjadi suatu masalah jika pesan yang kita kirimkan diuraikan oleh kripanalis yang bukan tujuan kita. Oleh karena itu dibutuhkan metode untuk menenkripsi suatu pesan agar keamanan pesan tetap terjaga. Namun saat ini sudah banyak orang yang mempelajarainya. Sehingga diperlukan suatu metode yang lebih rumit dari biasanya agar tidak mudah diuraikan (Meyer, 1982).

Metode pada kriptografi juga bermacam-macam. Metode tersebut diguanakan sesuai dengan kebutuhan, karena sudah banyak orang yang belajar kriptografi maka metode yang digunakan bisa lebih dari satu dan bisa di modifikasi. Misalnya menggunakan Metode super enkripsi, yaitu metode untuk mengenkripsikan pesan denga dua kali enkripsi. Namun untuk saat ini ketika menggunakan metode lebih dari satu, cara enkripsinya masih rumit dan memakan waktu. Padahal yang dibutuhkan bukan hanya keamanan yang rumit, namuan juga kecepatan enkripsi dan dekripsi suatu pesan. Enkripsi sendiri berarti proses menyamarkan pesan asli kedalam sebuah sandi yang telah disepakati. Sedangkan dekripsi adalah proses menguraikan pesan yang telah dienkripsi sebelumnya (Primartha, 2011).

Python adalah salah satu bahasa pemrograman yang dipakai secara luas pada dunia pendidikan dan industry karena bahasanya yang ringkas, sederhana, sintak sinuitif dan pustakanya luas. Python juga dapat membantu proses enkripsi dan dekripsi agar bisa lebih cepat dari pada mengenkripsi secara manual. Oleh karna itu dibuatlah suatu program metode super enkripsi menggunakan *Python* agar proses penyandian metode super enkripsi bisa memakan waktu yang singkat (Schuerer dan Maufrais, 2010).

Dari latar belakang tersebut masalah yang ingin dikembangkan adalah untuk membuat kemanan yang rumit dan dapat dilakukan dengan program agar tidak memakan banyak waktu.

1.2 Rumusan Masalah

Berdasarkan dari uraian latar belakang sebelumnya, maka rumusan masalah yang dapat diambil pada penelitian ini adalah

1. Bagaimana proses penyandian menggunakan metode super enkripsi *Atbash Cipher* dan *Rail Fence Cipher*?
2. Bagaimana implementasi metode super enkripsi *Atbash Cipher* dan *Rail Fence Cipher* menggunakan *Python*?

1.3 Tujuan Penelitian

Adapun tujuan yang ingin dicapai berdasarkan permasalahan yang ada di atas adalah

1. Untuk mengetahui proses penyandian menggunakan metode super enkripsi *Atbash Cipher* dan *Rail Fence Cipher*
2. Untuk mengetahui implementasi metode super enkripsi *Atbash Cipher* dan *Rail Fence Cipher* menggunakan *Python*

1.4 Manfaat Penelitian

Adapun manfaat dilakukannya penelitian ini adalah sebagai berikut:

1. Dengan bantuan komputer, dapat membantu mengamankan pesan teks yang akan dikirimkan
2. Sebagai bahan referensi untuk mengembangkan penelitian berikutnya
3. Sebagai bahan kepustakaan yang dijadikan sarana pengembangan wawasan kailmuan khususnya dibidang matematika

1.5 Batasan Masalah

Agar penulisan laporan ini tidak meluas pembahasannya dan jauh dari tujuan penelitian yang telah direncanakan sejak awal, maka penulis menetapkan batasa-batasan masalah agar dapat mempermudah mendapatkan data dan informasi yang diperlukan sebagai berikut:

1. Algoritma yang digunakan adalah metode super enkripsi (*Atbash Cipher* dan *Rail Fence Cipher*)
2. Penerapan penelitian ini dilakukan pada pesan teks
3. Program yang digunakan hanya *Python*

1.6 Definisi Istilah

Berdasarkan rumusan dan fokus dan rumusan masalah penelitian, maka uraian definisi istilah dalam penelitian ini adalah sebagai berikut:

1. Metode Super Enkripsi

   Metode Super Enkripsi adalah kombinasi dari dua atau lebih metode subtitusi dan transposisi untuk mendapatkan suatu algoritma dalam menyandikan pesan untuk berkomunikasi tanpa diketahui oleh pihak ketiga.

2. Metode *Atbash Cipher*

   Metode *Atbash Cipher* merupakan salah suatu teknik enkripsi, yang huruf alphabetnya disubstitusi dengan kebalikannya.

3. Metode *Rail Fence Cipher*

   Metode *Rail Fence Cipher* merupakan salah satu metode enkripsi transposisi yang sederhana yang terinspirasi dari model *Polybius square*. Polybius square sendiri maksudnya adalah menyusun huruf sebagai matriks

5x5 kemudian huruf A dikodekan menjadi 1-1, huruf B menjadi 1-2 dan seterusnya.

4. *Python*

Python adalah salah satu program yang digunakan untuk mempermudah pengerjaan sesuatu, dalam penelitian ini *Python* berfungsi mempermudah proses mengenkripsi dan mendekripsi menggunakan metode super enkripsi *Atbash Cipher* dan *Rail Fence Cipher*.

BAB II

KAJIAN TEORI

2.1 Kriptografi

Kriptografi atau dalam bahasa Yunani berarti rahasia atau tersembunyi pada umumnya disebut dengan ilmu penyandian data merupakan salah satu bidang ilmu dan seni (*art and science*) yang memiliki tujuan mengamankan suatu rahasia pada suatu pesan yang dikirimkan (Sutoyo, dkk., 2009).

Kriptografi pada masa modern ini dapat dianggap sebagai pengaplikasian operasian struktur matematika tertentu (Sadiki, 2012). Pada kriptografi ada 3 fungsi dasar, diantaranya:

1. Enkripsi, yaitu proses penyandian terhadap suatu pesan atau data
2. Dekripsi, yaitu proses penguraian pesan yang telah di enkripsikan
3. *Key*, yaitu kunci yang biasa dipakai pada proses mngenkripsi maupun mendekripsi. Pada umumnya algoritma enkripsi maupun dekripsi bergantung pada kunci (*key*) rahasia. Kunci atau *key* rahasia ini biasanya berupa simbol, alphabet, angka maupun simbol barisan bi-bi.

2.1.1 Sejarah Kriptografi

Sejarah mencatat kriptografi telah ada kurang lebih sekitar tahun 400 SM atau pada masa kejayaan Yunani. Dengan menggunakan alat yang bernama Scytale, pesan yang akan dikirimkan terjaga kerahasiaannya. Bentuk dari Scytale adalah silinder dengan kombinasi 18 huruf. (Ariyus, 2006)

Masih menurut Ariyus (2006) pada masa Romawi, saat kekuasaan dipegang oleh Julius Caesar, kriptografi digunakan secara intens untuk menjaga kestabilan negara. Meski tekniknya tidak serumit yang digunakan pada masa Yunani, namun untuk menguraikan pesan yang dirahasiakan terbilang cukup susah.

Berdasarkan perumpamaan tersebut, ada beberapa istilah kriptografi yang digunakan untuk memudahkan proses rahasia dalam pengiriman pesan. Proses yang biasa dilakukan Julius Caesar untuk mengacak pesan, disebut dengan istlah enkripsi. Sebaliknya pada saat sang jenderal menguraikan pesan yang teracak itu disebut dengan dekripsi. Lalu untuk pesan awal yang belum diacak biasa disebut dengan *plaintext* dan sedang pesan yang telah diuraikan disebut cipherteks (Ariyus, 2006)

2.1.2 Klasifikasi Algoritma Kriptografi

Berikut adalah beberapa klasifikasi kriptografi:

1. Berdasarkan Kunci

   Menurut, berdasarkan kuncinya algoritma kriptografi dibagi menjadi tiga, pertama algoritma simetri, algoritma asimetri, dan fungsi hash, sebagai berikut: (Ariyus, 2008)

   a. Algoritma Simetri

      Algoritma simetri menggunakan kunci yang sama pada baik pada proses enkripsi maupun proses dekripsi sehingga biasa disebut algoritma klasik. Algoritma simetri telah digunakan sejak lebih dari 4.000 tahun yang lalu. Ketika menggunakan algoritma ini saat mengirim pesan, penerima pesan

harus diberi tahu kunci yang digunakan pada pesan tersebut agar dapat menguraikannya. Kunci dari algoritma ini juga mempengaruhi tingkat keamanan pesan yang dikirimkan. Kelemahan dari algoritma ini adalah ketika kunci pesannya diketahui orang lain, karena siapapun yang mengetahui kunci pesannya berarti ia dapat menguraikan pesan yang telah dienkripsi. Berikut adalah beberapa algoritma yang menggunakan kunci simetri: (Ariyus, 2008)

1. International Data Encryption Algorithm (IDEA),
2. A5,
3. On Time Pad (OTP),
4. RC 2, RC 4, RC 5, RC 6;
5. Advanced Encryption Standard (AES),
6. Data Encryption Standard (DES), dan lain sebagainya.

b. Algoritma Asimetri

Algoritma asimetri atau biasa dikenal dengan algoritma kunci publik, digunakan pada proses enkripsi dan dekripsi dengan kunci yang berbeda. Dalam algoritma asimetri, kunci akan dibagi menjadi dua, yang pertama adalah kunci umum atau biasa disebut dengan *public key*, yaitu kunci yang boleh diketahui oleh semua orang atau bisa dipublikasikan. Kemudian yang berikutnya adalah kunci rahasia atau private key, yaitu kunci yang dirahasiakan dan hanya boleh diketahui oleh satu orang saja. Kedua kunci tersebut saling berhubungan. Fungsi dari kunci public adalah agar orang dapat mengenkripsikan pesan namun orang tersebut tidak

dapat mendekripsikannya karena yang memegang private key hanya satu orang. Dan hanya satu orang tersebut yang dapat mendekripsikan pesan tersebut. Daripada algoritma simetri, algoritma asimetri terbilang lebih aman dan berikut adalah beberapa contoh algoritma asimetri: (Ariyus, 2008)

1. Digital Signature Algorithm (DSA),
2. Kriptografi Quantum,
3. Elliptic Curve Cryptography (ECC),
4. Diffle-Hellman (DH),
5. RSA, dan lain sebagainya.

c. Fungsi Hash

Fungsi Hash memiliki sebutan lain diantaranya bisa disebut dengan fungsi message digest , fingerprint, funsi satu arah (one-way function), fungsi kompresi, dan message authentication code (MAC). Fungsi Hash adalah salah satu fungsi matematika yang menggunakan masukan panjang variabel kemudian diubah ke dalam urutan biner yang panjangnya tetap. Fungsi ini biasanya digunakan ketika suatu pesan akan dibuatkan sidik jari. Sidik jari pada pesan adalah salah satu tanda bahwa pesan yang dikirimkan tersebut benar adanya berasal dari orang-orang yang mengirimkan. (Ariyus, 2008)

2. Berdasarkan Jenisnya

Kriptografi jika diklasifikasikan berdasarkan jenis menjadi dua, yaitu klasik dan modern sebagai berikut:

a. Kriptografi klasik

Kriptografi klasik adalah kriptografi yang sudah digunakan dizaman ketika komputer belum ditemukan atau saat komputer sudah ditemukan tetapi belum begitu canggih seperti saat ini. Kriptografi klasik melakukan pengacakan huruf pada *plaintext* atau pesan awal. Kriptografi klasik biasanya hanya melakukan pengacakan pada terhadap huruf A sampai Z dan oleh karena itu tidak dianjurkan untuk mengamankan informasi yang sangat penting karena algoritmanya masih mudah untuk diselesaikan dengan cepat. Meskipun tidak lagi digunakan untuk pengamanan pesan dimasa sekarang, kriptografi ini masih ada sebagai pengantar ilmu kriptografi modern. Ada beberapa ciri dari kriptografi klasik, diantaranya: (Ariyus, 2008)

1. Menggunakan pena dan kertas saja, belum ada computer,
2. Berbasis karakter,
3. Termasuk ke dalam kriptografi kunci simetris,
4. Menggunakan pena dan kertas saja, belum ada computer.

b. Kriptografi Modern

Kriptografi modern adalah suatu kriptografi yang memperbaiki kriptografi klasik. Ada banyak macam algoritma pada kriptografi modern, hal tersebut bertujuan agar informasi yang dikirim melalui jaringan

komputer menjadi lebih aman. Pada umumnya kriptografi modern dioperasikan dalam mode bit. Tidak seperti kriptografi klasik yang dioperasikan dalam mode karakter (seperti yang dilakukan pada cipher substitusi atau cipher transposisi dari algoritma kriptografi klasik).

