SIGNCRYPTION DOKUMEN GAMBAR MENGGUNAKAN ALGORITMA LLKAKE DAN RSA

SKRIPSI

REYSHA TAZHA FADILLAH

211401095

PROGRAM STUDI S-1 ILMU KOMPUTER

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UNIVERSITAS SUMATERA UTARA

MEDAN

2024

SIGNCRYPTION DOKUMEN GAMBAR MENGGUNAKAN ALGORITMA LLKAKE DAN RSA

SKRIPSI

REYSHA TAZHA FADILLAH

211401095

PROGRAM STUDI S-1 ILMU KOMPUTER

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UNIVERSITAS SUMATERA UTARA

MEDAN

2024

PERSETUJUAN

Judul	:	*SIGNCRYPTION* DOKUMEN GAMBAR MENGGUNAKAN ALGORITMA LLKAKE DAN RSA
Kategori	:	SKRIPSI
Nama	:	REYSHA TAZHA FADILLAH
Nomor Induk Mahasiswa	:	211401095
Program Studi	:	SARJANA (S-1) ILMU KOMPUTER
Fakultas	:	ILMU KOMPUTER DAN TEKNOLOGI INFORMASI UNIVERSITAS SUMATERA UTARA
Tanggal Sidang	:	9 Januari 2025
Komisi Pembimbing	:	

Pembimbing 2	Pembimbing 1
Amer Sharif S.Si, M.Kom NIP. 196910212021011001	Dr. Mohammad Andri Budiman S.T., M.Comp.Sc., M.E.M. NIP. 197510082008011011

Diketahui/Disetujui Oleh
Program Studi S-1 Ilmu Komputer
Ketua,

Dr. Amalia ST., M.T.
NIP. 197812212014042001

PERNYATAAN

SIGNCRYPTION DIGITAL INFORMATION MENGGUNAKAN ALGORITMA LLKAKE DAN RSA

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing – masing telah disebutkan sumbernya.

Medan, 30 Oktober 2024

Reysha Tazha Fadillah
211401095

PENGHARGAAN

Bismillahirrahmanirrahim, segala puji syukur dipanjatkan kepada Allah *Subhanahu Wa Ta'ala* atas segala limpahan rahmat dan hidayah-Nya sehingga penulis dapat berada di tahap penyusunan skripsi ini sebagai syarat untuk mendapatkan gelar Sarjana Komputer di Program Studi S-1 Ilmu Komputer, Universitas Sumatera Utara. Tidak lupa shalawat serta salam tetap tercurahkan kepada Rasulullah *Shalallaahu 'Alayhi Wasallam* yang telah mengeluarkan umat manusia dari kegelapan menuju zaman terang benderang saat ini.

Dengan penuh rasa hormat pada kesempatan ini penulis mengucapkan terima kasih kepada Ibu, Eviana Kartikasari. atas segala bentuk perjuangan, kasih sayang, dan perlindungan dengan doa-doa yang dipanjatkan untuk penulis. Dan terimakasih kepada Ayah, Asyura Budidharma atas dukungan dan kasih sayang yang membersamai di setiap langkah penulis. Terima kasih untuk setiap dukungan yang telahdiberikan hingga penulis dapat berada di titik ini.

Penyusunan skripsi ini tidak terlepas dari bantuan, dukungan, dan bimbingan dari banyak pihak. Oleh karena itu, penulis mengucapkan banyak terima kasih kepada:

1. Bapak Prof. Dr. Muryanto Amin S.Sos., M.Si. selaku Rektor Universitas SumateraUtara.
2. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer danTeknologi Informasi Universitas Sumatera Utara dan Dosen Pembimbing Akademik yang telah memberi banyak dukungan dan motivasi kepada penulis.
3. Bapak Dr. Mohammad Andri Budiman S.T., M.Comp.Sc., M.E.M. selaku Wakil Dekan I Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara dan Dosen Pembimbing I yang telah memberi banyak dukungan, motivasi, masukan, dan bimbingan spiritual yang membangun kepada penulis selama penyusunan skripsi ini.
4. Bapak Amer Sharif S.Si, M.Kom selaku Dosen Pembimbing II yang telah memberi banyak dukungan, motivasi, dan bimbingan spiritual yang membangun kepada penulis selama penyusunan skripsi ini.
5. Ibu Dr. Amalia, S.T., M.T. selaku Ketua Program Studi S-1 Ilmu Komputer Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.

6. Ibu Sri Melvani Hardi S.Kom., M.Kom selaku Sekretaris Program Studi S-1 Ilmu Komputer Fakultas Ilmu Komputer Dan Teknologi Informasi Universitas Sumatera Utara.
7. Seluruh bapak dan ibu dosen Fasilkom-TI USU, khususnya dosen Program Studi S-1 Ilmu Komputer yang telah mendidik dan memberi wawasan serta moral yang berharga, baik di bangku perkuliahan maupun setelah lulus.
8. Keluarga besar dari Ibu dan Ayah, yaitu Oma, Nenek, Bunda, dan sepupu yang telah memberi kasih sayang, doa, dan dukungan yang berharga yang tidak terlupakan.
9. Sahabat dari keluarga cemara, Reggy Arauna, Bima, Reyvo Prawira, Adi teguh, Said Umar yang telah mendoakan, memberi tawa dan kegundahan, senantiasa menemani suka dan duka, serta sumber motivasi dan panutan bagi penulis.
10. Para sahabat dari ‘penghuni gudang’ yang ramai, meriah, dan akan selalu terkenang, terima kasih untuk seluruh pembelajaran barunya, baik dari aspek akademis maupun bekal dalam menjalani kehidupan dunia dan akhirat.
11. Teman dari stambuk 21 yang turut membantu dalam pembelajaran akademis dan organisasi.
12. Pengurus PEMA FASILKOM-TI USU periode 2023/2024 khususnya bidan Hubungan Masyarakat yang telah memberi pengalaman dengan aktivitas yang beragam dan bekerja sama dengan baik dalam menjalankan satu periode kepengurusan
13. Abang-kakak senior terkhusus stambuk 2020 dan 2019 yang telah memberi banyakmasukan, arahan, motivasi, dan doa baiknya kepada penulis selama masa perkuliahan dan penulisan skripsi ini.
14. Adik-adik stambuk 2022 dan 2023 yang telah meluangkan tenaga untuk bekerja sama selama masa kepanitiaan serta memberi doa dan dukungan berharga kepada penulis.

Dan seluruh pihak yang telah memberikan dukungan serta doa baik yang tidak dapat penulis sebutkan satu per-satu. Semoga Allah *Subhanahu Wa Ta‘ala*

senantiasa melimpahkan keberkahan serta kebaikan atas semua dukungan yang telah diberikan kepada penulis dan hasil penelitian ini dapat memberi manfaat maupun inspirasi untukkedepannya.

Medan, 30 Oktober 2024

Penulis,

Reysha Tazha Fadillah

ABSTRAK

Informasi digital merupakan segala sesuatu yang dilihat dan dengar dapat didigitalkan, yang berarti informasi digital dapat mencakup musik, film, foto, dokumen. Namun, peningkatan jumlah informasi digital akan selalu berhubungan dengan meningkatnya pelanggaran keamanan, sehingga dibutuhkannya ilmu dan seni untuk mengamankan dan juga menjaga keaslian pada informasi digital, yaitu dengan *signcryption cryptography. Signcryption* dirancang untuk memberikan kemanan yang efektif. *Signcryption* dibagi menjadi dua tahapan yaitu *digital signature* dan kriptografi asimetris. *Digital signature* LLKAKE memanfaatkan pembangkitan kunci dan bilangan yang berbeda dengan RSA. Pada algoritma ini pembangkitan p dan q menggunakan *Sophie Germain Prime* pada algoritma ini memiliki rumus $P = 2 \times p' + 1$, p' tersebut juga merupakan bilangan prima. Teknik kriptografi asimetris ini menggunakan algoritma RSA, algoritma asimetris yang menggunakan faktorisasi bilangan besar yang merupakan perkalian dari dua buah bilangan prima besar. Implementasi dari penelitian ini akan berbentuk *website* yang mampu melakukan signing dan enkripsi terhadap dokumen digital berupa gambar dengan berbagai format dan juga melakukan dekripsi dan verifikasi tanda tangan digital.

Kata Kunci: *Kriptografi, Signcryption,* Tanda Tangan Digital LLKAKE, Enkripsi dan Dekripsi RSA*, Sophie Germain Prime*

ABSTRACT

Digital information is everything that is seen and heard can be digitized, which means that digital information can include music, movies, photos, documents. However, the increasing amount of digital information will always be associated with increasing security breaches, so that science and art are needed for security and also maintaining the authenticity of digital information, namely with signcryption cryptography. Signcryption is designed to provide effective security. Sign encryption is divided into two stages, namely digital signatures and asymmetric cryptography. LLKAKE digital signatures utilize different keys and numbers than RSA. In this algorithm, the generation of p and q uses Sophie Germain Prime in this algorithm has the formula P = 2 × p '+ 1, p' is also a prime number. This asymmetric cryptography technique uses the RSA algorithm, an asymmetric algorithm that uses large number factorization which is a multiplication of two large prime numbers. The implementation of this research will be in the form of a website that is able to sign and encrypt digital documents in the form of images with various formats and also decrypt and verify digital signatures.

Keywords: *Cryptography, Signcryption, Digital Signature LLKAKE, Encryption and Decryption RSA, Sophie Germain Prime*

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

BAB 1
PENDAHULUAN

1.1. Latar Belakang

Pada era ini, terdapat banyak hal dalam perkembangan teknologi yang bisa membantu dalam memudahkan seseorang dalam berkegiatan maupun berkomunikasi, hal ini diimplementasikan dalam berbagai hal, contohnya seperti aplikasi pengiriman file. namun dalam perkembangan teknologi terdapat banyak data yang bersifat rahasia dan itu harus dipastikan keaslian dan kepemilikannya, oleh karena itu terdapat metode dalam mengamankan sebuah data dengan cara signcryption, yaitu menandatangani dan mengunci sebuah pesan.

Dalam pembuatan signcryption memiliki dua tahapan yaitu digital signature dan encryption, di mana digital signature yang memiliki fungsi dalam membentuk sebuah tanda tangan elektronik yang berguna sebagai tanda kepemilikan dari data tersebut, sedangkan encryption memiliki fungsi dalam melakukan proses mengubah sebuah informasi ataupun data menjadi tidak dapat dibaca oleh siapapun kecuali yang memiliki kunci khusus.

Diperkenalkan oleh trio ilmuwan MIT pada tahun 1977, RSA merupakan algoritma kriptografi asimetris yang memanfaatkan sepasang kunci, yaitu kunci publik dan kunci privat. Berbeda dengan algoritma simetris yang mengandalkan satu kunci tunggal, RSA menawarkan fleksibilitas yang lebih tinggi dengan memungkinkan distribusi kunci publik secara bebas, sementara kunci privat tetap dirahasikan.