Pengoperasian pada mode bit maksudnya adalah menyatakan dalam bentuk bit biner, 0 dan 1 setiap informasi juga data (*plaintext*, cipherteks dan kunci) yang akan disandikan. Jadi algoritma enkripsi dan dekripsi mengubah semua informasi dan data menjadi rangkaian bit 0 dan 1. (Ariyus, 2008)

2.2 Metode *Atbash Cipher* Kunci Geser

Metode subtitusi adalah salah satu jenis metode untuk mengekripsi dimana setiap satuan pada teks asli digantikan oleh teks yang telah tersandi dengan sistem yang teratur satuan yang dimaksud dapat diartikandengan satu huruf (pada umumya), pasangan hurus, suku kata, kata, dan sebagainya. Penerima pesan yang sudah tersandi dapat membaca pesan jika sudah melakukan subtitusi balik terlebih dahulu. Pada penyandian menggunakan metode subtitusi ini, satuan-satuan pada teks asli dapat diubah tetapi susunannya masih sama. Kebalikan dari sandi subtitusi ini adalah sandi trasnposisi, dimana satuan-satuan teks asli susunannya diacak sedemikian rupa sehingga susah untuk dibaca, namun tidak mnsubtitusi atau mengaanti huruf-huruf tersebut (Haryus, 2010).

Sandi *Atbash* adalah salah satu penyandiaan menggunakan teknik substitusi sederhana. Cara kerja penyandian *Atbash Cipher* adalah dengan cara membalikkan alfabet, oleh karena itu setiap huruf dipetakan ke huruf di posisi yang sama

kebalikan dari abjad. Awal mula percobaan *atbash* cipher dilakukan menggunakan abjad ibrani dan referensi Perjanjian Lama oleh karena itu *atbash cipher* juga sering dikaitkan dengan berbagai bentuk mistisisme. Kemudian dizaman modern ini disebut sebagai kode alfabet terbalik, *translater Atbash* ini (termasuk *encoder Atbash* dan *decoder Atbash*) dapat membantu mengenkripsi dan dekripsi kode pesan (Haryus, 2010).

Kemudian sejak tahun 600 SM sandi *Atbash telah* digunakan bangsa Yahudi mengganti alfabet Hebrew dengan korespondensi kebalikannya. Maka ketika diterapkan pada alfabet latin maka akan berupa:

Tabel 2.1 *Atbash Cipher*

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

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

Maka dari penerapan pada alfabeth tersebut menghasilkan rumus:

$$E_{(x)} = D_{(x)} = (-x \; mod \; m) + 1$$

keterangan:

$E_{(x)}$: Proses enkripsi

$D_{(x)}$: Proses Dekripsi

x : *Plaintext* atau *Ciphertext*

m : Jumlah karakter yang digunakan (Pada *Atbash Cipher* karakter yang digunakan ada 53, maka $m = 53$)

Kriptografi saat ini sudah banyak mengalami perkembangan yang pesat, termasuk kriptografi klasik. Namun, yang banyak dikembangkan pada kriptografi klasik adalah *caesar cipher*, tidak dengan Atbash yang tidak setenar *caesar cipher*. Pada umumnya *atbash cipher* masih mengenkripsikan hurus capital saja. Oleh karena itu dibuatlah yang lebih komplek dengan mengenkripsiskan huruf kapital, huruf kecil dan ‘spasi’(Latifah, 2017).

Sehingga tabel yang bisa didapat dengan menggunakan *Atbash Cipher* adalah sebagai berikut:

Tabel 2.2 *Atbash Cipher* dalam Indeks angka

Indeks	**Pi**	**Ci**
1	A	‘spasi'
2	B	z
3	C	y
4	D	x
5	E	w
6	F	v
7	G	u
8	H	t
9	I	s
10	J	r
11	K	q
12	L	p
13	M	o
14	N	n
15	O	m
16	P	l
17	Q	k
18	R	j

Indeks	**Pi**	**Ci**
19	S	i
20	T	h
21	U	g
22	V	f
23	W	e
24	X	d
25	Y	c
26	Z	b
27	a	a
28	b	Z
29	c	Y
30	d	X
31	e	W
32	f	V
33	g	U
34	h	T
35	i	S
36	j	R

Indeks	**Pi**	**Ci**
37	k	Q
38	l	P
39	m	O
40	n	N
41	o	M
42	p	L
43	q	K
44	r	J
45	s	I
46	t	H
47	u	G
48	v	F
49	w	E
50	x	D
51	y	C
52	z	B
53	‘spasi'	A

Setelah melakukan penyandian menggunakan metode *atbash cipher*, penyandian akan dilanjutkan dengan memodifikasi menggunakan kunci geser. Caranya adalah dengan menggeser huruf-huruf pada pesan sebanyak kunci (bilangan bulat) kedepan atau kebelakan. Atau bisa juga dengan menggunakan rumus perhitungan

$$E_{(E_x)} = (E_x - k)\ mod\ m$$

keterangan:

$E_{(E_x)}$: Proses enkripsi kunci geser

$E_{(E_x)}$: Hasil enkripsi pertama *atbash cipher*

x : *Plaintext* atau *Ciphertext*

k : Kunci

m : Jumlah karakter yang digunakan (Pada *Atbash Cipher* karakter yang digunakan ada 53, maka $m = 53$)

Misalkan kunci adalah 2 maka nantinya *ciphertext* akan bergeser sebanyak 2 kedepan, bisa dihitung menggunakan rumus dan hasilnya seperti pada tabel 2.3

Tabel 2.3 *Atbash Cipher* kunci geser (Key=2)

Indeks	**Pi**	**Ci**	**Indeks**	**Pi**	**Ci**	**Indeks**	**Pi**	**Ci**
1	A	B	19	S	k	37	k	S
2	B	A	20	T	j	38	l	R
3	C	‘spasi'	21	U	i	39	m	Q
4	D	z	22	V	h	40	n	P
5	E	y	23	W	g	41	o	O
6	F	x	24	X	f	42	p	N
7	G	w	25	Y	e	43	q	M
8	H	v	26	Z	d	44	r	L
9	I	u	27	a	c	45	s	K

Indeks	Pi	Ci
10	J	t
11	K	s
12	L	r
13	M	q
14	N	p
15	O	o
16	P	n
17	Q	m
18	R	l

Indeks	Pi	Ci
28	b	b
29	c	a
30	d	Z
31	e	Y
32	f	X
33	g	W
34	h	V
35	i	U
36	j	T

Indeks	Pi	Ci
46	t	J
47	u	I
48	v	H
49	w	G
50	x	F
51	y	E
52	z	D
53	‘spasi'	C

2.3 Metode Rail Fence Cipher

Metode penyandian transposisi merupakan salah satu cara menyandikan pesan yang prosesnya dilakukan dengan cara mengacak rangkaian karakter pada teks. Kemudian untuk membaca pesan yang telah diacak dapat dilakukan dengan cara mengembalikan letak dari pesan yang telah disandikan menggunakan kunci yang sudah ditetapkan diawal. (Munir, 2019).

Ada banyak metode transposisi yang bisa digunakan salah satunya adalah *Rail Fence Cipher (RFC)*. Metode enkripsi *Rail Fence* adalah salah satu bentuk cipher transposisi sederhana yang berawal dari model *Polybius square*. *Polybius square* sendiri adalah menyusun huruf sebagai matriks 5x5 kemudian huruf A dikodekan menjadi 1-1, huruf B menjadi 1-2 dan seterusnya. Masing-masing karakter pada *Polybius square* akan diubah menjadi indeks cell matriks dengan tidak menggunakan kunci khusus dan hanya akan merubah letak hingga teks tidak bisa terbaca. Metode *Rail Fence* sedikit berbeda dengan *Polybius square* karena

metode *Rail Fence* menyusun teks secara zig-zag dan model matriksnya diketahui oleh pengirim serta penerima pesan (Siahaan, 2016).

2.3.1 Enkripsi Algoritma *Rail Fence Cipher*

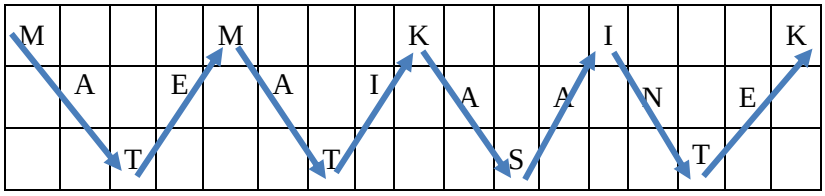
Berikut ini adalah proses pengenkripsian menggunakan algoritma transposisi:

Misalkan kita mempunyai *plaintext*: MATEMATIKASAINTEK

Kunci: 3 (yang artinya kolom = 3)

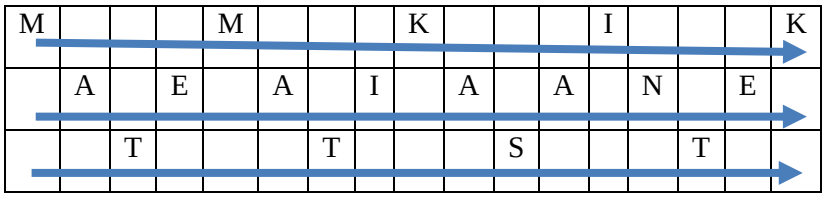
Untuk mendapatkan cipherteksnya, kita bagi *plaintext* ke tabel yang kuncinya tiga secara zigzag seperti pada table berikut ini

Tabel 2.4 Pengacakan Zig-zag



Setelah semua terisi kemudian baca teks secara mendatar perbaris diikuti baris berikutnya hingga semua teks tersusun.

Tabel 2.5 Membaca Hasil Pengacakan



Sehingga didapatkan

Cipherteks (*c*): MMKIKAEAIAANETTST

2.3.2 Dekripsi Algoritma *Rail Fence Cipher*

Berikut ini adalah proses pendekripsian menggunakan algoritma transposisi:

Cipherteks yang didapat dari proses enkripsi sebelumya adalah MMKIKAEAIAANETTST

Kunci: 3 baris = 3)

Untuk mendapatkan *plaintext* nya, kita bagi cipherteksnya ke tabel secara zigzag berdasarkan urutan baris ke-1, baris ke-2, dan baris ke-3

Tabel 2.6 Dekripsi *Rail Fence*

M				M				K				I				K
	A		E		A		I		A		A		N		E	
		T				T				S				T		

Setelah semua terisi kemudian baca teks sesuai dengan urutan kolom sehingga diperoleh *Plaintext* (*p*): MATEMATIKASAINTEK

2.4 Super Enkripsi

Kriptografi klasik pada dasarnya dibagi menjadi dua, yang pertama subsitusi dan yang kedua adalah permutasi. Kedua algoritma tersebut relatif mudah dipecakan melalui brute force. Maksud dari brute force yaitu menguraikan sandi dengan cara mencoba satu persatu sandi yang mungkin pada cipherteks hingga menjadi *plaintext*. Pada umumnya kunci yang dicoba adalah setengah dari kemungkinan yang diperkirakan akan berhasil (Stallings, 2003).

Oleh karena itu agar algoritma susah untuk ditemukan kuncinya oleh pikah yang tidak bertanggungjawab, maka dikembangkan algoritma super enkripsi, yaitu

algoritma yang menggabungkan dua teknik algoritma yang berbeda. Metode super enkripsi adalah suatu konsep penyandian yang menggunakan kombinasi dari dua atau lebih dari teknik subtitusi maupun transposisi agar mendapatkan suatu algoritma yang lebih susah untuk dipecahkan. Sebelum mengoperasikan teknik super enkripsi, harus paham terlebih dahulu tentang teknik subtitusi maupun transposisi. Karena metode super enkripsi dioperasikan dengan melakukan teknik subtitusi terlebih dahulu baru kemudian dilanjutkan menggunakan teknik transposisi dan dilakukan sebaliknya ketika akan menguraikan pesan yang telah tersandi (Ariyus, 2009).

2.5 Python

Python merupakan bahasa pemrogaman yang umum digunakan untuk kalangan engineer seluruh dunia pada pembuatan perangkat lunaknya. Python juga Bahasa pemrogaman yang *freeware* atau bisa diartikan tidak ada batasan saat menyalin atau mendistribusikannya. Lengkap dengan *debugger* dan *profiler*, *source code*, fungsi sistem, GUI, dan lain sebagainnya. Python juga telah menyediakan modul-modul siap pakau untuk berbagai keperluan. Meskipun python termasuk Bahasa pemrogaman dengan level tinggi, python sendiri sudah dirancan agar mudah dipahami dan dipelajari (Clinton dan Sengkey, 2019).

Python sendiri memiliki kelebihan dan kekurangan. Kekurangannya misalnya pada kecepatan eksekusi yang tidak secepat pemrogaman yang dikompilasi seperti C ataupun C++.. kelebihan Python menurur antara lain (Lutz, 2013)

1. Kualitas *software* Bahasa pemrograman Pythom dibuat agar mudah untuk dibaca, schingga memudahkan penggunaan kembali *source code* (code rausability) dan jika akan dilakukan perubakan, progamer juga lebih mudah untuk mengatur *source code* tersebut;
2. Deveoper atau pengembang dari Bahasa pemrogaman *python* juga lebih baik dari pada yang Bahasa pemrogaman yang lainnya, misalnya seperti C, C++ dan *Java Source code. Python* biasanya juga memiliki ukuran file yang lebih kecil dari Bahasa pemrogaman yang lainnya seperti C-atau Java.
3. Sebagian besar dari program yang menggunakan Bahasa pemrogaman *python* berjalan tanpa adanya perubahan pada perangkat yang berbeda-beda. Misalnya letika progamer akan mejalankan program *python* pada perangkat yang mneggunakan Linux maupun Windows, program akan lebih mudah dijalankan meskipun tidak adanya modifikasi.

2.6 Kewajiban Menjaga Amanah

Amanah adalah semua tugas atau pembebanan agama yang meliputi perkara dunia dan akhirat yang ditujukan kepada manusia (Ibnu Katsir, 2013). Menjalankan amanah menjadi suatu hal yang penting bagi setiap individu, karena salah satu hikmah dari menjalankan amanah adalah seseorang dapat lebih dipercaya oleh orang lain. Hal ini merupakan modal yang sangat berharga ketika seseorang berinteraksi dengan orang lain. Kebalikannya ktika seseorang tidak menjalankan amanah dengan baik, maka orang lain akan enggan untuk bersimpati karena sudah tidak percaya lagi dengan orang tersebut seperti yang telah dijelaskan pada Al-

Qur’an juga dijelaskan tentang pentingnya menjada amanah, seperti yang di Firmankan oleh Allah SWT. pada QS: Al-Anfaal ayat 27:

يَٰٓأَيُّهَا ٱلَّذِينَ ءَامَنُواْ لَا تَخُونُواْ ٱللَّهَ وَٱلرَّسُولَ وَتَخُونُوٓاْ أَمَٰنَٰتِكُمۡ وَأَنتُمۡ تَعۡلَمُونَ

Artinya: “Hai orang-orang yang beriman, janganlah kamu mengkhianati Allah dan Rasul (Muhammad) dan (juga) janganlah kamu mengkhianati amanat-amanat yang dipercayakan kepadamu, sedang kamu mengetahui.”

Pada ayat tersebut dijelaskan bahwa kita sebagai muslim harus senantiasa menjaga amanah yang diberikan kepada kita. Hal ini sudah dijelaskan oleh Ibnu Katsir dalam bukunya yang berjudul Tafsil Ibnu Katsir. Dalam kitabnya juga dijelaskan bahwa ayat tersebut berisi peringatan dari Allah kepada orang-orang beriman agar tidak mengkhianati Allah dan Rasul-Nya. Dijelaskan juga pada ayat ini akan ada konsekuensi bagi mereka yang mengkhianati amanah tersebut. Adapun beberapa konsekuensi dari mengkhianati suatu amanah adalah hilangnya keberkahan, hilangnya kepercayaan, dosa besar serta runtuhnya moralitas dan integritas (Abdullah, 2005).

Kitab Tafsir Jalalain menjelaskan Surah Al-Anfal ayat 27 bahwa mengkhianati amanah berarti mengkhianati tanggung jawab yang telah diberikan oleh Allah, baik dalam hal kepercayaan, atau tugas-tugas tertentu yang dipercayakan kepada kita. Termasuk mengkhianati Rasul-Nya dengan mengabaikan Sunnah dan petunjuk beliau karena hal tersebut adalah amanah yang telah diberikan oleh Allah SWT. untuk menguji keimanan dan ketakwaan hambanya (Jalaluddin, 2008).

Pada Tafsir As-Sa’di dijelaskan bahwa berkhianat adalah Tindakan tidak setia dan melanggar janji atau kepercayaan yang telah diberikan orang lain. Perilaku khianat sering dikaitkan dengan sifat orang-orang munafik yang tampal seolah-olah

mereka beriman dan setia, tetapi dalam hati mereka menyimpan niat buruk. Tindakan khianat bukan hanya merusak hubungan antar manusia, tetapi juga hubungan dengan Allah SWT, karena pengkhianatan mencerminkan ketidakjujuran dan pengabaian terhadap amanah yang telah diberikan oleh-Nya (Syaikh Abdurrahman, 2015).

BAB III

METODE PENELITIAN

3.1 Jenis Penelitian

Jenis peneliian yang digunakan dalam penelitian ini adalah kajian kepustakaan dengan mengkaji dan menelaah beberapa buku, jurnal, karya tulis baik yang sudah dipublikasikan maupun yang belum serta beberapa referensi yang berkaitan dengan topik penelitian. Kajian kepustakaan harus meninjau semu permasalahan yang diteliti agar dapat mendukung pembahasan dan pemecahan masalah hingga tuntas (Kasiran, 2008).

3.2 Pra Penelitian

Pada penelitian kali ini, karena menggunakan metode penelitian kajian kepustakaan, maka sebelumnya akan dikaji terlebih dahulu teori-teori yang berkaitan dengan penelitian ini. Teori-teori yang sudah dikaji akan menjadi landasan untuk langkah-langkah penelitian berikutnya.

3.3 Tahapan Penelitian

Berikut adalah langkah-langkah yang dilakukan dalam penelitian ini agar tercapai tujuan yang diinginkan:

1. Melakukan proses penyandian menggunakan *atbash cipher* kunci geser

   Pada penelitian ini langkah-langkah yang dilakukan dalam proses enkripsi dan dekripsi, yaitu:

   a. Menyiapkan *plaintext* (*p*) yang akan disandikan

b. Menyiapkan bilangan bulat (*k*) sebagai kunci enkripsi
c. Mengubah *plaintext* (*p*) kedalam bilangan menurut tabel *atbash cipher*
d. Mengganti atau mensubtitusikan setiap bilangan (*p*) dengan urutan indeks bilangan yang sebaliknya
e. Menggeser huruf-huruf pada pesan sebanyak nilai kunci (*k*) dari bilangan (p_i) kedepan atau kebelakang
f. Mengubah kembali indeks bilangan (p_i) menjadi karakternya menurut tabel *atbash cipher* dan diperoleh *ciphertext* (*c*) pada proses enkripsi *atbash cipher* kunci geser
g. Menyiapkan kunci dekripsi berupa bilangan bulat (*k*) dari kunci enkripsi
h. Mengubah *ciphertext* (*c*) kedalam bentuk indeks bilangannya menurut tabel *atbash cipher*
i. Menggeser huruf-huruf pada pesan sebanyak nilai kunci (*k*) dari bilangan (p_i) kearah yang berlawanan pada proses enkripsi
j. Mengganti atau mensubtitusikan setiap bilangan *ciphertext* (*c*) dengan urutan bilangan sebenarnya
k. Mengubah kembali bilangan (c_i) menjadi karakternya menurut tabel *atbash cipher* dan diperoleh *plaintext* (*p*) pada proses dekripsi *atbash cipher* kunci geser

2. Melakukan proses penyandian menggunakan *rail fence cipher*

Pada penelitian ini langkah-langkah yang dilakukan dalam proses enkripsi dan dekripsi, yaitu:

a. Menentukan kunci (*k*) enkripsi

b. Membuat tabel dengan jumlah baris sesuai kunci (*k*)
c. Memasukkan *plaintext* (*p*) kedalam tabel secara zig-zag sesuai ketentuan *rail fence cipher*
d. Menyusun *plaintext* (*p*) dalam tabel secara horizontal perbaris dan diperoleh ciperteks (*c*) pada proses enkripsi *rail fence cipher*
e. Menentukan kunci dekripsi (*k*) dari kunci enkripsi
f. Membuat tabel dengan jumlah baris sesuai kunci (*k*)
g. Memasukkan *ciphertext* (*c*) kedalam tabel secara horizontal perbaris sesuai ketentuan *rail fence cipher*
h. Menyusun *ciphertext* (*c*) secara zig-zag dan diperoleh *plaintext* (*p*) pada proses dekripsi *rail fence cipher*

3. Melakukan proses penyandian menggunakan metode super enkripsi *atbash cipher* dan *rail fence cipher*
   a. Enkripsi menggunakan super enkripsi *atbash cipher* kunci geser dan *rail fence cipher*
      1) Menyiapkan *plaintext* (*p*) yang akan disandikan
      2) Menyiapkan bilangan bulat (*k*) sebagai kunci enkripsi
      3) Mengubah *plaintext* (*p*) kedalam bentuk bilangannya menurut tabel *atbash cipher*
      4) Mengganti atau mensubtitusikan setiap bilangan (*p*) dengan urutan kode bilangan yang sebaliknya
      5) Menggeser huruf-huruf pada pesan sebanyak nilai kunci (*k*) dari bilangan (p_i) kedepan atau kebelakang

6) Mengubah kembali bilangan (p_i) menjadi karakternya menurut tabel *atbash cipher* dan diperoleh *ciphertext* (*c*) pada proses enkripsi *atbash cipher kunci geser*
7) Menentukan kunci (*k*) enkripsi *rail fence cipher*
8) Membuat tabel dengan jumlah baris sesuai kunci (*k*)
9) Memasukkan *ciphertext* pertama (c_1) kedalam tabel secara zig-zag sesuai ketentuan *rail fence cipher*
10) Menyusun *ciphertext* pertama (c_1) dalam tabel secara horizontal perbaris dan diperoleh *ciphertext* terakhir (c_2) pada proses enkripsi *rail fence cipher*

b. Dekripsi menggunakan *atbash cipher* kunci geser dan *rail fence cipher*
1) Menyiapkan *ciphertext* (*c*) yang akan diuraikan
2) Menentukan kunci dekripsi (*k*) dari kunci enkripsi
3) Membuat tabel dengan jumlah baris sesuai kunci (*k*)
4) Memasukkan *ciphertext* (*c*) kedalam tabel secara horizontal perbaris sesuai ketentuan *rail fence cipher*
5) Menyusun *ciphertext* (*c*) secara zig-zag dan diperoleh *plaintext* pertama (p_1) pada proses dekripsi *rail fence cipher*
6) Mengubah *plaintext* pertama (p_1) kedalam bentuk bilangannya menurut tabel *atbash cipher*
7) Menggeser huruf-huruf pada pesan sebanyak nilai kunci (*k*) dari bilangan (p_{1i}) kearah yang berlawanan pada proses enkripsi
8) Mengganti atau mensubtitusikan setiap bilangan *ciphertext* (*c*) dengan urutan kode bilangan sebenarnya

9) Mengubah kembali bilangan (p_{1i}) menjadi karakternya menurut tabel *atbash cipher* dan diperoleh *plaintext* awal (*p*) pada proses dekripsi

2. Membuat dan merancang program metode super enkripsi *atbash cipher* dan *rail fence cipher* menggunakan *python*
3. Melakukan enkripsi dan dekripsi menggunakan program *Python* yang telah dibuat.

BAB IV

PEMBAHASAN

Pada bab ini penulis membahas tentang bagaimana proses enkripsi dan dekripsi dari super enkripsi menggunakan algoritma *atbash cipher* dan *rail fence cipher* serta implementasinya pada *python*.

4.1 Proses Penyandian Menggunakan Super Enkripsi *Atbash Cipher* dan *Rail Fence Cipher*

Berikut adalah proses penyandian menggunakan Super Enkripsi *atbash cipher* kunci geser dan *rail fence cipher*

4.1.1 Proses Penyandian Metode *Atbash Cipher* Kunci Geser

Proses penyandian menggunakan metode *atbash cipher* kunci geser merupakan proses memodifikasi algoritma dari *atbash cipher* dengan menambahkan kunci saat melakukan penyandiannya. Pada algoritma *atbash cipher*, proses penyandiannya tidak menggunakan kunci. Sedangkan pada *atbash cipher* kunci geser ini penyandiannya menggunakan kunci yang nantinya digunakan untuk menggeser karakter pada pesan baik kedepan maupun kebelakang.

Pada proses penyandian menggunakan metode *atbash cipher* kunci geser dibutuhkan konversi karakter kedalam angka, maka berikut ini adalah tabel hasil konversi untuk proses enkripsi dan dekripsi.