Algoritma LLKAKE merupakan algoritma tanda tangan digital yang dikembangkan dari algoritma RSA dan ECC, di buat oleh Farid Lalem, Abdelkader Laouid, Mostefa Kara, Mohammed Al-Khalidi, Amma Elleyan pada tahun 2023. Algoritma LLKAKE memiliki cara pembangkitan kunci yang berbeda dari RSAdan juga ECC, yaitu menggunakan teknik $pk = k + r \times p$, pk adalah kunci public, k dan p adalah 3 kunci privat, dan r merupakan bilangan integer untuk menyamarkan kunci privat dan membuat gangguan acak sehingga mempersulit analisis dan serangan.

1.2. Rumusan Masalah

Informasi digital merupakan bagian sangat penting bagi masyarakat dalam perkembangan teknologi. Maka diperlukan sebuah algoritma yang berfungsi mengamankan dan menjaga keaslian dari dokumen, sehingga pihak luar tidak bisa mengetahui dokumen tersebut, apakah signryption pada algoritma LLKAKE dan RSA dapat menambah efisiensi dan efektifitas pada proses data yang akan diamankan

1.3. Batasan Masalah

Agar penelitian ini lebih terarah, peneliti membatasi ruang lingkup penelitian.

1. Menggunakan metode signcryption
2. Menggunakan algoritma LLKAKE dan RSA
3. Penelitian ini menggunakan bahasa pemrograman python
4. Bentuk implementasi berupa aplikasi web
5. Dokumen yang akan di *signcryption* akan terbatas pada gambar.

1.4. Tujuan Penelitian

Peneliti memiliki tujuan untuk membangun sebuah program untuk membantu seseorang dalam mengamankan dokumen dan memsatikan kepemilikan dari dokumen tersebut, dan juga memastikan efisiensi dan efektivitas pada algoritma *signcryption* tersebut.

1.5. Manfaat Penlitian

Peneliti memiliki harapan untuk penelitian ini dapat memiliki manfaat dalam mengamankan data dengan menggunakan metode signcryption dengan algoritma LLKAKE dan RSA, sehingga dapat meningkatkan efisiensi dan keamanan dari algoritma LLKAKE dan RSA, dan juga memberikan rasa aman dan privasi terhadap pihak yang ingin bertukar pesan ataupun berkomunikasi dari pihak asing. Memberikan bahan pembelajaran tentang kriptografi sehingga mahasiswa mudah dalam menimbang dan mengembangkan mata kuliah pembelajaran kriptografi.

1.6. Metodologi Penelitian

Langkah – langkah yang dilakukan pada proses penyusunan tugas akhir ini adalah:

1. Studi Pustaka

   Sebagai tahap awal, dilakukan kajian pustaka yang komprehensif dengan merujuk pada berbagai publikasi ilmiah, baik berupa buku, jurnal, maupun artikel daring, untuk menggali informasi terkini mengenai metode enkripsi token, tanda tangan digital, enkripsi, dan dekripsi.

2. Analisa dan perancangan

   Peneliti menyelidiki algoritma LLKAKE sebagai algoritma kunci publik yang sesuai untuk digunakan pada informasi digital, dalam membantu meningkatkan keamanan dan privasi data yang di kirim antara pengirim informasi digital dengan penerima. Serta menganalisis kebutuhan penelitian yang akan dilakukan untuk perancangan sistem.

3. Implementasi

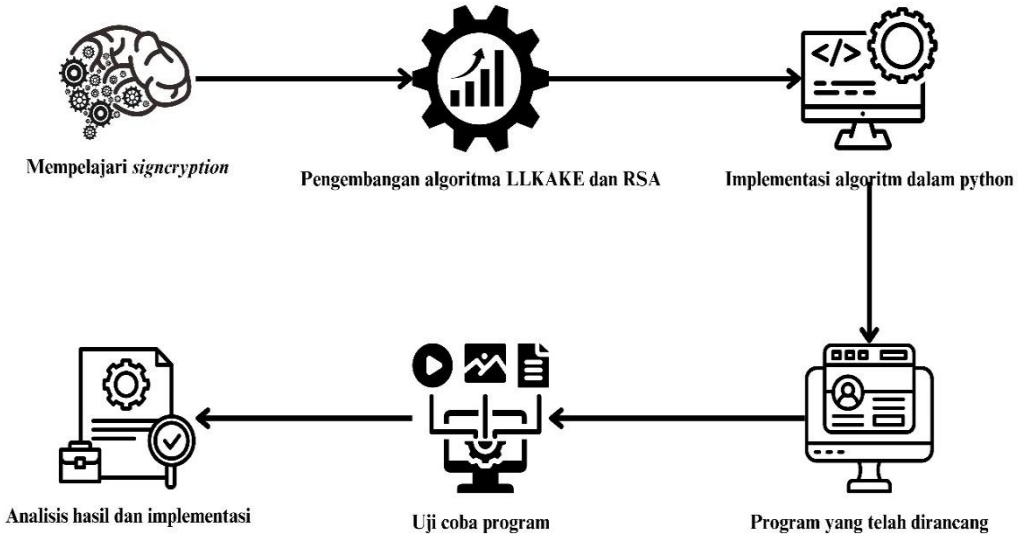
   Tahap ini peneliti melakukan proses pembuatan program berdasarkan diagram alir (*flowchart*) yang telah dirancang dan menganalisis *signcryption* pada algoritma LLKAKE dalam melakukan pengamanan pada digital information, seperti privasi, integritas, dan autentikasi Ketika mengirimkan sebuah data dengan mengimplementasikan algoritma LLKAKE dan RSA.

4. Pengujian

   Tahap ini, peneliti melakukan proses uji coba apakah penggunaan algoritma LLKAKE dan RSA *efisien* dan *efektif* pada metode *signcryption.*

5. Dokumentasi

   Pada tahap ini, peneliti melakukan dokumentasi pada setiap tahap yang terjadi dalam penelitian dan membuat Kesimpulan akhir dalam bentuk skripsi.



Gambar 1. 1 Alur penelitian

1.7. Penelitian Relevan

Kajian literatur menunjukkan bahwa beberapa penelitian sebelumnya telah meneliti topik yang serupa, seperti berikut:

1. Berdasarkan penelitian (Lalem, Laouid, Kara, Al-Khalidi, & Eleyan 2023) menyajikan skema tanda tangan digital inovatif yang tangguh dan efisien, mengungguli teknik modern lainnya dalam hal ukuran tanda tangan dan waktu eksekusi. Skema yang diusulkan menyediakan tanda tangan digital baru untuk teknik enkripsi asimetris, memanfaatkan kunci publik dalam bentuk k + r × p, di mana k dan p merupakan privat key yang sangat diperlukan dalam proses verifikasi dan tanda tangan. Melalui analisis terhadap skema yang diusulkan, menunjukkan ketahanannya terhadap serangan pemulihan kunci privat dan serangan pemalsuan. Lebih jauh, peneliti menyajikan contoh tanda tangan di dunia nyata dan melakukan serangkaian pengujian dengan berbagai ukuran masukan untuk menunjukkan waktu eksekusi proses tanda tangan dan verifikasi. Dengan membandingkan skema yang diusulkan dengan teknik modern lainnya, peneliti menetapkan bahwa pendekatan tersebut menawarkan ukuran tanda tangan yang lebih kecil, hanya 320 bit, berkat penggunaan kunci publik berukuran kecil. Selain itu, peneliti mencapai waktu eksekusi yang lebih cepat, hanya 48,4 ms untuk seluruh operasi tanda tangan, termasuk

penandatanganan dan verifikasi, karena lebih sedikit operasi yang digunakan oleh peneliti. Melihat ke arah masa depan, peneliti bermaksud mengembangkan metode *digital signature* berdasarkan kumpulan proposal peneliti, yang memungkinkan dua atau lebih tanda tangan digital untuk digabungkan menjadi satu tanda tangan yang lebih pendek. Ini akan memungkinkan penanda tanganan yang berbeda untuk membuat tanda tangan menggunakan skema yang disajikan.

2. Dalam penelitian yang dilakukan oleh Arief dan Saputra (2016), perbandingan kinerja antara algoritma RSA dan RSA-CRT pada aplikasi pesan instan menunjukkan hasil yang menarik. Hasil penelitian tersebut mengindikasikan bahwa penggunaan RSA-CRT mampu meningkatkan efisiensi proses dekripsi secara signifikan, terutama untuk data dengan ukuran yang besar. Hal ini menunjukkan bahwa RSA-CRT memiliki potensi yang besar untuk diaplikasikan pada sistem yang membutuhkan pemrosesan data yang cepat dan real-time.
3. Berdasarkan penelitian (Mohamad, Din, & Ahmad 2021) makalah ini telah menyajikan dan mempelajari beberapa skema RSA sejak dekade terakhir. Berdasarkan survei literatur, dapat disimpulan bahwa kelemahan skema RSA pun memiliki ukuran kunci yang besar, yaitu kekuatan adalah keamanan dan dapat diimplementasikan pada aplikasi yang berbasis teknologi internet. Itu juga menunjukkan bahwa ada banyak upaya yang telah dikemukakan oleh para peneliti sebelumnya dalam satu dekade terakhir.
4. Berdasarkan penilitan (Ginting, C.L., Budiman, M. A., & Nasution, S. 2024) analisis dan pengujian metode *signcryption* yaitu enkripsi kemudian tanda tangan dan verifikasi kemudian dekripsi menggunakan skema tanda tangan digital RSA dengan modifikasi matriks dan algoritma Cayley-Purser telah berhasil dilakukan dengan plaintext berupa kumpulan string yang terdiri dari jumlah karakter yang bervariasi pada setiap string serta menggunakan nilai modulus n dari 10 digit hingga maksimal yang panjangnya tidak dibatasi. Proses enkripsi dan dekripsi memakai algoritma Caley-Purser berbanding lurus dengan jumlah karakter plaintext, dimana semakin banyak jumlah karakter maka running time sebenarnya saat melakukan enkripsi dan dekripsi akan semakin lama.

1.8. Sistematika Penulisan

Dalam Struktur penyusunan tugas akhir ini, terbagi menjadi lima bab, dengan setiap bab dijelaskan sebagai berikut:

BAB 1 PENDAHULUAN

Pada bagian ini kita akan memahami penjelasan di balik pemilihan judul, pembahasan masalah, termasuk sasaran penelitian, pemanfaatan hasil ujian, strategi penelitian, dan ikhtisar artikel

BAB 2 LANDASAN TEORI

Kriptografi, ilmu yang mempelajari teknik pengamanan informasi, menjadi fokus utama dalam penelitian ini. Dengan menitikberatkan pada algoritma kunci publik seperti RSA dan teknik signcryption, penelitian ini menggali lebih dalam mengenai mekanisme otentikasi data digital. Konsep pembagi sekutu terbesar (GCD) sebagai salah satu landasan matematis dalam kriptografi juga dibahas secara mendalam, terutama dalam konteks algoritma seperti LLKAKE.