Tabel 4.1 *Konversi Atbash Cipher*

A	=	1	O	=	15	c	=	29	q	=	43
B	=	2	P	=	16	d	=	30	r	=	44
C	=	3	Q	=	17	e	=	31	s	=	45
D	=	4	R	=	18	f	=	32	t	=	46
E	=	5	S	=	19	g	=	33	u	=	47
F	=	6	T	=	20	h	=	34	v	=	48
G	=	7	U	=	21	i	=	35	w	=	49
H	=	8	V	=	22	j	=	36	x	=	50
I	=	9	W	=	23	k	=	37	y	=	51
J	=	10	X	=	24	l	=	38	z	=	52
K	=	11	Y	=	25	m	=	39	spasi	=	53
L	=	12	Z	=	26	n	=	40			
M	=	13	a	=	27	o	=	41			
N	=	14	b	=	28	p	=	42			

Berikut ini adalah contoh penerapan algoritma pada proses enkripsi dan dekripsi menggunakan *atbash cipher* kunci geser:

Misalkan pesan awal atau *plaintext*: Kota Malang

Kunci enkripsi : 3

1. Proses enkripsi *atbash cipher* kunci geser

a. Ubah *plaintext* (p) = **Kota Malang** menjadi bilangan menurut tabel konversi *atbash cipher 4.1*

K = 11	M = 13
o = 41	a = 27
t = 46	l = 38
a = 27	a = 27
‘space’ = 53	n = 40
	g = 33

b. Merubah setiap bilangan ke urutan bilangan yang sebaliknya menurut tabel konversi *atbash cipher*, dapat juga dihitung menggunakan rumus

$$E_{1(p)} = (-p \; mod \; m) + 1$$

Dengan m adalah jumlah karakter yang digunakan, pada algoritma *atbash cipher* kunci geser ini yang digunakan sebanyak 53 karakter. Maka berikut perhitungannya:

$$E_{1(11)} = (-11 \; mod \; 53) + 1 = 43$$

$$E_{1(41)} = (-41 \; mod \; 53) + 1 = 13$$

$$E_{1(46)} = (-46 \; mod \; 53) + 1 = 8$$

$$E_{1(27)} = (-27 \; mod \; 53) + 1 = 27$$

$$E_{1(53)} = (-53 \; mod \; 53) + 1 = 1$$

$$E_{1(13)} = (-13 \; mod \; 53) + 1 = 41$$

$$E_{1(27)} = (-27 \; mod \; 53) + 1 = 27$$

$$E_{1(38)} = (-38 \; mod \; 53) + 1 = 16$$

$$E_{1(27)} = (-27 \; mod \; 53) + 1 = 27$$

$$E_{1(40)} = (-40 \; mod \; 53) + 1 = 14$$

$$E_{1(33)} = (-33 \; mod \; 53) + 1 = 21$$

c. Menggeser huruf-huruf pada pesan sebanyak nilai kunci (*k*) dari setiap bilangan (E_1) kedepan atau kebelakang, dapat juga dihitung menggunakan rumus

$$E_{2(E_1)} = (E_1 - k) \; mod \; m$$

Dengan m adalah jumlah karakter yang digunakan, pada algoritma *atbash cipher* kunci geser ini yang digunakan sebanyak 53 karakter dan kunci yang digunakan adalah 3. Maka berikut perhitungannya:

$$E_{2(43)} = (43 - 3)\ mod\ 53 = 40$$

$$E_{2(13)} = (13 - 3)\ mod\ 53 = 10$$

$$E_{2(8)} = (8 - 3)\ mod\ 53 = 5$$

$$E_{2(27)} = (27 - 3)\ mod\ 53 = 24$$

$$E_{2(1)} = (1 - 3)\ mod\ 53 = 51$$

$$E_{2(41)} = (41 - 3)\ mod\ 53 = 38$$

$$E_{2(27)} = (27 - 3)\ mod\ 53 = 24$$

$$E_{2(16)} = (16 - 3)\ mod\ 53 = 13$$

$$E_{2(27)} = (27 - 3)\ mod\ 53 = 24$$

$$E_{2(14)} = (14 - 3)\ mod\ 53 = 11$$

$$E_{2(21)} = (21 - 3)\ mod\ 53 = 18$$

d. Mengubah kembali bilangan hasil penyandian *atbash cipher* kunci geser (E_2) menjadi karakternya menurut tabel konversi *atbash cipher 4.1*

40 = n	24 = X
10 = J	13 = M
5 = E	24 = X
24 = X	11 = K
51 = y	18 = R
38 = l	

Sehingga didapatkan *ciphertext* (c) = **nJEXylXMXKR**

2. Proses dekripsi *atbash cipher* kunci geser

a. Teks yang sebelumnya sudah dienkripsikan akan kita dekripsikan, *ciphertext* (c) sebelumnya adalah **nJEXylXMXKR**

dengan kunci yang sama dengan enkripsi adalah 3

b. Ubah *ciphertext* (c) = **nJEXylXMXKR** menjadi bilangan menurut tabel konversi *atbash cipher 4.1*

n = 40	X = 24
J = 10	M = 13
E = 5	X = 24
X = 24	K = 11
y = 51	R = 18
l = 38	

c. Menggeser huruf-huruf pada pesan sebanyak nilai kunci (*k*) dari bilangan (*c*) kearah yang berlawanan dengan proses enkripsi, dapat juga dihitung menggunakan rumus

$$D_{1(c)} = (c + k)\ mod\ m$$

Dengan m adalah jumlah karakter yang digunakan, pada algoritma *atbash cipher* kunci geser ini yang digunakan sebanyak 53 karakter dan kunci yang digunakan adalah 3. Maka berikut perhitungannya:

$$D_{1(40)} = (40 + 3)\ mod\ 53 = 43$$

$$D_{1(10)} = (10 + 3)\ mod\ 53 = 13$$

$$D_{1(5)} = (5 + 3)\ mod\ 53 = 8$$

$$D_{1(24)} = (24 + 3)\ mod\ 53 = 27$$

$$D_{1(51)} = (51 + 3)\ mod\ 53 = 1$$

$$D_{1(38)} = (38 + 3)\ mod\ 53 = 41$$

$$D_{1(24)} = (24 + 3)\ mod\ 53 = 27$$

$$D_{1(13)} = (13 + 3)\ mod\ 53 = 16$$

$$D_{1(24)} = (24 + 3)\ mod\ 53 = 27$$

$$D_{1(11)} = (11 + 3)\ mod\ 53 = 14$$

$$D_{1(18)} = (18 + 3)\ mod\ 53 = 21$$

d. Merubah setiap bilangan hasil dekripsi pertama (D_1) ke urutan bilangan yang sebaliknya menurut tabel konversi *atbash cipher*, dapat juga dihitung menggunakan rumus

$$D_{2(D_1)} = (-D_1\ mod\ m) + 1$$

Dengan m adalah jumlah karakter yang digunakan, pada algoritma *atbash cipher* kunci geser ini yang digunakan sebanyak 53 karakter. Maka berikut perhitungannya:

$$D_{2(43)} = (-43\ mod\ 53) + 1 = 11$$

$$D_{2(13)} = (-13\ mod\ 53) + 1 = 41$$

$$D_{2(8)} = (-8\ mod\ 53) + 1 = 46$$

$$D_{2(27)} = (-27\ mod\ 53) + 1 = 27$$

$$D_{2(1)} = (-1\ mod\ 53) + 1 = 53$$

$$D_{2(41)} = (-41\ mod\ 53) + 1 = 13$$

$$D_{2(27)} = (-27\ mod\ 53) + 1 = 27$$

$$D_{2(16)} = (-16\ mod\ 53) + 1 = 38$$

$$D_{2(27)} = (-27\ mod\ 53) + 1 = 27$$

$$D_{2(14)} = (-14\ mod\ 53) + 1 = 40$$

$$D_{2(21)} = (-21\ mod\ 53) + 1 = 33$$

e. Ubah kembali setiap bilangan hasil dekripsi (D_2) menjadi karakter berdasarkan tabel konversi *atbash cipher 4.1*.

11 = K	13 = M
41 = o	27 = a
46 = t	38 = l
27 = a	27 = a
53 = ‘spasi’	40 = n
	33 = g

Sehingga didapatkan kembali pesan teks awal atau *plaintext* (*p*) adalah **Kota Malang**

4.1.2 Proses Penyandian Metode *Rail Fence Cipher*

Proses penyandian metode *rail fence cipher* yaitu penyandian dengan cara mengacak urutan huruf-huruf pada pesan secara ‘zig-zag’, sesuai dengan ukuran baris berdasarkan kunci yang telah ditetapkan. Pada penyandian ini *ciphertext* diperoleh dengan cara membaca susunan karakter secara horizontal.

Berikut ini adalah contoh penerapan algoritma pada proses enkripsi dan dekripsi menggunakan *rail fence cipher*:

Misalkan pesan awal atau *plaintext*: UIN Malang

Kunci enkripsi : 3

1. Proses enkripsi *rail fence cipher*

a. Membuat tabel dengan jumlah baris sesuai dengan kunci dan jumlah kolom menyesuaikan banyaknya karakter yang ada pada pesan. Pada

contoh kali ini menggunakan kunci 3, sehingga akan dibuat tabel dengan bari sebanyak 3 dan kolom sebanyak karakter pada pesan, yaitu 10 kolom.

b. Buat pola zig-zag pada tabel agar memudahkan dalam penyusunan pesan teks. Pola zig-zag dibuat dari pojok kiri atas lalu turun secara diagonal, kemudian kembali naik secara diagonal lagi hingga seluruh kolom terisi.

Tabel 4.2 Pola pada *rail fence cipher*

•				•				•	
	•		•		•		•		•
		•				•			

c. Susun *plaintext* secara zig-zag, berurutan dari atas kebawah dan keatas lagi sesuai dengan pola yang telah dibuat sebelumnya.

Tabel 4.3 Urutan memasukkan karakter pada pola

• 1				• 5				• 9	
	• 2		• 4		• 6		• 8		• 10
		• 3				• 7			

Tabel 4.4 Hasil penyusunan pesan sesuai urutan pola

U				M				n	
	I		‘spasi’		a		a		g
		N				l			

d. Baca susunan pesan pada tabel secara horizontal tiap barisnya

Tabel 4.5 Membaca pesan perbaris

U				M				n	
	I		‘spasi’		a		a		g
		N				l			

Maka didapatkan *ciphertext* adalah **UMnI aagNl**

2. Proses dekripsi *rail fence cipher*

a. Teks yang sebelumnya sudah dienkripsikan akan kita dekripsikan, *ciphertext* sebelumnya adalah **UMnI aagNl**

dengan kunci yang sana dengan proses enkripsi yaitu 3

b. Membuat tabel dengan bayak baris sesuai dengan kunci dan banyak kolom sebanyak jumlah karakter pada pesan.

c. Buat pola zig-zag pada tabel agar memudahkan dalam penyusunan pesan teks. Pola zig-zag dibuat sama dengan pola pada proses enkripsi.

Tabel 4.6 Pola pada *rail fence cipher*

•				•				•	
	•		•		•		•		•
		•				•			

d. Menysusun teks yang sudah dienkripsikan sebelumya kedalam pola dengan urutan penyusunan dimulai dari baris paling atas kemudian dilanjutkan ke baris berikutnya hingga semua pola pada setiap baris terisi

Tabel 4.7 Urutan memasukkan karakter pada pola

• 1				• 5				• 9	
	• 2		• 4		• 6		• 8		•10
		• 3				• 7			

Setiap karakter dari pesan teks dimasukkan sesuai dengan urutannya.

Tabel 4.8 Hasil penyusunan pesan sesuai urutan pola

U				M				n	
	I		‘spasi’		a		a		g
		N				l			

e. Baca susunan pesan pada tabel secara zig-zag

Tabel 4.9 Membaca pesan secara zig-zag

U				M				n	
	I		‘spasi’		a		a		g
		N				l			

Maka didapatkan *plaintext* adalah **UIN Malang**

4.1.3 Proses Penyandian Metode Super Enkripsi *Atbash Cipher* Kunci Geser dan *Rail Fence Cipher*

Pada proses enkrinsi dan dekripsi menggunakan super enkripsi *atbash cipher* kunci geser dan *rail fence cipher*, penyandian pesan akan dilakukan dua kali yang nantinya bertujuan agar pesan yang disandikan akan lebih susah untuk diuraikan.

Berikut adalah proses enkripsi dan dekripsi pada penyandian menggunakan metode super enkripsi *atbash cipher* kunci geser dan *rail fence cipher*:

Misalkan pesan awal atau *plaintext*: **Matematika UIN Malang**

Kunci enkripsi : **5**

1. Proses enkripsi metode super enkripsi *atbash cipher* kunci geser dan *rail fence cipher*

   a. Ubah *plaintext* (p) = **Matematika UIN Malang** menjadi bilangan menurut tabel konversi *atbash cipher 4.1*

M = 13	'spasi' = 53
a = 27	U = 21
t = 46	I = 9
e = 31	N = 14
m = 39	'spasi' = 53
a = 27	M = 13
t = 46	a = 27
i = 35	l = 38
k = 37	a = 27
a = 27	n = 40
g = 33	

   b. Merubah setiap bilangan ke urutan bilangan yang sebaliknya menurut tabel konversi *atbash cipher*, dapat juga dihitung menggunakan rumus

$$E_{1(p)} = (-p \; mod \; m) + 1$$

Dengan m adalah jumlah karakter yang digunakan, pada algoritma *atbash cipher* kunci geser ini yang digunakan sebanyak 53 karakter. Maka berikut perhitungannya:

$$E_{1(13)} = (-13 mod\ 53) + 1 = 41$$

$$E_{1(27)} = (-27\ mod\ 53) + 1 = 27$$

$$E_{1(46)} = (-46\ mod\ 53) + 1 = 8$$

$$E_{1(31)} = (-31\ mod\ 53) + 1 = 23$$

$$E_{1(39)} = (-39\ mod\ 53) + 1 = 15$$

$$E_{1(27)} = (-27\ mod\ 53) + 1 = 27$$

$$E_{1(46)} = (-46\ mod\ 53) + 1 = 8$$

$$E_{1(35)} = (-35\ mod\ 53) + 1 = 19$$

$$E_{1(37)} = (-37\ mod\ 53) + 1 = 17$$

$$E_{1(27)} = (-27\ mod\ 53) + 1 = 27$$

$$E_{1(53)} = (-53\ mod\ 53) + 1 = 1$$

$$E_{1(21)} = (-21\ mod\ 53) + 1 = 33$$

$$E_{1(9)} = (-9\ mod\ 53) + 1 = 45$$

$$E_{1(14)} = (-14\ mod\ 53) + 1 = 40$$

$$E_{1(53)} = (-53\ mod\ 53) + 1 = 1$$

$$E_{1(13)} = (-13\ mod\ 53) + 1 = 41$$

$$E_{1(27)} = (-27\ mod\ 53) + 1 = 27$$

$$E_{1(38)} = (-38\ mod\ 53) + 1 = 16$$

$$E_{1(27)} = (-27\ mod\ 53) + 1 = 27$$

$$E_{1(40)} = (-40\ mod\ 53) + 1 = 14$$

$$E_{1(33)} = (-33\ mod\ 53) + 1 = 21$$

c. Menggeser huruf-huruf pada pesan sebanyak nilai kunci (*k*) dari setiap bilangan (E_1) kedepan atau kebelakang, dapat juga dihitung menggunakan rumus

$$E_{2(E_1)} = (E_1 - k)\ mod\ m$$

Dengan m adalah jumlah karakter yang digunakan, pada algoritma *atbash cipher* kunci geser ini yang digunakan sebanyak 53 karakter dan kunci yang digunakan adalah 3. Maka berikut perhitungannya:

$$E_{2(41)} = (41 - 5)\ mod\ 53 = 36$$

$$E_{2(27)} = (27 - 5)\ mod\ 53 = 22$$

$$E_{2(8)} = (8 - 5)\ mod\ 53 = 3$$

$$E_{2(23)} = (23 - 5)\ mod\ 53 = 18$$

$$E_{2(15)} = (15 - 5)\ mod\ 53 = 10$$

$$E_{2(27)} = (27 - 5)\ mod\ 53 = 22$$

$$E_{2(8)} = (8 - 5)\ mod\ 53 = 3$$

$$E_{2(19)} = (19 - 5)\ mod\ 53 = 14$$

$$E_{2(17)} = (17 - 5)\ mod\ 53 = 12$$

$$E_{2(27)} = (27 - 5)\ mod\ 53 = 22$$

$$E_{2(1)} = (1 - 5)\ mod\ 53 = 49$$

$$E_{2(33)} = (33 - 5)\ mod\ 53 = 28$$

$$E_{2(45)} = (45 - 5)\ mod\ 53 = 40$$

$$E_{2(40)} = (40 - 5)\ mod\ 53 = 35$$

$$E_{2(1)} = (1 - 5)\ mod\ 53 = 49$$

$$E_{2(41)} = (41 - 5)\ mod\ 53 = 36$$

$$E_{2(27)} = (27 - 5)\ mod\ 53 = 22$$

$$E_{2(16)} = (16 - 5)\ mod\ 53 = 11$$

$$E_{2(27)} = (27 - 5)\ mod\ 53 = 22$$

$$E_{2(14)} = (14 - 5)\ mod\ 53 = 9$$

$$E_{2(21)} = (21 - 5)\ mod\ 53 = 16$$

d. Mengubah kembali bilangan hasil penyandian *atbash cipher* kunci geser (E_2) menjadi karakternya menurut tabel konversi *atbash cipher 4.1*

36 = j	28 = b
22 = V	40 = n
3 = C	35 = i
18 = R	49 = w
10 = J	36 = j
22 = V	22 = V
3 = C	11 = K
14 = N	22 = V
12 = L	9 = I
22 = V	16 = P
49 = w	

e. Sehingga didapatkan hasil enkripsi pertama

ciphertext (c_1) = **jVCRJVCNLVwbniwjVKVIP**

f. Membuat tabel dengan jumlah baris sesuai dengan kunci dan jumlah kolom menyesuaikan banyaknya karakter yang ada pada pesan. Pada contoh kali ini menggunakan kunci 5, sehingga akan dibuat tabel dengan

bari sebanyak 5 dan kolom sebanyak karakter pada pesan, yaitu 21 kolom.

g. Buat pola zig-zag pada tabel agar memudahkan dalam penyusunan pesan teks. Pola zig-zag dibuat dari pojok kiri atas lalu turun secara diagonal, kemudian kembali naik secara diagonal lagi hingga seluruh kolom terisi.

Tabel 4.10 Pola pada *rail fence cipher*

•								•							•				
	•						•		•							•			
		•				•				•				•				•		
			•		•						•		•						•	
				•								•								•

h. Susun *plaintext* secara zig-zag, berurutan dari atas kebawah dan keatas lagi sesuai dengan pola yang telah dibuat sebelumnya.

Tabel 4.11 Urutan memasukkan karakter pada pola

• 1								• 9							• 17				
	• 2						• 8		• 10							• 16		• 18			
		• 3				• 7				• 11				• 15					• 19		
			• 4		• 6						• 12		• 14							• 20	
				• 5								• 13									• 21

Setiap karakter dari pesan teks dimasukkan sesuai dengan urutannya.

Tabel 4.12 Hasil penysusunan pesan sesuai urutan pola

j								**L**								**V**			
	V						**N**		**V**						**j**		**K**		
		C				**C**				**w**				**w**				**V**	
			R		**V**						**b**		**i**						**I**
				J								**n**							**P**

i. Baca susunan pesan pada tabel secara horizontal tiap barisnya.

Tabel 4.13 Membaca pesan perbaris

j								**L**								**V**			
	V						**N**		**V**						**j**		**K**		
		C				**C**				**w**				**w**				**V**	
			R		**V**						**b**		**i**						**I**
				J								**n**							**P**

Maka didapatkan hasil akhir dari penyandian yaitu,

ciphertext = **jLVVNVjKCCwwVRVbiIJnP**

2. Proses dekripsi metode super enkripsi *atbash cipher* kunci geserdan *rail fence cipher*

a. Teks yang sebelumnya sudah dienkripsikan akan kita dekripsikan, *ciphertext* sebelumnya adalah

jLVVNVjKCCwwVRVbiIJnP

dengan kunci yang sana dengan proses enkripsi yaitu 5

b. Membuat tabel dengan bayak baris sesuai dengan kunci dan banyak kolom sebanyak jumlah karakter pada pesan.

c. Buat pola zig-zag pada tabel agar memudahkan dalam penyusunan pesan teks. Pola zig-zag dibuat sama dengan pola pada proses enkripsi.

Tabel 4.14 Pola pada *rail fence cipher*

•								•							•				
	•						•		•					•		•			
		•				•				•			•				•		
			•		•						•		•					•	
				•								•							•

d. Menysusun teks yang sudah dienkripsikan sebelumya kedalam pola dengan urutan penyusunan dimulai dari baris paling atas kemudian dilanjutkan ke baris berikutnya hingga semua pola pada setiap baris terisi

Tabel 4.15 Urutan pola memasukkan karakter pada pola

•1								•2							•3				
	•4						•5		•6						•7		•8			
		•9				•10				•11				•12				•13		
			•14		•15						•16		•17						•18	
				•19								•20								•21

Setiap karakter dari pesan teks dimasukkan sesuai dengan urutannya.

Tabel 4.16 Hasil penysusunan pesan sesuai urutan pola

j								**L**								**V**			
	V						**N**		**V**						**j**		**K**			
		C				**C**				**w**				**w**				**V**		
			R		**V**						**b**		**i**						**I**	
				J								**n**								**P**

e. Baca susunan pesan pada tabel secara zig-zag

Tabel 4.17 Membaca pesan secara zig-zag

j								**L**								**V**			
	V						**N**		**V**						**j**		**K**			
		C				**C**				**w**				**w**				**V**		
			R		**V**						**b**		**i**						**I**	
				J								**n**								**P**

Maka didapatkan penguraian pesan pertama adalah

jVCRJVCNLVwbniwjVKVIP

f. Ubah hasil penguraian pesan pertama menjadi bilangan (D_1) menurut tabel konversi *atbash cipher 4.1*

j = 36	b = 28
V = 22	n = 40
C = 3	i = 35
R = 18	w = 49

J = 10	j = 36
V = 22	V = 22
C = 3	K = 11
N = 14	V = 22
L = 12	I = 9
V = 22	P = 16
w = 49	

g. Menggeser huruf-huruf pada pesan sebanyak nilai kunci (*k*) dari bilangan (D_1) kearah yang berlawanan dengan proses enkripsi, dapat juga dihitung menggunakan rumus

$$D_{2(D_1)} = (D_1 + k)\ mod\ m$$

Dengan m adalah jumlah karakter yang digunakan, pada algoritma *atbash cipher* kunci geser ini yang digunakan sebanyak 53 karakter dan kunci yang digunakan adalah 5. Maka berikut perhitungannya:

$$D_{2(36)} = (36 + 5)\ mod\ 53 = 41$$

$$D_{2(22)} = (22 + 5)\ mod\ 53 = 27$$

$$D_{2(3)} = (3 + 5)\ mod\ 53 = 8$$

$$D_{2(18)} = (18 + 5)\ mod\ 53 = 23$$

$$D_{2(10)} = (10 + 5)\ mod\ 53 = 15$$

$$D_{2(22)} = (22 + 5)\ mod\ 53 = 27$$

$$D_{2(3)} = (3 + 5)\ mod\ 53 = 8$$

$$D_{2(14)} = (14 + 5)\ mod\ 53 = 19$$

$$D_{2(12)} = (12 + 5)\ mod\ 53 = 17$$

$$D_{2(22)} = (22 + 5)\ mod\ 53 = 27$$

$$D_{2(49)} = (49 + 5)\ mod\ 53 = 1$$

$$D_{2(28)} = (28 + 5)\ mod\ 53 = 3$$

$$D_{2(40)} = (40 + 5)\ mod\ 53 = 45$$

$$D_{2(35)} = (35 + 5)\ mod\ 53 = 40$$

$$D_{2(49)} = (49 + 5)\ mod\ 53 = 1$$

$$D_{2(36)} = (36 + 5)\ mod\ 53 = 41$$

$$D_{2(22)} = (22 + 5)\ mod\ 53 = 27$$

$$D_{2(11)} = (11 + 5)\ mod\ 53 = 16$$

$$D_{2(22)} = (22 + 5)\ mod\ 53 = 27$$

$$D_{2(9)} = (9 + 5)\ mod\ 53 = 14$$

$$D_{2(16)} = (16 + 5)\ mod\ 53 = 21$$

h. Merubah setiap bilangan hasil dekripsi kedua (D_2) ke urutan bilangan yang sebaliknya menurut tabel konversi *atbash cipher*, dapat juga dihitung menggunakan rumus

$$P_{(D_2)} = (-D_2\ mod\ m) + 1$$

Dengan m adalah jumlah karakter yang digunakan, pada algoritma *atbash cipher* kunci geser ini yang digunakan sebanyak 53 karakter. Maka berikut perhitungannya:

$$P_{(41)} = (-41\ mod\ 53) + 1 = 13$$

$$P_{(27)} = (-27\ mod\ 53) + 1 = 27$$

$$P_{(8)} = (-8\ mod\ 53) + 1 = 46$$

$$P_{(23)} = (-23\ mod\ 53) + 1 = 31$$

$$P_{(15)} = (-15\ mod\ 53) + 1 = 39$$

$$P_{(27)} = (-27\ mod\ 53) + 1 = 27$$

$$P_{(8)} = (-8\ mod\ 53) + 1 = 46$$

$$P_{(19)} = (-19\ mod\ 53) + 1 = 35$$

$$P_{(17)} = (-17\ mod\ 53) + 1 = 37$$

$$P_{(27)} = (-27\ mod\ 53) + 1 = 27$$

$$P_{(1)} = (-1\ mod\ 53) + 1 = 53$$

$$P_{(33)} = (-33\ mod\ 53) + 1 = 21$$

$$P_{(45)} = (-45\ mod\ 53) + 1 = 9$$

$$P_{(40)} = (-40\ mod\ 53) + 1 = 14$$

$$P_{(1)} = (-1\ mod\ 53) + 1 = 53$$

$$P_{(41)} = (-41\ mod\ 53) + 1 = 13$$

$$P_{(27)} = (-27\ mod\ 53) + 1 = 27$$

$$P_{(16)} = (-16\ mod\ 53) + 1 = 38$$

$$P_{(27)} = (-27\ mod\ 53) + 1 = 27$$

$$P_{(14)} = (-14\ mod\ 53) + 1 = 40$$

$$P_{(21)} = (-21\ mod\ 53) + 1 = 33$$

i. Ubah kembali setiap bilangan hasil dekripsi (P) menjadi karakter berdasarkan tabel konversi *atbash cipher 4.1*.