BAB 3 ANALISIS DAN PERANCANGAN

Analisis perancangan merupakan tahapan yang menjelaskan mengenai analisis pada algoritma dan dilakukan perancangan diagram yang diperlukan, seperti diagram alir.

BAB 4 IMPLEMENTASI DAN PENGUJIAN

Pada bagian ini menjelaskan penerapan algoritma pada program yang diuji pada aplikasi berbasis web dan menjelaskan hasil pengujian yang dilakukan.

BAB 5 PENUTUP

Pada bagian ini berisi kesimpulan yang dapat diperoleh berdasarkan pemaparan pada setiap bab serta saran yang diberikan peneliti sebagai masukan untuk penelitian kedepannya.

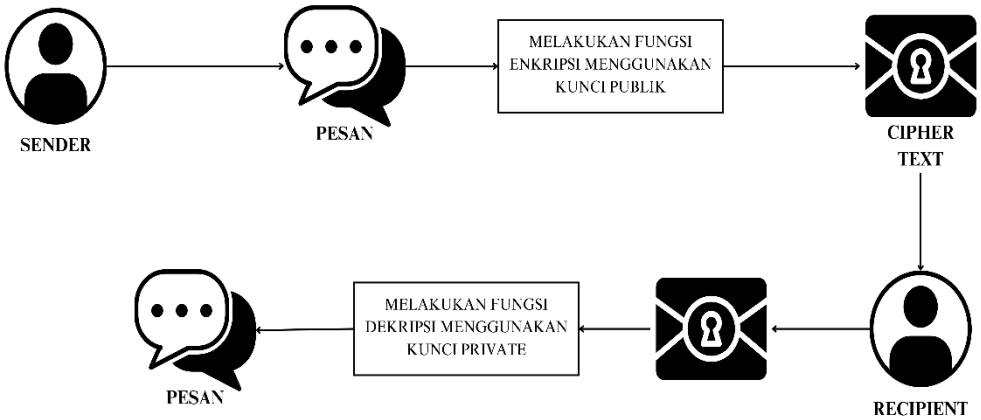
BAB 2
LANDASAN TEORI

2.1. Kriptografi

Merupakan studi tentang penerapan komunikasi yang aman terhadap dua pihak. Biasanya, dua pihak ingin saling bertukar pesan, namun tidak ingin pihak ketiga dapat memahami pesan tersebut. Enkripsi sering digunakan untuk meningkatkan keamanan sistem informasi dan mencegah pelanggaran data dan intersepsi data selama pertukaran informasi. Prosedur enkripsi memastikan keamanan komputer pemilik dan mencegah akses luar mencapai komputer secara langsung, namun prosedur tersebut harus dilakukan menggunakan prosedur enkripsi. Kriptografi terbagi dua yaitu kriptografi asimetris dan kriptografi simetris.

2.1.1 kriptografi asimetris

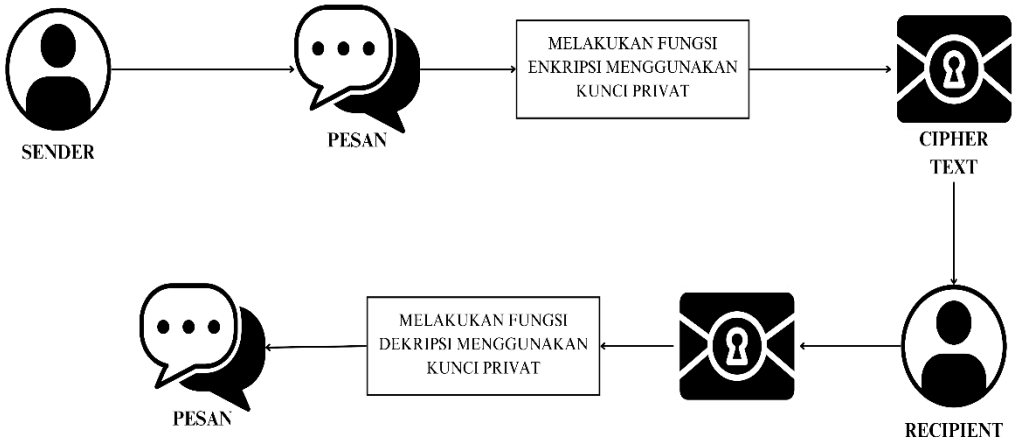
merupakan sebuah metode pengamanan data menggunakan sepasang kunci untuk mengenkripsi dan mendekripsi yang dibuat oleh penerima pesan, kunci tersebut dinamakan kunci privat dan kunci publik. Kriptografi asimetris memiliki kelemahan pada waktu komputasi sangat lama.



Gambar 2. 1 Kriptografi asimetris

2.1.2 Kriptografi simetris

Merupakan sebuah metode pengamanan data yang menggunakan satu kunci yang sama untuk proses enkripsi dan dekripsi. Kriptografi simetris memiliki kelemahan terhadap kunci enkripsi dan dekripsi yang sama.



Gambar 2. 2 Kriptografi simetris

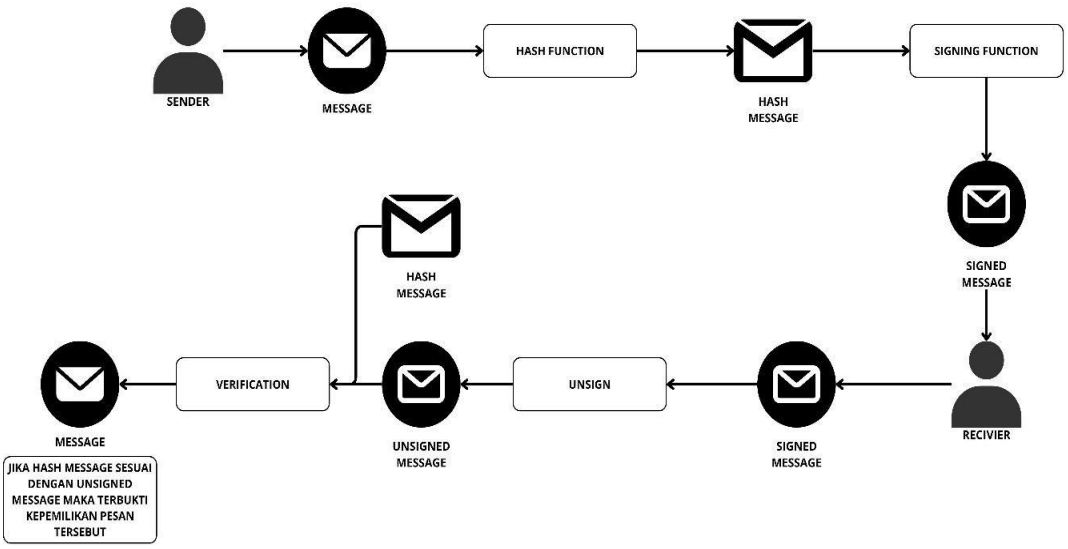
2.1.3 Aspek Ratio enkripsi

Aspek rasio pada enkripsi merupakan perbandingan besar ukuran file setelah melakukan proses enkripsi dengan file semula.

$$aspek\ rasio = \frac{file\ enkripsi}{file\ asli}$$

2.2. Digital Signature

Sebagai sebuah mekanisme matematis, tanda tangan digital berfungsi untuk memastikan keaslian dan utuh tidaknya data digital, seperti pesan, perangkat lunak, atau dokumen. Tanda tangan digital merupakan implementasi tanda tangan elektronik yang mematuhi peraturan hukum yang ketat.



Gambar 2. 3 Metode *Digital Signature*

Langkah-langkah *digital signature*

1. Pesan diubah ke dalam *hash message* menggunakan fungsi hash
2. Hash message merupakan intisari pesan yang telah diubah ke dalam fungsi hash
3. *Hash message* di signing menggunakan metode *encryption* menjadi *signed message*
4. *Signed message* merupakan perubah dari hash message menggunakan *signing function*
5. Melakukan *unsign* terhadap *signed message*
6. *Signed message* yang telah di-*unsign* diverifikasi dengan melakukan perbandingan dengan *hash message*
7. Apabila *signed message* yang telah di *unsign* sama dengan *hash message* maka telah terverifikasi bahwa pengirimlah yang mengirim pesan tersebut

2.3 Fermat little theorem

Fermat little theorem adalah sebuah konsep fundamental dalam teori bilangan yang memiliki aplikasi yang sangat luas, terutama dalam bidang kriptografi. Teorema ini menunjukkan p sebuah angka prima sedangkan a sebuah angka bulat positif yang tidak habis ketika dibagi oleh p, maka:

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

Mengambil suatu bilangan bulat a dalam ruang lingkup p – 1, di mana:

$$1 < a < p$$

Jika a menghasilkan satu dalam modulus p, maka p adalah prima

Contoh: apakah 5 merupakan bilangan prima ?

$$p = 5$$

$$1 < a < p = \{2, 3, 4\}$$

$$a = 2, 2^4 \; mod \; 5 = 1$$

$$a = 3, 3^4 \; mod \; 5 = 1$$

$$a = 4, 4^4 \; mod \; 5 = 1$$

2.4 Diophantine linear

Diophantine linear adalah jenis persamaan khusus dalam matematika yang mencari Solusi bilangan bulat. Persamaan ini memiliki bentuk yang sederhana

$$ax + by = c$$

2.5 Extended Euclidean algorithm

Extended Euclidean algorithm adalah pengembangan dari algoritma Euclides yang tidak hanya menghitung *great common divisor* dari dua bilangan bulat, tetapi juga menemukan koefisien – koefisien yang memenuhi identitas bezout

$$ax + by = \gcd(a, b)$$

Contoh: $56x + 16y =$

Tabel 2. 1 Penyelesaian *extended euclidean algorithm*

X	Y	D	K
1	0	56	
0	1	16	3
1	-3	8	

$$56x + 16y = 8$$

- Nilai a dan nilai b diganti dalam d
- ketika nilai a lebih besar dari nilai b maka X berada di paling atas
- ketika nilai b lebih besar dari a maka Y berada di paling atas

- dilakukan perhitungan, di mana D pada nilai Y harus mendekati nilai D pada nilai X.

$$16 \times 3 = 48$$

- nilai yang dikalikan pada Y diubah menjadi variabel K
- selanjutnya nilai X dan Y pada D akan dikalikan dengan K
- pengulangan dilakukan sampai nilai D = 0.

2.6 Inverse modulo

Inverse merupakan balikan dan *modulo* merupakan hasil sisa operasi pembagian satu bilangan dengan bilangan lainnya. $m^{-1}\ (mod\ n)$ adalah inverse m dalam $modulo\ n$ memiliki inverse adalah:

$$GCD\ (m, n) \text{ dan } m > 1$$

Inverse dari $m\ (mod\ n)$ adalah bilangan bulat dinotasikan sebagai m^{-1}, sehingga :

$$m^{-1}.m\ (mod\ n) = 1$$

Perkalian *inverse modulo* merupakan solusi menghitung inverse dalam modulo. Dan perhitungan *inverse modulo* banyak dimanfaatkan pada beberapa algoritma kriptografi, salah satunya algoritma RSA.

Contoh: Berapa *inverse* dari 5 $(mod\ 7)$

Tabel 2. 2 *Inverse Modulo*

$\boldsymbol{m^{-1}}$	$\boldsymbol{m^{-1} \cdot 5\ (mod\ 7)}$
1	5
2	3
3	1

- Berdasarkan Tabel 2.2 didapatkan bahwa bilangan bulat m^{-1}yang memenuhi $m^{-1}.m\ (mod\ n) = 1$ adalah 3, maka *inverse* dari 5 $(mod\ 7)$ adalah 3

2.7 Fungsi hash

Merupakan sebuah fungsi matematika yang memproses masukan data dengan ukuran berapapun dan menghasilkan keluaran berupa string dengan Panjang yang tetap. Output ini sering disebut sebagai nilai hash, digest, atua sidik jari digital dari data input.

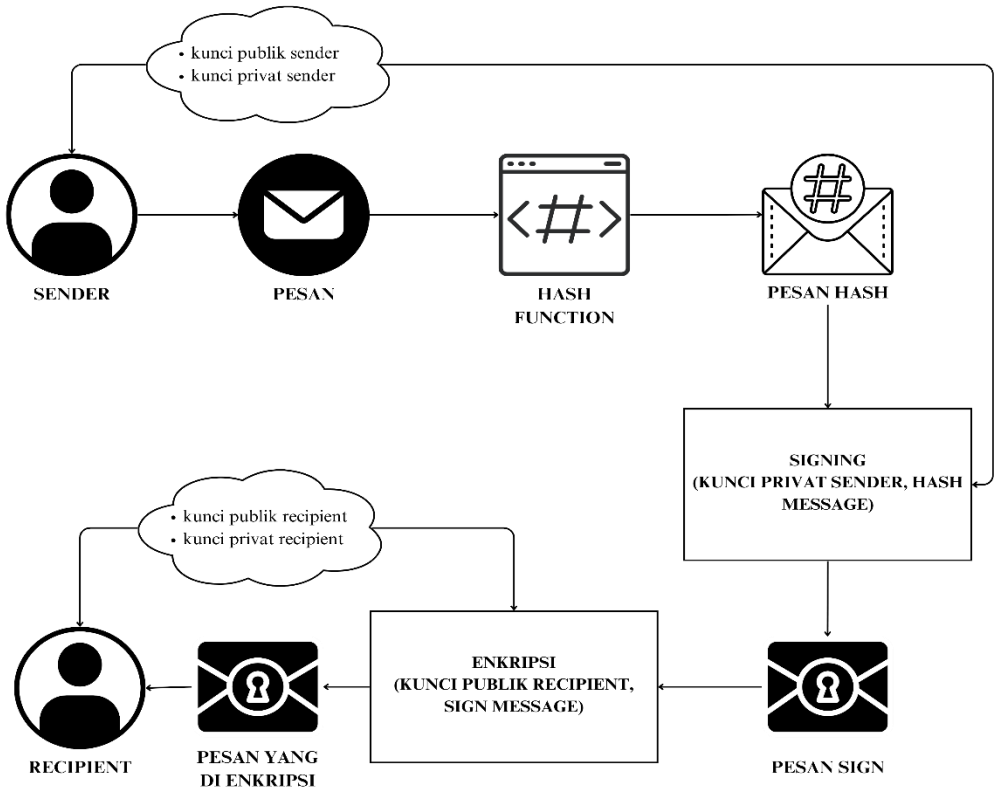
Karakteristik Fungsi *Hash*:

1. Fungsi hash merupakan fungsi yang satu arah sehingga tidak mungkin untuk menghitung input asli nilai hash yang diberikan.
2. Nilai masukan yang sama akan selalu menghasilkan hasil hash yang sama.
3. Perubahan sekecil apapun pada masukan akan menghasilkan perubahan yang sangat besar pada nilai *hash*.
4. Sangat sulit untuk menemukan dua input yang berbeda menghasilkan nilai hash yang sama.

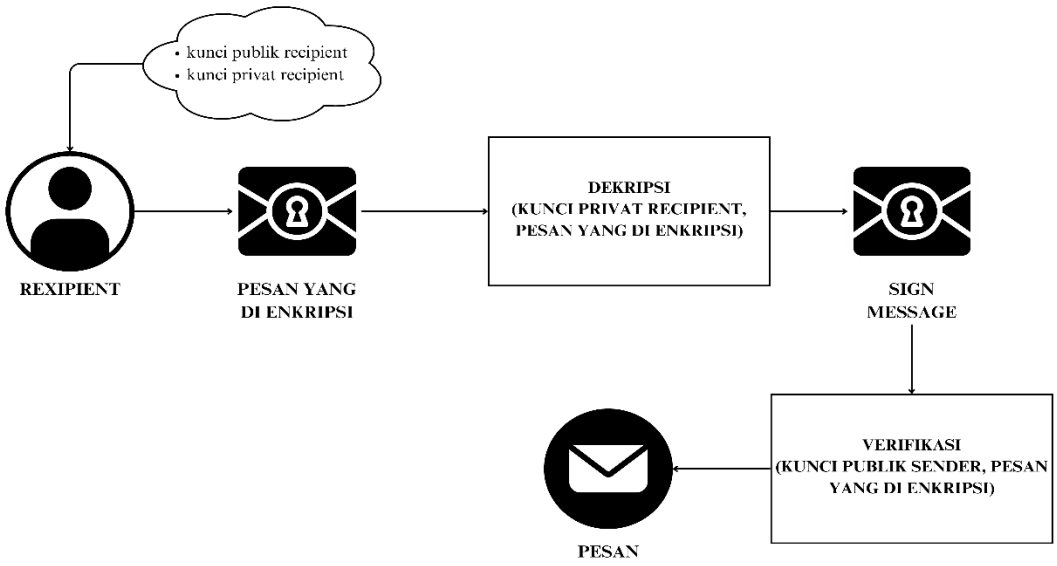
2.8. *Signcryption*

Signcryption adalah sebuah teknik kriptografi yang menggabungkan fungsionalitas dari tanda tangan digital (*digital signature*) dan enkripsi dalam satu langkah komputasi. Ini berarti bahwa pesan yang dienkripsi juga secara simultan ditandatangani, sehingga memberikan jaminan autentikasi (keaslian) dan kerahasiaan pesan dalam satu proses.

Teknik *signcryption* memiliki dua langkah algoritma, yang pertama dengan melakukan *digital signature* dan selanjutnya melakukan *encryption* cara ini dinamakan *sign then encrypt,* cara kedua merupakan kebalikan dari cara pertama, cara ini melakukan *encryption* terlebih dahulu dan selanjutnya melakukan *digital signature,* cara ini dinamakan *encrypt then sign.*Dengan menggunakan signcryption, sebuah informasi digital dapat diamankan dengan efisiensi dan efektif, sehingga dapat mencegah pihak yang tidak sah mengetahui informasi tersebut. *Signcryption* berguna untuk pengamanan maksimal dengan menjaga tiga hal penting yaitu kerahasiaan, integritas, dan autentikasi.



Gambar 2. 4 Tanda tangan dan enkripsi



Gambar 2. 5 Dekripsi dan verifikasi

Langkah – Langkah *signcryption*

1. Pengirim merubah pesan menjadi *hash message* menggunakan fungsi hash.

2. Pengirim melakukan proses *signing hash message* dengan menggunakan kunci privat pengirim.
3. Pengirim melakukan enkripsi.
4. Pesan yang telah melakukan proses enkripsi selanjutnya akan diserahkan kepada penerima.
5. Penerima menggunakan fungsi dekripsi untuk melakukan dekripsi pada pesan.
6. Penerima melakukan *unsigning* pada pesan.
7. Penerima melakukan verifikasi terhadap pesan yang telah di unsigning.
8. Apabila pesa telah diverifikasi dan memiliki hasil yang sama maka terbukti bahwa pesan tersebut dari pengirim.

Contoh Langkah – Langkah *sign then encrypt*

1. Alice (pengirim) membuat kunci untuk melakukan tanda tangan.
2. Bob (penerima) membuat kunci untuk melakukan enkripsi.
3. Alice melakukan tanda tangan terhadap pesan, menggunakan kunci privat milik si Alice.
4. Setelah pesan ditanda tangani selanjutnya Alice akan melakukan enkripsi terhadap pesan menggunakan kunci publik milik si Bob.
5. Setelah Alice melakukan enkripsi terhadap pesan tersebut, maka alice akan mengirimkan pesan tersebut kepada bob.
6. Pesan yang dikirim Alice akan diterima si Bob.
7. Lalu Bob akan melakukan dekripsi pada pesan tersebut menggunakan kunci privat milik si Bob.
8. Setalah itu Bob akan mendapat pesan yang sudah di tanda tangani.
9. Selanjutnya Bob akan melakukan *usign* terhadap pesan tersebut.
10. Bila pesan tersebut terbaca, maka benarlah bahwa:
    - Alice lah yang mengirimkan pesan tersebut.
    - Pesan telah terkirim secara aman.

2.9. Algoritma LLKAKE

Mengambil dari Teknik *digital signature* dari RSA dan ECC mengembangkan nya menjadi algoritma LLKAKE yang dikembang kan oleh Farid Lalem, Abdelkader Laouid, Mostefa Kara, Mohammed Al-Khalidi, Amma Eleyan pada tahun 2023,yang Dimana secara garis besar memiliki cara pembangkitan kunci yang berbeda dari RSA

dan juga ECC. Berbeda dalam pembangkitan kunci dengan RSA dan ECC, algoritma LLKAKE menggunakan rumus pembangkitan kunci publik $Pk = k + r \times p$.

***Key Generation* (pembangkitan kunci)**

Teknik yang diusulkan menggunakan kunci privat k dan *trapdoor* p untuk menghasilkan *digital signature* yang direpresentasikan dalam dua bagian S1 dan S2, yang digunakan dalam melakukan proses *signing* dan verifikasi.