13 =M	21 = U
27 = a	9 = I
46 = t	14 = N

31 = e	53 = ‘spasi’
39 = m	13 = M
27 = a	27 = a
46 = t	38 = l
35 = i	27 = a
37 = k	40 = n
27 = a	33 = g
53 = ‘spasi’	

Sehingga didapatkan kembali pesan teks awal atau *plaintext* (*p*) adalah **Matematika UIN Malang**

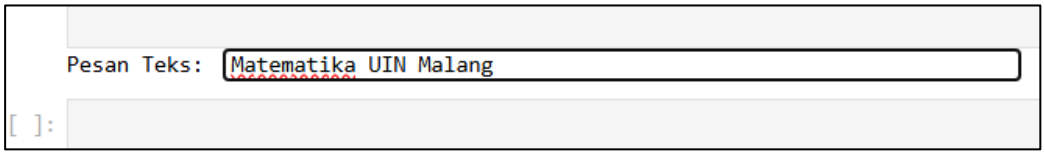
4.2. Implementasi pada *Pyhton*

Penyandian dengan menggunakan super enkripsi *atbash cipher* kunci geser dan *rail fence cipher* penyandiannya rumit karena menyandikan pesan menggunakan dua metode penyandian. Agar penyandian pesan bisa lebih cepat dan akurat dapat menggunakan program *python*. Berikut langkah-langkah pengujian mengenkripsi super enkripsi *atbash cipher* kunci geser dan *rail fence cipher* menggunakan program *python*:

1. Buka Jupyter Notebook dan buka file program untuk mengenkripsi pesan
2. Setelah itu *run* program untuk mengenkripsi pesan
3. Masukkan pesan yang akan di enkripsikan

   Misalkan pesan yang akan dienkripsikan adalah “Matematika UIN Malang”

Gambar 4.1 Input pesan teks awal



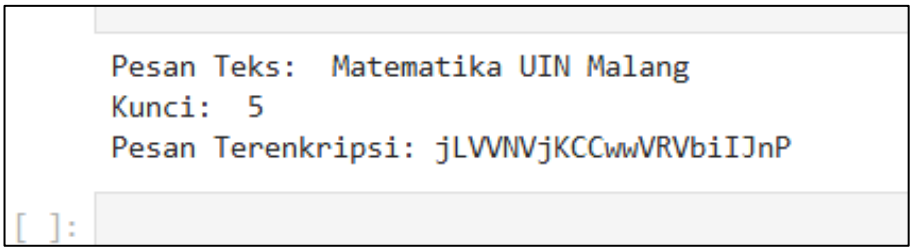
4. Masukkan kunci enkripsi

   Misalkan kunci yang digunakan adalah 5

Gambar 4.2 Input kunci enkripsi

5. Setelah itu akan muncul hasil dari enkripsi pesan

Gambar 4.3 Hasil enkripsi program *python*



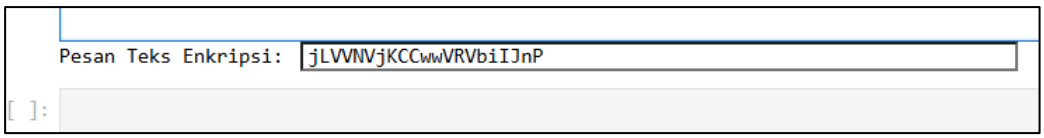
6. Sehinnga hasil akhir dari enkripsi menggunakan metode super enkripsi *atbash cipher* kunci geser dan *rail fence cipher* adalah "**jLVVNVjKCCwwVRVbiIJnP**"

Berikut langkah-langkah pengujian mendekripsikan super enkripsi *atbash cipher* kunci geser dan *rail fence cipher* menggunakan program *python*:

1. Buka program untuk mendekripsi pesan
2. Masukkan hasil enkripsi pesan

Hasil enkripsi pesan sebelumnya adalah "**jLVVNVjKCCwwVRVbiIJnP**"

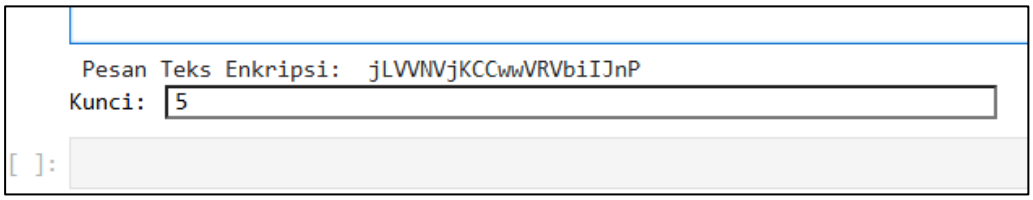
Gambar 4.4 Input teks hasil enkripsi



3. Masukkan kunci yang digunakan pada proses enkripsi

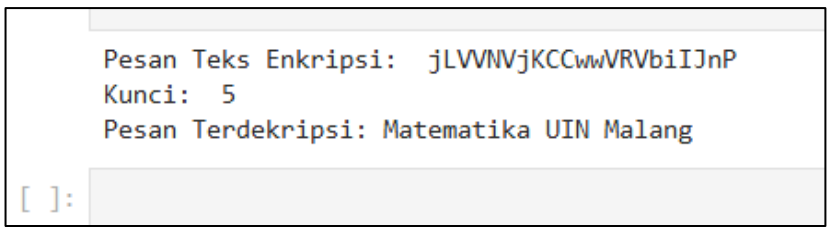
   Kunci yang digunakan adalah 5

Gambar 4.5 Input kunci dekripsi



4. Setelah itu akan muncuk hasil dari dekripsi pesan yang sudah terenkripsi

Gambar 4.6 Hasil dekripsi program *pyhton*



5. Sehinnga hasil akhir dari dekripsi menggunakan metode super enkripsi *atbash cipher* kunci geser dan *rail fence cipher* adalah "**Matematika UIN Malang**"

Setelah dijelaskan terkait langkah-langkah penggunaan dan hasil dari program *python*, berikut ini adalah beberapa contoh pesan teks yang diujikan pada program *python*:

Tabel 4.18 Hasil uji coba pesan teks pada program *python*

No	Pesan Awal	Kunci	Hasil Enkripsi	Hasil Dekripsi
1	Jurusan Matematika UIN Malang	3	oFlLNpXRDDXyXTX PXdklMKGKEEyyX	Jurusan Matematika UIN Malang
2	Sarah pergi kuliah menuju jurusan Matematika di UIN Malang pada hari Senin Pukul 07.00 WIB	5	dEVBICbIONwnVRP NOMwVwVNwnVPw VIIK0ZuEGNKwBM DjJLNiKwVERwB7w VwwBJIBBVRVSwV GSNdgL.0OLRECwj VwB0	Sarah pergi kuliah menuju jurusan Matematika di UIN Malang pada hari Senin Pukul 07.00 WIB
3	Islam menekankan bahwa setiap muslim harus menjaga dan menunaikan amanah yang diberikan kepada mereka. Yang berarti menyampaiakan apa yang dipercayakan kepada kita kepada orang yang berhak menerima. Pentingnya menjaga amanah juga	13	fJHoJNLKJNNGxDF KFowwvNNGqBxqBx NNNAxHoMAJNNoF AvKoJvKF CoFEoNN oJAzJqzoJFNNBHDx vwCqoNGAoNHCNN NAKAANDJNzNJ BB EBJuoBAJNNGDNH NDNJBszJNNHKNBF FDDxoNANBAA Nw MoNoANvoCGoGoB ABoN.oNKNoND.JA	Islam menekankan bahwa setiap muslim harus menjaga dan menunaikan amanah yang diberikan kepada mereka. Yang berarti menyampaiakan apa yang dipercayakan kepada kita kepada orang yang berhak menerima. Pentingnya menjaga amanah juga

	untuk membangun kepercayaan dan keharmonisan hubungan antar sesama manusia. Sebaliknya ketika kita sudah tidak menjalankan amanah dengan baik, maka kita tidak akan dipercaya lagi oleh orang lain.		ooNAoNJDKBNAFN ooBNBooNKzoFDoA NANoBNDDKFABVJ uJBoDDN JMuwBoFJ PvNHovHGYoGuJow NNovwANMoNqxAo wuJAMDNxAADFoxJ NovzANHo.FoEoNN ANN.JACxANJoANo NJDqJAqEAJNAABN DNNAFDoLAADNF NuDxAHxNqzoNMv AuuxNouNFNvCNDN KxHFNDBGAFFNoJz oNNAoFHHoLqGMA woFND,BFJoNAoND MNKHANNuuDAozC	untuk membangun kepercayaan dan keharmonisan hubungan antar sesama manusia. Sebaliknya ketika kita sudah tidak menjalankan amanah dengan baik, maka kita tidak akan dipercaya lagi oleh orang lain.

Berdasarkan tabel hasil uji coba diatas, dapat diketahui bahwa program *python* tersebut tidak hanya bisa menyandikan hanya pesan singkat namun juga dapat menyandikan sebuah paragraf. Namun ketika ada karakter selain huruf kapital, huruf kecil dan ‘spasi’ maka tidak akan diubah karena pada penelitian ini karakter yang digunakan masih terbatas.

4.3 Penyandian Menggunakan Metode Super Enkripsi *Atbash Cipher* dan *Rail Fence Cipher* dalam Pandangan Islam

Perkembangan teknologi adalah salah satu aspek dalam kehidupan manusia yang terus berkembang secara dinamis. Dengan berkembanganya tekonologi maka banyak hal yang dapat dilakukan dengan mudah termasuk dalam hal mengirimkan pesan. Namun, karena mudahnya mengirimkan pesan juga membuat pesan tersebut lebih mudah untuk dimanfaatkan oleh pihak-pihak yang tidak bertanggung jawab. Oleh karena itu perlu adanya penyandian yang dilakukan untuk melindungi kerahasiaan pesan yang akan dikirim. Salah satu cara untuk menyandikan pesan adalah dengan menggunakan kriptografi agar nantinya pesan yang dikirimkan tidak dimanfaatkan oleh orang yang tidak bertanggugn jawab.

Melindungi kerahasiaan pesan pesan termasuk kedalam amanah yang harus dijaga oleh setiap muslim dan disampaikan kepada orang yang berhak menerima pesan tersebut. seperti yang sudah dijelaskan dalam Al-Qur'an pada Surah An-Nisa' ayat 58:

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

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

Ayat diatas menekankan pentingnya menyampaikan amanah kepada yang berhak menerima. Sehingga untuk menghindari suatu pesan disalahgunakan oleh orang yang tidak bertanggung jawab maka dilakukan penyandian terhadap pesan tersebut. Agar penyandian pesan bisa lebih aman maka dilakukan penyandian

dengan mengombinasikan dua metode penyandian atau biasa disebut dengan super enkripsi. Pada penelitian ini metode penyandian yang digunakan adalah *atbash cipher* kunci geser dan *rail fence cipher*.