1. Bangkitkan bilangan bulat r yang dipecah menjadi dua yaitu $r1$ & $r2$
2. Kunci privat sender adalah $(Ks, r2)$
3. Kunci publik sender adalah $(PKs, n, r1)$
4. Bangkitkan kunci publik menggunakan rumus

$$PKs = Ks + r \times p$$

5. Untuk membangkitkan kunci publik yang aman bilang bulat r harus terverifikasi

$$r > q$$

6. Bangkitkan n dengan rumus

$$n = p \times q$$

7. p & q haruslah bilangan prima yang aman di mana:
   - $p = 2 \times p' + 1$
   - $q = 2 \times q' + 1$
   - p'& q' harus bilangan prima

***Signing* (tanda tangan)**

1. Pengirim melakukan signing pesan dengan menghitung nilai $S1$ & $S2$

2. Menghitung nilai S1

$$S1 = m^{Ks} \, mod \, n$$

3. Menghitung nilai S2

$$S2 = m^{r2 \times p} \, mod \, n$$

4. *Signing message* yang diberikan kepada penerima adalah $sig_m = (S1, S2)$

***Verify* (verifikasi)**

1. Penerima harus menghitung nilai x menggunakan kunci PKs

$$x = m^{PKs} \, mod \, n$$

2. Penerima menggunakan nilai m yang dihitung ntuk mendeteksi manipulasi teks sandi dan memastikan integritas data, penggunaan kunci publik pengirim memastikan keaslian, kemudian hitunglah nilai y:

$$y = (S1 \times S2^{r1}) mod \, n$$

3. Lalu bandingkan x dengan y

$$x = y$$

Contoh:

1. Bangkitkan nilai r1 dan r2

$$r1 = 5, r2 = 4$$

2. Hitung nilai r

$$r = r1 \times r2$$

$$r = 5 \times 4$$

$$r = 20$$

3. Bangkitkan nilai p' dan q' dengan syarat nilainya merupakan bilangan prima

$$p' = 3, q' = 5$$

4. Hitunglah nilai p, q.

$$p = 2 \times p' + 1$$

$$p = 7$$

$$q = 2 \times q' + 1$$

$$q = 11$$

5. Nilai r harus lebih besar dari q sehingga

$$r > q$$

6. Bangkitkan Ks

$$Ks = 6$$

7. Hitung PKs

$$PKs = Ks + r \times p$$

$$PKs = 6 + 20 \times 7$$

$$PKs = 146$$

8. Misalkan m adalah “BA”

9. Ubah m menjadi tabel encoding

A	B	C	D	E	F
0	1	2	3	4	5

Gambar 2. 6 Tabel *encode*

10. Nilai m = 10

11. Hitunglah S1

$$S1 = m^{Ks} \, mod \, n$$

$$S1 = 10^{6} \, mod \, 77$$

$$S1 = 1$$

12. Hitunglah S2

$$S2 = m^{r2 \times p} \, mod \, n$$

$$S2 = 10^{28} \, mod \, 77$$

$$S2 = 67$$

13. Simpan S1 dan S2

14. Penerima menerima S1 dan S2

15. Penerima menerima S1 dan S2

16. Penerima menghitung X

$$x = m^{PKs} \, mod \, n$$

$$x = 10^{146} \, mod \, 77$$

$$x = 23$$

17. Hitunglah nilai Y

$$y = (S1 \times S2^{r1}) mod \, n$$

$$y = (1 \times 67^{5}) mod \, 77$$

$$y = 23$$

18. Bandingkan nilai X dan Y

$$x = 23, y = 23$$

$$x = y$$

19. Pesan telah terverifikasi dimana nilai X dan Y itu sama

2.10. Algoritma RSA

Merupakan algoritma kunci publik yang paling popular dan sering digunakan sampai saat ini. Pada tahun 1977, tiga ilmuwan dari Massachusetts Institute of Technology, yaitu Ron Rivest, Adi Shamir, dan Len Adleman, memperkenalkan algoritma RSA kepada publik. RSA merupakan algoritma yang menggunakan dua kunci berbeda, yaitu kunci privat dan kunci publik. Berbeda dengan algoritma simetris yang hanya menggunakan satu kunci, yaitu kunci privat. Algoritma RSA membangkitkan kunci dengan cara mengambil dua bilangan prima yang berbeda.

Key Generation (pembangkitan kunci)

1. Bangkitkan dua bilangan prima yaitu p dan q dimana p ≠ q
2. Hitung $n = p \times q$
3. Hitung $\varphi(n) = (p-1) \times (q-1)$
4. Bangkitkan e dengan syarat:
    - e ∈ bilangan bulat positif
    - $1 < e < \varphi(n)$
    - $GCD(e, \varphi(n))$
5. Hitung $d \equiv e^{-1}(mod\ \varphi(n))$
6. Publish kunci publik $(e,\ n)$
7. Simpan kunci privat $(d, p, q, \varphi(n))$

Encryption **(enkripsi)**

1. Dapatkan kunci publik dari recipient (e, n)
2. Ubah message menjadi angka dengan memanfaatkan tabel encoding atau ASCII
3. Enkripsi dengan rumus $C = m^e\ mod\ n$
4. Kirim nilai C tanpa mengubah nya menjadi simbol / karakter

***Decryption* (dekripsi)**

1. Terimalah nilai C dari *sender*
2. Dekripsi dengan rumus $m = c^d \ mod \ n$

Contoh:

1. Bangkitkan lah bilangan prima p dan q

$$p = 5, q = 7$$

2. Hitung n

$$n = p \times q = 5 \times 7 = 35$$

3. Hitung $\varphi(n)$

$$\varphi(n) = (5-1) \times (7-1) = 24$$

4. Bangkitkan e

$$e = 5$$

5. Bangkitkan nilai d

$$d = e^{-1}(mod \ \varphi(n))$$

Tabel 2. 3 *Extended Euclidean Algorithm*

d	$d = e^{-1}(mod \ \varphi(n))$
1	5
2	10
3	15
4	20
5	1

6. Misalkan pesan adalah “C”

A	B	C	D	E	F
0	1	2	3	4	5

Gambar 2. 7 Tabel *encode*

7. Nilai m = C = 2
8. Enkripsi dengan rumus

$$C = m^e \ mod$$

$$C = 2^5 \ mod$$

$$C = 32$$

9. Kirim nilai C tanpa mengubah nya menjadi simbol
10. Penerima melakukan dekripsi dengan rumus

$$m = C^d \, mod \, n$$

$$m = 32^5 \, mod \, 35$$

$$m = 2$$

11. Pesan telah kembali sesuai dengan tabel *encoding*

BAB 3
ANALISIS DAN PERANCANGAN

3.1. Analisis

Merupakan langkah memecahkan dan menguraikan informasi menjadi bagian yang lebih kecil sehingga mudah dipahami. Proses analisis merupakan landasan awal sebelum melakukan perancangan dan pengembangan suatu sistem, agar sesuai dengan kebutuhan dan lebih terstruktur dalam mencapai tujuan akhir.

3.1.1 Analisis Masalah

Tahapan mengidentifikasi sebab dan akibat suatu permasalahan. Penelitian ini mengidentifikasi permasalahan pada dokumen digital, terutama dalam hal keamanan. Dokumen digital merupakan sebuah dokumen yang sering digunakan setiap orang dalam berbagai hal, dari hal pembelajaran, berbagi informasi, hiburan. Dalam kondisi tertentu dokumen tersebut sangat penting dan bersifat rahasia.

Pada tahap mengidentifikasi masalah, penelitian ini menggunakan metode *5-Whys* untuk mempermudah proses analisis. Metode *5-Whys* merupakan metode tanya – jawab sederhana yang cukup efektif ketika focus utamanya adalah mengidentifikasi sebab dan akibat dari suatu masalah. Metode ini dilakukan dengan bertanya "mengapa" secara berulang sebanyak 5 kali atau lebih, diantaranya sebagai berikut:

a. Mengapa ada kebutuhan sistem kriptografi pada dokumen digital?
   Dokumen digital merupakan sebuah dokumen elektronik dan merupakan bentuk digital dari dokumen fisik, dokumen digital berfungsi dalam memudahkan pekerjaan dan mengurangi sumber biaya, tetapi pada era digital ini semakin banyak dokumen digital yang penting yang membutuhkan keamanan, sehingga hanya pihak yang berwenang lah yang bisa membaca dokumen digital tersebut. Oleh karena itu kriptografi bertujuan dalam mengamankan dokumen digital, degan adanya kriptografi sebuah dokumen yang awalnya bisa dibaca dapat

diubah atau dimanipulasi menjadi sebuah karakter yang tidak mungkin dibaca, sehingga pihak luar tidak dapat mengetahui apa isi dokumen tersebut.

b. Mengapa dokumen digital memerlukan *Signcryption*?

Pada era digital sekarang ada dua hal yang sangat penting. Yaitu keamanan dan keaslian, dua hal ini dimiliki pada *signcryption*. *Signcryption* menawarkan dua hal yaitu kriptografi asimetris dan juga digital signature. Kriptografi asimetris sebagai langkah keamanan sedangkan, digital signature sebagai langkah kepemilikan yang berarti bahwa dokumen yang akan dikirim akan dipastikan kepemilikannya, sehingga tidak adanya pemalsuan dokumen.

c. Mengapa efektivitas sangat menjadi hal yang penting dalam kriptografi?

Efektivitas merupakan salah satu pilar utama dalam dunia kriptografi. Kriptografi, sebagai ilmu yang mempelajari teknik-teknik untuk mengamankan data, sangat bergantung pada algoritma dan sistem yang tidak hanya aman, tetapi juga efektif.

d. Mengapa penting untuk mempertimbangkan keamanan informasi dalam era digital?

Pada era digital ini semua hal banyak dilakukan secara *online*, dan mengurangi pekerjaan fisik. Berjalan dengan perkembangan era digital ini, banyak data yang tecipta, mulai dari sidik jari, bentuk wajah, rambut, suara, dan gambar. Karena itu keamanan merupakan bagian penting untuk keberlangsungan dari era ini untuk tetap bertahan.

e. Mengapa menggunakan RSA dan LLKAKE sebagai teknik kriptografi asimetris dan juga tanda tangan digital?

Algoritma RSA merupakan algoritma yang populer dalam melakukan proses enkripsi dan juga tanda tangan digital, RSA memiliki faktorisasi bilangan bulat yang sangat besar menjadi faktor prima penyusunnya membuat nya menjadi kompleks dan susah untuk diserang. Algoritma LLKAKE merupakan algoritma yang dikembangkan pada tahun 2023 dengan berfokus pada tanda tangan digital, algoritma ini dikembangkan dari algoritma RSA sehingga memiliki cara faktorisasi yang terbilang sama dengan RSA.