Islam menekankan bahwa setiap muslim harus menjaga dan menunaikan amanah yang diberikan kepada mereka. Yang berarti menyampaiakan apa yang dipercayakan kepada kita kepada orang yang berhak menerima. Pentingnya menjaga amanah juga untuk membangun kepercayaan dan keharmonisan hubungan antar sesama manusia. Sebaliknya ketika kita sudah tidak menjalankan amanah dengan baik, maka kita tidak akan dipercaya lagi oleh orang lain. Seperti yang dijelaskan pada hadits yang riwayatkan oleh Abu Dawud dan Tirmidzi yang berbunyi:

رَسُولَ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ يَقُولُ أَدِّ الْأَمَانَةَ إِلَى مَنْ ائْتَمَنَكَ وَلَا تَخُنْ مَنْ خَانَكَ

"Rasulullah Shallallahu'alaihiwasallam bersabda: "Tunaikanlah amanat kepada orang yang memberimu amanat, dan janganlah kamu berkhianat kepada orang yang telah menghianati dirimu." (HR. Abu Dawud dan Tirmidzi)

Pada hadits tersebut juga dijelaskan bahwa ketika ada orang yang tidak amanah lagi atau berkhianat maka kita tidak boleh membalasnya dengan berkhianat juga. Karena Islam tidak pernah mengajarkan membalas suatu hal yang buruk dengan keburukan juga karena hal tersebut tidak akan menyelesaikan masalah, justru akan memperumit masalah tersebut. Oleh karena itu Islam mendorong umatnya untuk selalu mengutamakan kebaikan dan berusaha menjalin hubungan baik dengan sesama manusia sehingga dapat memunculkan rasa aman dan tentram pada setiap pribadi manusia.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan pembahasan sebelumnya, maka dapat diambil kesimpulan bahwa proses enkripsi pesan teks dengan super enkripsi *atbash cipher* kunci geser dan *rail fence cipher* adalah dengan membuat tabel atbash cipher dengan cara membalik urutan karakternya. Kemudian menggesernya sebanyak kunci yang berupa bilangan bulat kedepan atau kebelakang. Kemudian dilanjutkan dengan mentrasnformasi pesan secara zig-zag, sehingga diperoleh pesan teks hasil penyandian. Pada proses dekripsi dengan super enkripsi *atbash cipher* kunci geser dan *rail fence cipher* berkebalikan dengan proses enkripsinya. Pada proses ini dilakukan dilakukan transformasi pesan teks secara zig-zag. Lalu dilanjutkan dengan menggeser urutan teks berkebalikan dengan proses enkripsi dan mensubtitusikan karakter sesuai tabel atbash cipher, sehingga didapatkan hasil pesan teks yang sama dengan pesan teks yang asli.

Proses penyandian pesan teks dengan super enkripsi *atbash cipher* kunci geser dan *rail fence cipher* menggunakan program python dapat membantu proses penyandian menjadi lebih cepat dan efektif. Dengan menginput pesan teks dan kunci maka hasil enkripsi langsung ditampakkan. Begitu juga denga proses dekripsi, dengan memasukkan pesan yang terenkripsi dan kunci yang sama pada proses enkripsi maka hasil dari pesan teks awal langsung ditampakkan.

5.2 Saran

Penelitian ini membahas tentang penyandian menggunakan metode super enkripsi *atbash cipher* kunci geser dan *rail fence cipher*. Pada penelitian selanjutnya disarankan untuk membuat suatu modifikasi algoritma kriptografi dari metode tersebut misalnya dengan menambahkah karakter baru atau menggunakan karakter yang ada di tabel ASCII tidak terbatas hanya karakter huruf saja atau dengan menambahkan proses trasnformasi atau subtitusinya menjadi lebih rumit lagi. Agar nantinya pesan teks yang disandikan bisa lebih rumit sehingga tidak mudah diuraiakan dan kemanan pesan teks tersebut menjadi meningkat.

DAFTAR PUSTAKA

Al-Quran Kementrian Agama RI. 2015. *Alquran dan Terjemahannya*. Jakarta: Lajnah Pentashihan Mushad Al-Quran

A. Sadikin, Rifki. 2012. *Kriptografi untuk Keamanan Jaringan*. Yogyakarta.

Abdullah, M. 2007. *Tafsir Ibnu Katsir Jilid 4*. Bogor: Pustaka Imam Asy-Syafi'i

Abdullah, M. 2007. *Tafsir Ibnu Katsir Jilid 5*. Bogor: Pustaka Imam Asy-Syafi'i

Abdurrahman bin Nashir as-Sa'di. Syaikh. 2015. *Tafsir Al-Qur'an Jilid 3*, Jakarta: Darul Haq

Akbar, Haryus Aminul. 2010. *Analisis Serangan Dictionary attack pada Ciphertext Berbasis Substitusi Monoalfabetik*. Bandung: ITB.

Al-Mahalli, Jalaluddin dan Jalaluddin As-Suyuti, *Tafsir Al-Jalalain, diterjemahkan Bahrun Abubakar, Terjemahan tafsir Jalalain Berikut Asbabun Nuzul Jilid.* Bandung : Penerbit Sinar Baru Algensindo, 2008. Al-Quran dan terjemahannya, 2008. Bandung: Dipenegoro: Departemen Agama RI

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

Ariyus, D. 2008. *Pengantar Ilmu Kriptografi*. Yogyakarta: C.V. Andi Offset.

Clinton, Rombang Mathew Raphael dan Rizal Sengkey. 2019. *Purwarupa Sistem Daftar Pelanggaran Lalulintas Berbasis Mini-Komputer Raspberry pi*. Teknik Elektro, Universitas Sam Ratulangi Manado.

Kasiran, Moh. 2008. *Metodologi Penelitian*. Yogyakarta: UIN Maliki Press

Latifah, Renani. Ambo, Siti Nurbaya. Kurnia, Syafitri Indah. 2017. *Modifikasi Algoritma Caesar Cipher dan Rail Fence untuk Peningkatan Keamanan Teks*

Alfanumerik dan Karakter Khusus. Jakarta: Universitas Muhammadiyah Jakarta.

Lutz, M. 2013. *Learning Python 5th Edition*. Sebastopol: O'Reilly Media, Inc.

Meyer, C., Matyas, S. M., 1982, *Cryptography, Anew Dimension in Computer Data Security,* John Wiley & Son1

Paryati, 2008. *Keamanan Sistem Informasi. Seminar Nasional Informatika* 2008. 47(1)

Primartha, Rifkie. 2011. *Penerapan Enkripsi dan Dekripsi File Menggunakan Algoritma Data Encryption*. Jurnal Sistem Informasi Vol 3 No. 2 (371-387)

Purnama, Benni., Rohayani Hetty, AH. 2015. *New Modified Caesar Cipher Cryptography Method With Legible Ciphertext From A Message To Be Encrypted*. Procedia Computer Science.

Siahaan, A.P.U. 2016. *Rail Fence Cryptography in Securing Information. International Journal of Science & Engineering Research (IJSER)*. Vol 7, Issue 7.

Schuerer, Katja dan Corinne Maufrais. 2010. *Introduction to Progamming Using Python*.

Sutoyo, T., Mulyanto, E., Suhartono, V., Nurhayati, Oky D., dan Wijanarto. 2009. *Teori Pengolahan Citra Digital*. Yogyakarta: Andi Offset.

Sugiyanti, U., & Pambudi, A. 2022. *Perlindungan Data Privasi dan Kebebasan Informasi dalam Platform WhatsApp*. Jurnal IPI Vol.7 No. 2 (2022):60-70

LAMPIRAN

Lampiran 1. Program Enkripsi Python

```python
import string

def atbash_cipher(text):
    # Buat kamus penggantian huruf
    alphabet = string.ascii_uppercase + string.ascii_lowercase + ' '
    reversed_alphabet = alphabet[::-1]

    # Buat tabel translasi
    translation_table = str.maketrans(alphabet, reversed_alphabet)

    # Terapkan translasi pada teks
    encrypted_text = text.translate(translation_table)

    return encrypted_text

def kuncigeser(text, shift):
    result = ''
    alphabet = ' ' + ''.join(chr(122 - i) for i in range(26)) + ''.join(chr(90 - i) for i in range(26))

    for char in text:
        if char in alphabet:
            shifted_index = (alphabet.index(char) + shift) % 53
            result += alphabet[shifted_index]
        else:
            result += char  # Biarkan karakter non-alfabet tidak berubah
    return result

def encrypt_rail_fence(text, key):
    rail = [''] * key
    direction = False
    row = 0

    for char in text:
        rail[row] += char
        if row == 0 or row == key - 1:
            direction = not direction
        row += 1 if direction else -1

    return ''.join(rail)

# PENGGUNAAN
pesan = input("Pesan Teks: ")
key = int(input("Kunci: "))

pesan_terenkripsi_atbash = atbash_cipher(pesan)
pesan_terenkripsi_kuncigeser = kuncigeser(pesan_terenkripsi_atbash, key)
ciphertext = encrypt_rail_fence(pesan_terenkripsi_kuncigeser, key)
print("Pesan Terenkripsi:", ciphertext)
```

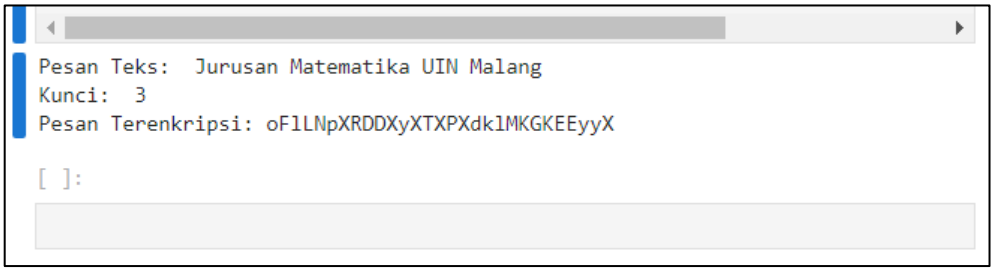
Lampiran 2. Program Dekripsi Python

```
import string

def rail_fence_decrypt(text, rails):
    fence = [[] for _ in range(rails)]
    rail = 0
    direction = 1

    for char in text:
        fence[rail].append(None)
        rail += direction

        if rail == rails - 1 or rail == 0:
            direction *= -1

    index = 0
    for rail in range(rails):
        for i in range(len(fence[rail])):
            fence[rail][i] = text[index]
            index += 1
    rail = 0
    direction = 1
    decrypted_text = ''
    for _ in range(len(text)):
        decrypted_text += fence[rail].pop(0)
        rail += direction

        if rail == rails - 1 or rail == 0:
            direction *= -1

    return decrypted_text

def decrypt_kuncigeser(ciphertext, shift):
    result = ''
    alphabet = ' ' + ''.join(chr(122 - i) for i in range(26)) + ''.join(chr(90 - i) for i in range(26))

    for char in ciphertext:
        if char in alphabet:
            shifted_index = (alphabet.index(char) - shift) % 53
            result += alphabet[shifted_index]
        else:
            result += char  # Biarkan karakter non-alfabet tidak berubah
    return result
def decrypt_atbash(text):
    alphabet = string.ascii_uppercase + string.ascii_lowercase + ' '
    reversed_alphabet = alphabet[::-1]

    translation_table = str.maketrans(reversed_alphabet, alphabet)

    decrypted_text = text.translate(translation_table)

    return decrypted_text

# PENGGUNAAN
pesan = input("Pesan Teks Enkripsi: ")
key = int(input("Kunci: "))

pesan_terdekripsi_rail_fence = rail_fence_decrypt(pesan, key)
pesan_terdekripsi_kuncigeser = decrypt_kuncigeser(pesan_terdekripsi_rail_fence, key)
pesan_terdekripsi_atbash = decrypt_atbash(pesan_terdekripsi_kuncigeser)
print("Pesan Terdekripsi:", pesan_terdekripsi_atbash)
```

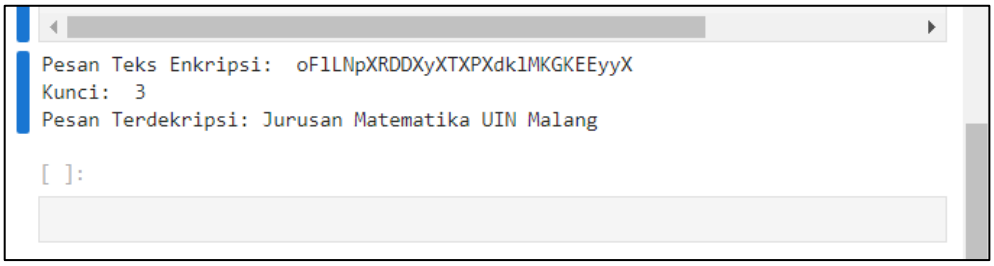
Lampiran 3. Hasil Uji Coba Pada Program *Pyhton*