3.1.2 Analisis kebutuhan

Analisis kebutuhan adalah langkah penting yang berfokus pada pengenalan dan pemahaman kebutuhan yang diperlukan untuk merancang sistem dalam memenuhi

tujuan. Analisis kebutuhan dibagi menjadi dua bagian utama, yaitu pertama adalah fungsional dan yang kedua adalah non-fungsional.

1. Kebutuhan fungsional

   Kebutuhan fungsional adalah spesifikasi fungsionalitas yang dapat dilakukan dan harus ada pada suatu sistem dalam mencapai tujuan. Penelitian ini memiliki kebutuhan fungsional utama, yaitu:

   a. Membangkitkan dua buah kunci, yaitu kunci publik dan juga kunci privat pada algoritma RSA yang akan digunakan dalam proses enkripsi dan dekripsi.
   b. Membangkitkan kunci publik dan kunci privat pada algoritm LLKAKE yang akan digunakan dalam proses *signing* dan verifikasi pada *digital signature*.
   c. Pembangkit bilangan prima menggunakan *Fermat little theorem* dalam membantu pembangkitan kunci publik dan kunci privat algoritma RSA dan LLKAKE.
   d. Menerima masukan *plaintext* dan mengubahnya dalam bentuk angka sesuai dengan panjang tabel encoding yang telah disepakati, yaitu tabel ASCII.
   e. *Plaintext* yang telah diubah menjadi angka dibagi dalam beberapa blok mengikuti ukuran kunci.
   f. Proses enkripsi dan signing dilakukan per-satu karakter dengan panjang tabel ASCII.
   g. Menghasilkan *ciphertext* dan *signed message* yang tidak terbaca dan bermakna dari proses enkripsi dan *digital signature* yang telah dilakukan.
   h. Melakukan proses dekripsi dengan mengubah pesan terenkripsi menjadi pesan asli menggunakan algoritma RSA menggunakan kunci yang dihasilkan. Hasil angka yang telah di dekripsi di kembalikan lagi melalui konversi tabel ASCII.
   i. Hasil angka yang telah diubah kembali menjadi karakter lalu digabungkan sehingga menjadi *plaintext*.
   j. Melakukan uji coba perbandingan pada pembangkitan kunci dengan waktu proses.

2. Kebutuhan non-fungsional

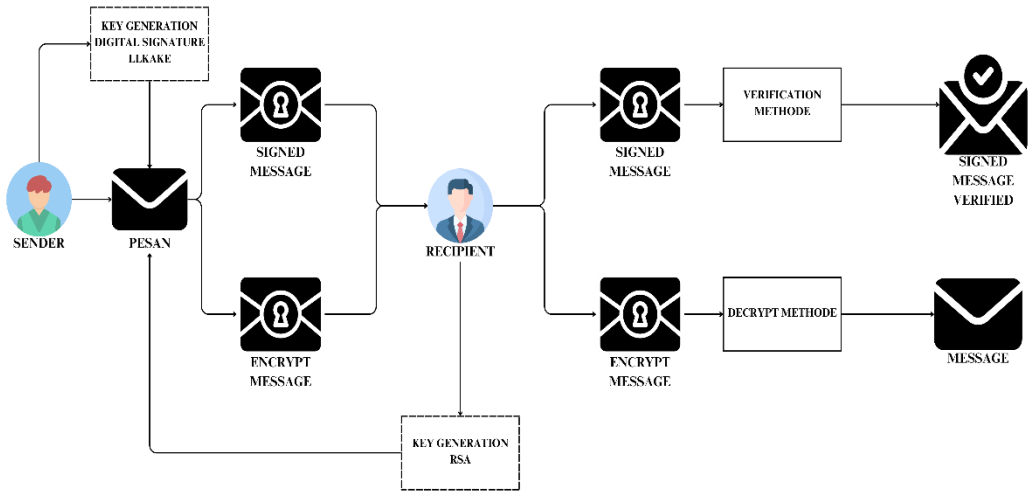
   Kebutuhan non-fungsional adalah spesifikasi tambahan yang mendukung sistem, dapat berupa kinerja, keamanan, serta batasan dari sistem. Penelitian ini

memiliki beberapa kebutuhan non-fungsional, yaitu:

a. Memiliki tombol kontrol yang akan melakukan pembangkitan bilangan random untuk memastikan jika bilangan random tersebut prima.
b. Memiliki tombol kontrol pada pemilihan file yang akan di enkripsi dan *signing*.
c. Menampilkan hasil pembangkitan kunci dan hasil enkripsi *plaintext* dalam bentuk file.txt.
d. Menampilkan hasil kembalian dari *ciphertext* yang telah dilakukan dekripsi menjadi file yang seharusnya.

3.2. Perancangan Sistem

Perancangan sistem dibangun berdasarkan analisis yang telah dilakukan terhadap penelitian. Peningkatan efektifitas dan efisiensi sistem menjadi tujuan utama perancangan ini, yang memuat spesifikasi lengkap dalam bentuk diagram, termasuk alur prosedur perancangan sistem pada program penelitian ini. penelitian ini menggunakan metode signcryption menggunakan algoritma LLKAKE dan RSA. Berikut representasi skema *signcyrption* yang lebih spesifik, yaitu LLKAKE dan RSA.



Gambar 3. 1 Diagram umum *signcryption*

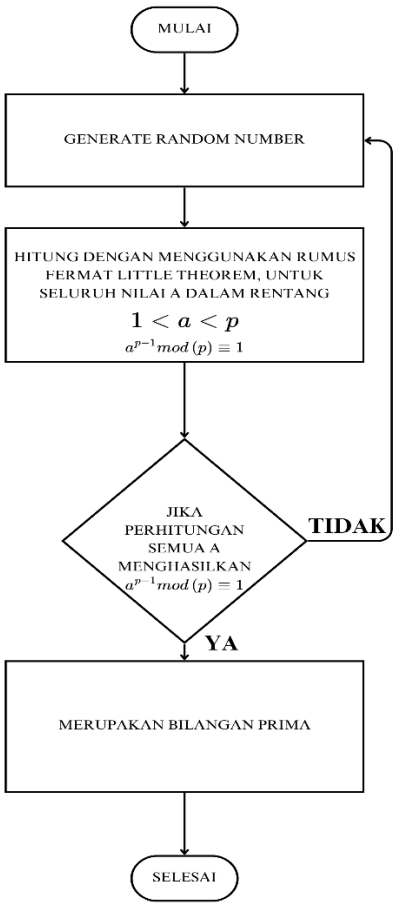
Berdasarkan gambar diatas diperlihatkan algoritma gabungan ini memiliki skema melakukan enkripsi dan *digital signature* secara bersamaan. Proses dilakukan oleh digital signature LLKAKE terlebih dahulu, mengubah pesan menjadi *hash message* menggunakan fungsi hash, dilanjutkan dengan melakukan *signing* pada *hash message.* Tahap selanjutnya adalah enkripsi pesan yang dilakukan oleh RSA, pesan yang telah di *signing* dan di enkripsi dikirm ke penerima, dan penerima melakukan verifikasi dan dekripsi.

3.3. *Flowchart* (Diagram Alir)

Menggunakan berbagai simbol yang ditentukan oleh American National Standards Institute (ANSI), termasuk panah, persegi panjang, segi enam, dan lain-lain, *flowchart* adalah representasi visual dari rangkaian tindakan, pilihan, dan perkembangan logis suatu sistem. Beberapa *flowchart* yang berkaitan dengan algoritma LLKAKE dan RSA disertakan dalam penelitian ini.

3.3.1 Flowchart *generate prime number fermat*

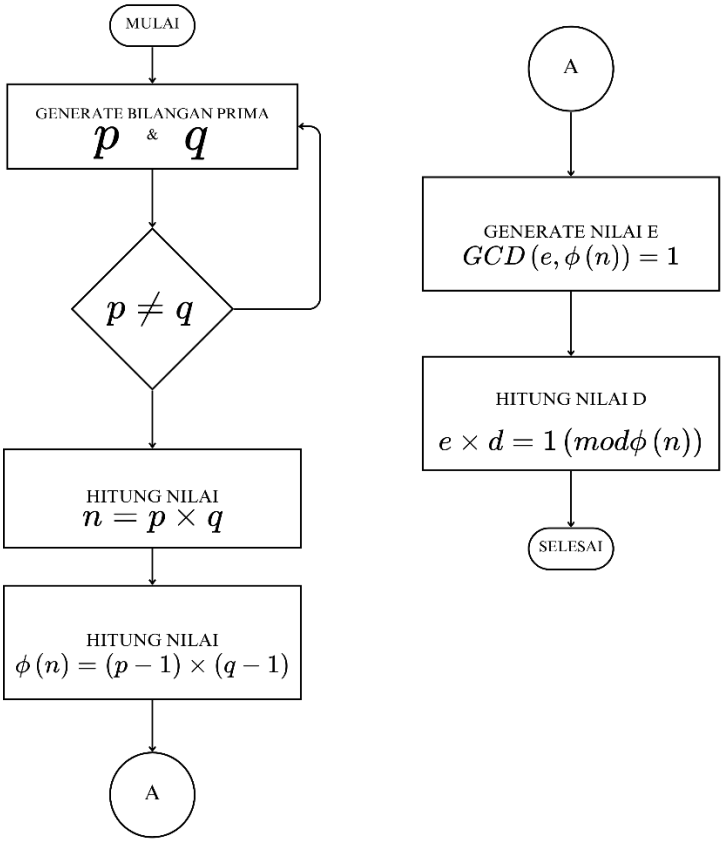
Merupakan diagram alir untuk membuat sebuah bilangan prima menggunakan *Fermat little theorem.*



Gambar 3. 2 Diagram *Fermat little theorem*

3.3.2 *Flowchart* key generation algoritma RSA

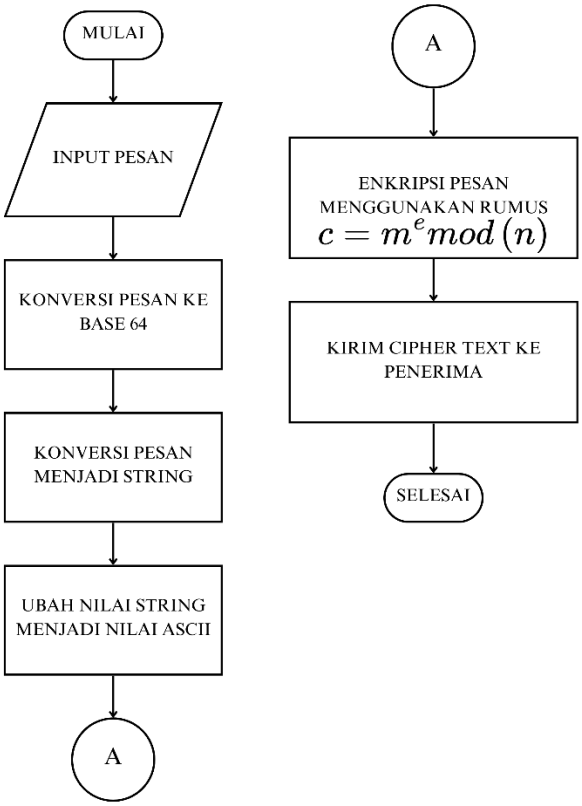
Flowchart key generation yang dilakukan oleh penerima pesan ditunjukkan pada Gambar 3.4. Proses pembangkitan kunci dimulai dengan syarat p dan q haruslah merupakan bilangan prima, dan juga p dan q tidak boleh memiliki nilai yang sama, selanjutnya hitung nilai n dan juga $\varphi(n)$. Setelah menghitung kedua nilai tersebut, bangkitkan nilai e yang memiliki syarat e harus lah bilangan ganjil, $1 < e < \varphi(n)$, dan $GCD(e, \varphi(n)) = 1$. Setelah berhasil membangkitkan nilai e dengan syarat tersebut, hitung nilai d. Dari perhitungan yang dilakukan maka didapatkan nilai d sebagai kunci privat dan nilai e dan nilai n sebagai kunci public.