Contoh 1

Hasil Enkripsi Contoh 1

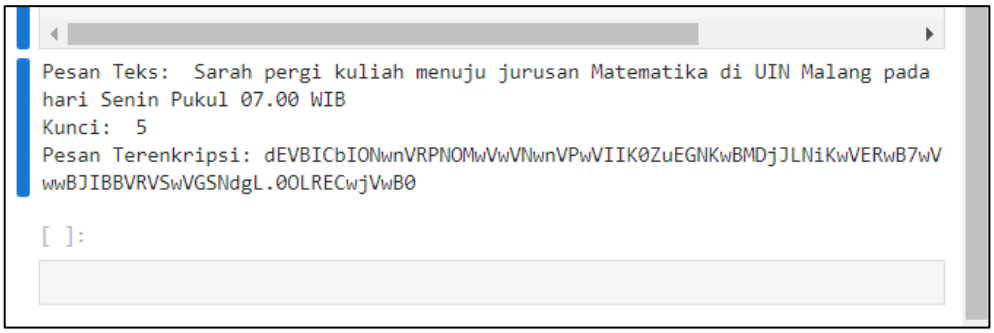


Hasil Dekripsi Contoh 1

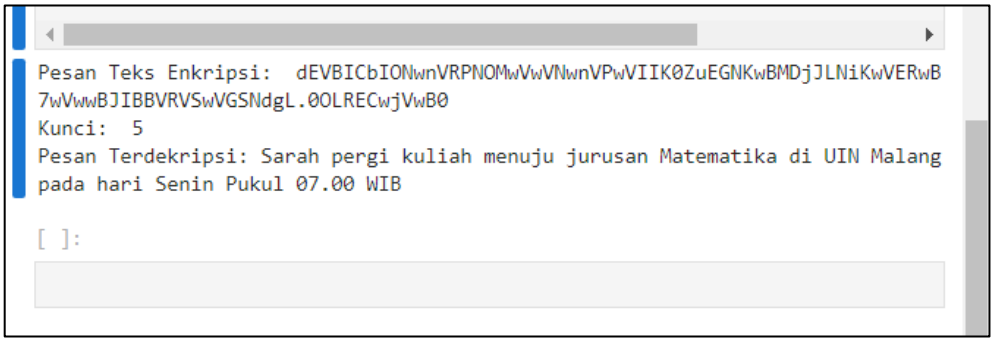


Contoh 2

Hasil Enkripsi Contoh 2

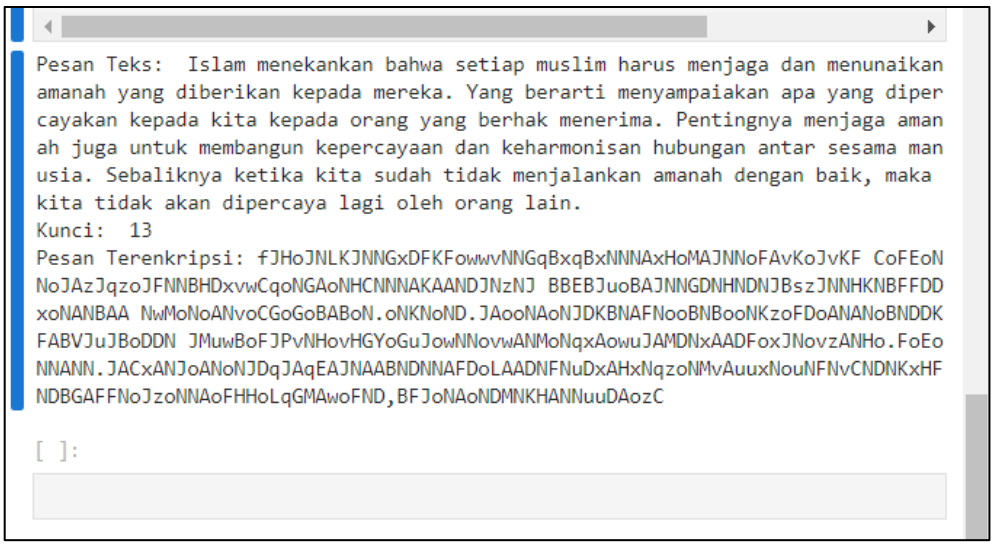


Hasil Dekripsi Contoh 2



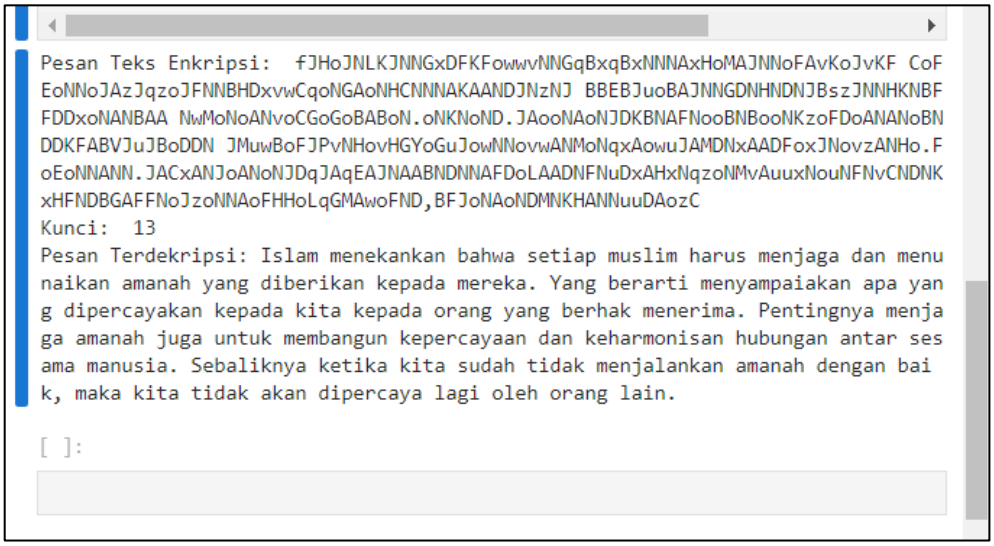
Contoh 3

Hasil Enkripsi Contoh 3



```
Pesan Teks:  Islam menekankan bahwa setiap muslim harus menjaga dan menunaikan
amanah yang diberikan kepada mereka. Yang berarti menyampaiakan apa yang diper
cayakan kepada kita kepada orang yang berhak menerima. Pentingnya menjaga aman
ah juga untuk membangun kepercayaan dan keharmonisan hubungan antar sesama man
usia. Sebaliknya ketika kita sudah tidak menjalankan amanah dengan baik, maka
kita tidak akan dipercaya lagi oleh orang lain.
Kunci:  13
Pesan Terenkripsi: fJHoJNLKJNNGxDFKFowwvNNGqBxqBxNNNAxHoMAJNNoFAvKoJvKF CoFEoN
NoJAzJqzoJFNNBHDxvwCqoNGAoNHCNNNAKAANDJNzNJ BBEBJuoBAJNNGDNHNDNJBszJNNHKNBFFDD
xoNANBAA NwMoNoANvoCGoGoBABoN.oNKNoND.JAooNAoNJDKBNAFNooBNBooNKzoFDoANANoBNDDK
FABVJuJBoDDN JMuwBoFJPvNHovHGYoGuJowNNovwANMoNqxAowuJAMDNxAADFoxJNovzANHo.FoEo
NNANN.JACxANJoANoNJDqJAqEAJNAABNDNNAFDoLAADNFNuDxAHxNqzoNMvAuuxNouNFNvCNDNKxHF
NDBGAFFNoJzoNNAoFHHoLqGMAwoFND,BFJoNAoNDMNKHANNuuDAozC

[ ]:
```

Hasil Dekripsi Contoh 3



```
Pesan Teks Enkripsi:  fJHoJNLKJNNGxDFKFowwvNNGqBxqBxNNNAxHoMAJNNoFAvKoJvKF CoF
EoNNoJAzJqzoJFNNBHDxvwCqoNGAoNHCNNNAKAANDJNzNJ BBEBJuoBAJNNGDNHNDNJBszJNNHKNBF
FDDxoNANBAA NwMoNoANvoCGoGoBABoN.oNKNoND.JAooNAoNJDKBNAFNooBNBooNKzoFDoANANoBN
DDKFABVJuJBoDDN JMuwBoFJPvNHovHGYoGuJowNNovwANMoNqxAowuJAMDNxAADFoxJNovzANHo.F
oEoNNANN.JACxANJoANoNJDqJAqEAJNAABNDNNAFDoLAADNFNuDxAHxNqzoNMvAuuxNouNFNvCNDNK
xHFNDBGAFFNoJzoNNAoFHHoLqGMAwoFND,BFJoNAoNDMNKHANNuuDAozC
Kunci:  13
Pesan Terdekripsi: Islam menekankan bahwa setiap muslim harus menjaga dan menu
naikan amanah yang diberikan kepada mereka. Yang berarti menyampaiakan apa yan
g dipercayakan kepada kita kepada orang yang berhak menerima. Pentingnya menja
ga amanah juga untuk membangun kepercayaan dan keharmonisan hubungan antar ses
ama manusia. Sebaliknya ketika kita sudah tidak menjalankan amanah dengan bai
k, maka kita tidak akan dipercaya lagi oleh orang lain.

[ ]:
```

RIWAYAT HIDUP

Anisa Rahma Fadhila lahir di Malang pada tanggal 3 Januari 1999. Bertempat Tinggal di Sumbersari Kota Malang, Jawa Timur. Merupakan anak pertama dari Bapak Sukaryadi dan Ibu Siti Rodliyah. Perempuan yang akrab disapa Sarah ini telah menempuh pendidikan formal mulai dari TK Muslimat NU 31, Malang. Kemudian melanjutkan pendidikannya di MIN 1 Kota Malang dan Lulus pada Tahun 2011. Menempuh pendidikan SMP di SMPN 13 Malang dan lulus pada tahun 2014. Kemudian melanjutkan pendidikannya di MAN 1 Kota Malang lulus pada tahun 2017. Selanjutnya pada tahun 2017 melanjutkan ke jenjang pendidikan strata 1 di Universitas Islam Negeri Maulana Malik Ibrahim Malang dengan menempuh Program Studi Matematika, Fakultas Sains dan Teknologi. Semasa menjadi mahasiswa aktif mengikuti kegiatan organisasi yang ada didalam kampus, seperti menjadi pengurus Divisi Kewirssausahaan (2017-2018) dan Pengurus Divisi PNJ HMJ “Integral” Matematika UIN Malang (2018-2019).

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

BUKTI KONSULTASI SKRIPSI

Nama : Anisa Rahma Fadhila
NIM : 17610047
Fakultas / Jurusan : Sains dan Teknologi / Matematika
Judul Skripsi : Implementasi Metode Super Enkripsi *Athash Cipher* dan *Rail Fence Cipher* pada Penyandian Pesan Teks Menggunakan *Python*
Pembimbing I : Prof. Dr. H. Turmudi, M.Si., Ph.D
Pembimbing II : Ach. Nashichuddin, M.A

No	Tanggal	Hal	Tanda Tangan
1.	7 Agustus 2023	Konsultasi Topik dan Data	1.
2.	21 Agustus 2023	Konsultasi Bab I, II, dan III	2.
3.	4 September 2023	Konsultasi Bab I, II, dan III	3.
4.	7 September 2023	Konsultasi Kajian Agama Bab I dan II	4.
5.	11 September 2023	ACC Kajian Agama Bab I dan II	5.
6.	14 September 2023	ACC Bab I, II, dan III	6.
7.	18 September 2023	ACC Seminar Proposal	7.
8.	26 September 2023	Konsultasi Revisi Seminar Proposal	8.
9.	22 April 2024	Konsultasi Bab IV dan V	9.
10.	30 April 2024	Konsultasi Bab IV dan V	10.
11.	20 Mei 2024	Konsultasi Kajian Agama Bab IV	11.
12.	28 Mei 2024	ACC Kajian Agama Bab IV	12.
13.	3 Juni 2024	ACC Bab IV dan V	13.

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

14.	7 Juni 2024	ACC Seminar Hasil	14.
15.	11 Juni 2024	ACC Seminar Hasil lanjutan	15.
16.	18 Juni 2024	Konsultasi Revisi Seminar Hasil	16.
17.	21 Juni 2024	ACC Sidang Skripsi	17.
18.	26 Juni 2024	Konsultasi Revisi Sidang Skripsi	18.
19.	28 Juni 2024	ACC Keseluruhan	19.

Malang, 28 Juni 2024
Mengetahui,
Ketua Program Studi Matematika

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