Gambar 3. 3 Pembuatan kunci RSA

3.3.3 *Flowchart* enkripsi RSA

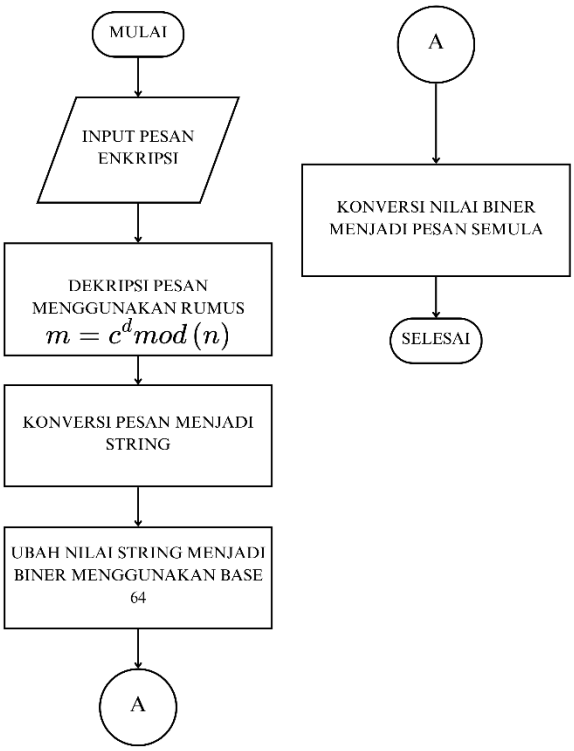
Flowchart enkripsi dilakukan oleh pengirim ditunjukkan pada Gambar 3.4, pengirim melakukan konversi terhadap pesan menjadi bilangan menggunakan ASCII, selanjutnya pengirim akan melakukan enkripsi.



Gambar 3. 4 Enkripsi RSA

3.3.4 *Flowchart* dekripsi RSA

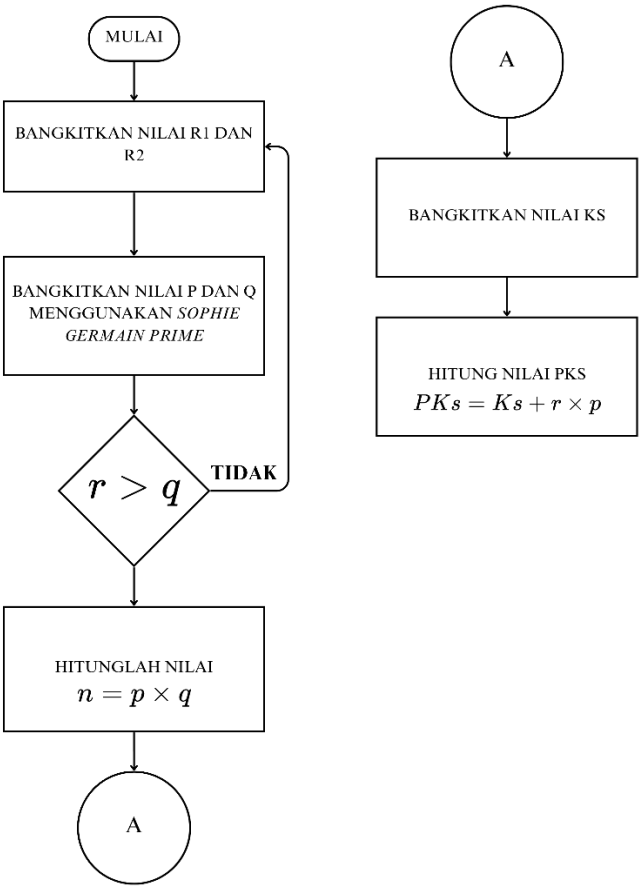
Flowchart dekripsi ini dilakukan oleh penerima yang ditunjukkan pada Gambar 3.5, penerima melakukan input pada file enkripsi, pada tahap selanjutnya penerima melakukan dekripsi dan mengubah hasil dekripsi menjadi bentuk string, berikutnya nilai string akan diubah menjadi nilai biner menggunakan *base64,* lalu nilai biner akan dikonversi menjadi file aslinya.



Gambar 3. 5 Dekripsi RSA

3.3.5 *Flowchart key generation* LLKAKE

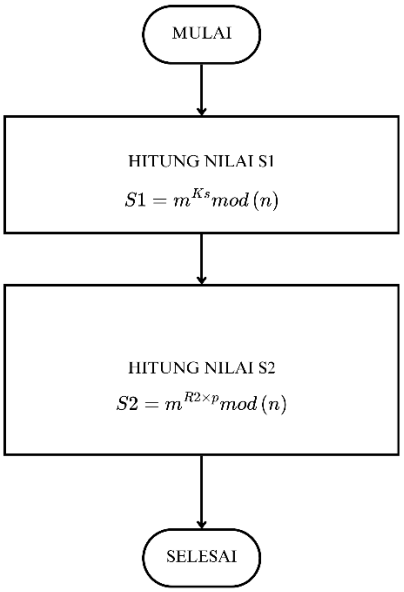
Flowchart key generation ini dilakukan oleh pengirim yang ditunjukkan pada Gambar 3.6, flowchart ini menjelaskan bagaimana proses pembangkitan kunci berjalan pada sistem yang akan dikembangkan.



Gambar 3. 6 Pembuatan kunci LLKAKE

3.3.6 *Flowchart* signing LLKAKE

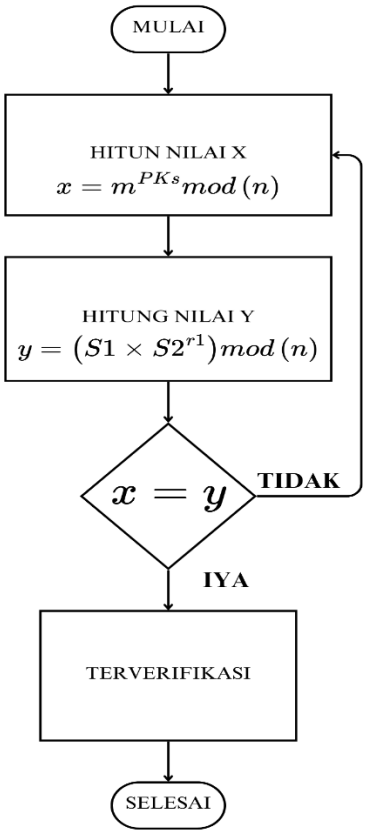
Flowchart signing ini dilakukan oleh pengirim yang ditunjukkan pada Gambar 3.7, *flowchart* ini menjelaskan bagaimana proses signing oleh pengirim.



Gambar 3. 7 Proses tanda tangan LLKAKE

3.3.7 *Flowchart* verifikasi algoritma LLKAKE

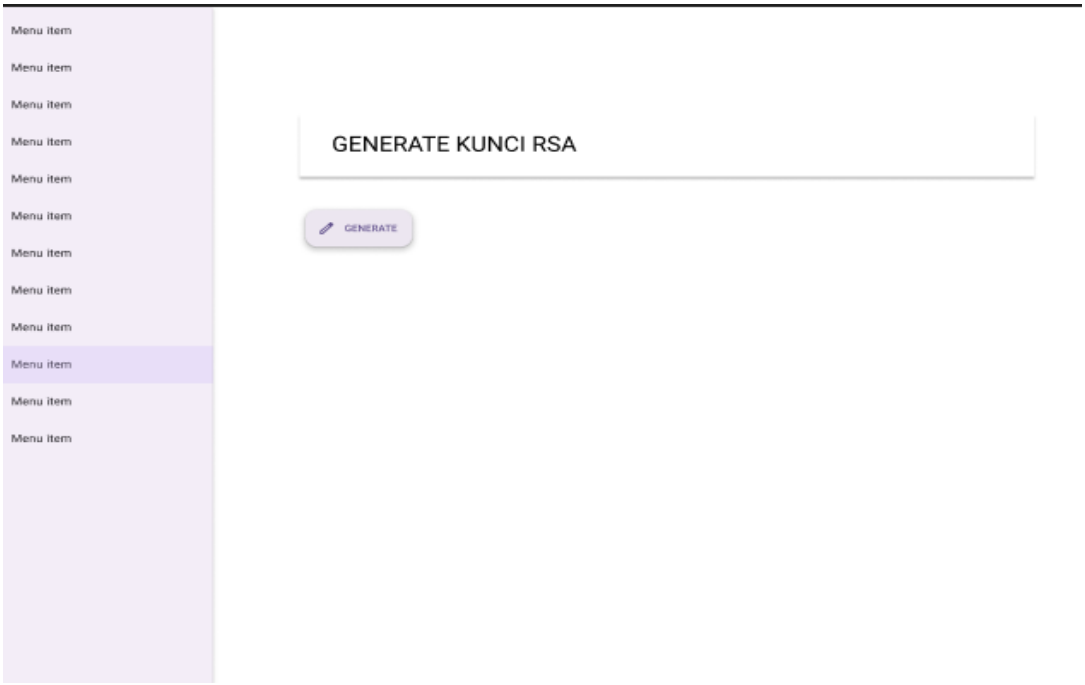
Flowchart ini dilakukan oleh penerima, pada *flowchart* ini menjelaskan bagaimana proses verifikasi oleh penerima.



Gambar 3. 8 Verifikasi menggunakan LLKAKE

3.4 Perancangan aplikasi

Merupakan alur untuk membentuk sebuah user interface pada aplikasi. Perancangan ini terdapat laman pembuatan kunci, laman enkripsi dan sign, laman dekripsi dan verifikasi.



Gambar 3. 9 Laman pembuatan kunci RSA

Gambar 3. 10 Laman pembuatan kunci llkake

Gambar 3. 11 Halaman tanda tangan dan enkripsi

Gambar 3. 12 Laman verifikasi dan dekripsi

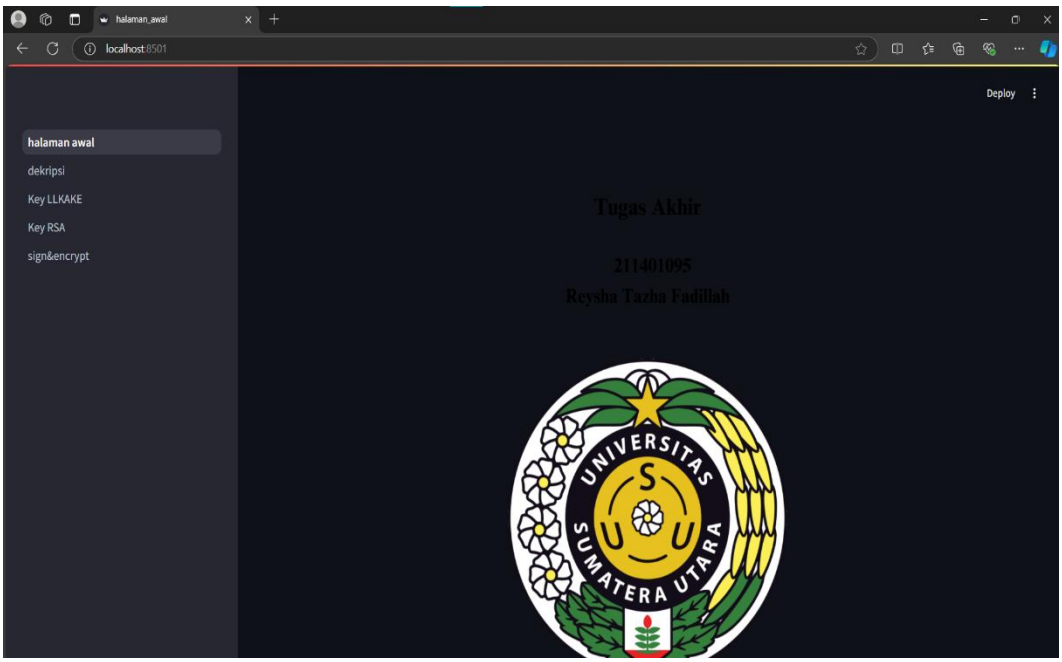
BAB 4
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Sistem yang dikembangkan dalam penelitian ini mengadopsi arsitektur berbasis Python-Streamlit-PyCharm. Sistem ini terdiri dari lima modul utama, yakni halaman beranda, modul generasi kunci RSA dan LLKAKE, serta modul tanda tangan, enkripsi, verifikasi, dan dekripsi.

4.1.1 Laman awal

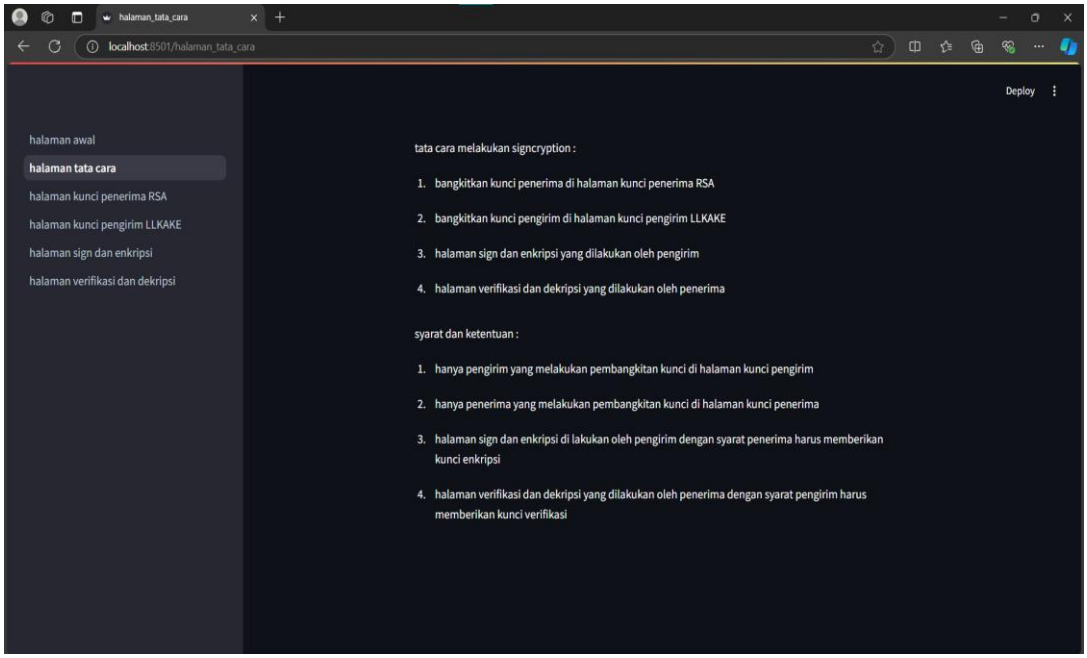
Laman awal merupakan halaman yang akan ditampilkan pertama kali ketika web dibuka. Halaman awal ditunjukkan pada Gambar 4.1.



Gambar 4. 1 Halaman awal

4.1.2 Laman Tata cara penggunaan aplikasi

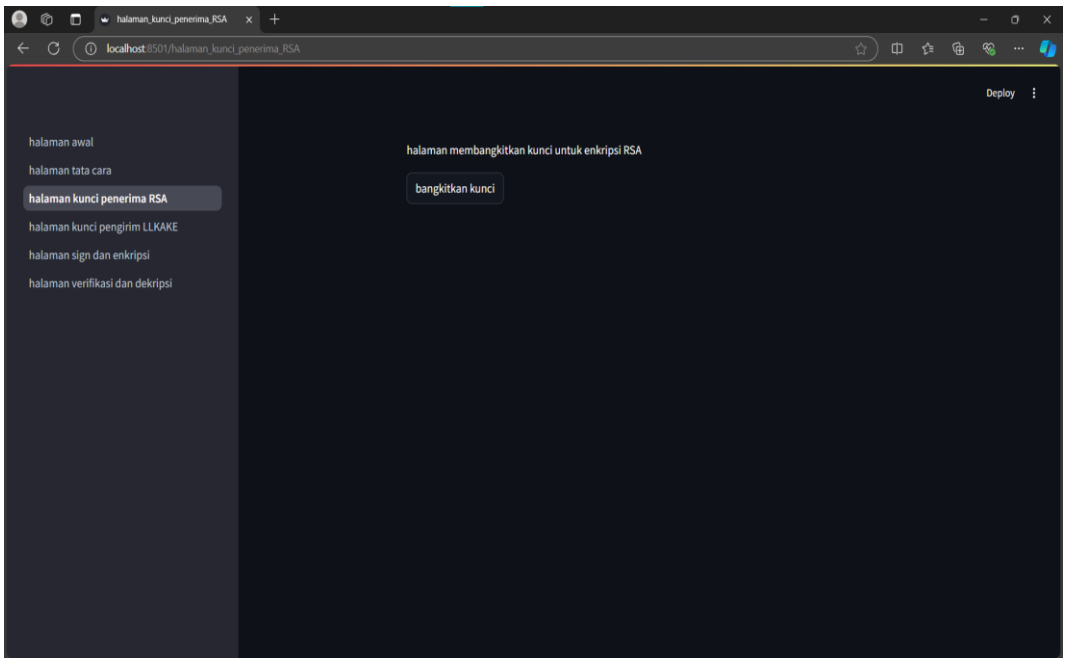
Merupakan laman pemberitahuan tata cara penggunaaan aplikasi.



Gambar 4. 2 Laman tata cara

4.1.3 Laman *key generation* RSA

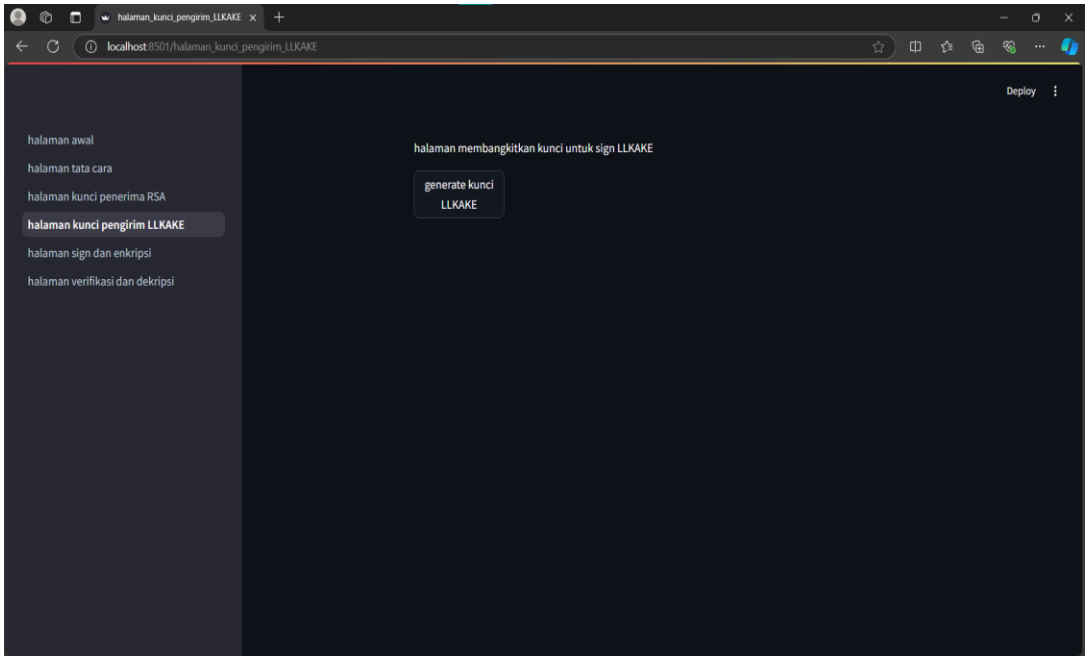
Laman ini memiliki fungsi untuk pembuatan kunci *encryption* dan *decryption.*



Gambar 4. 3 Laman pembuatan kunci RSA

4.1.4 Laman *key generation* LLKAKE

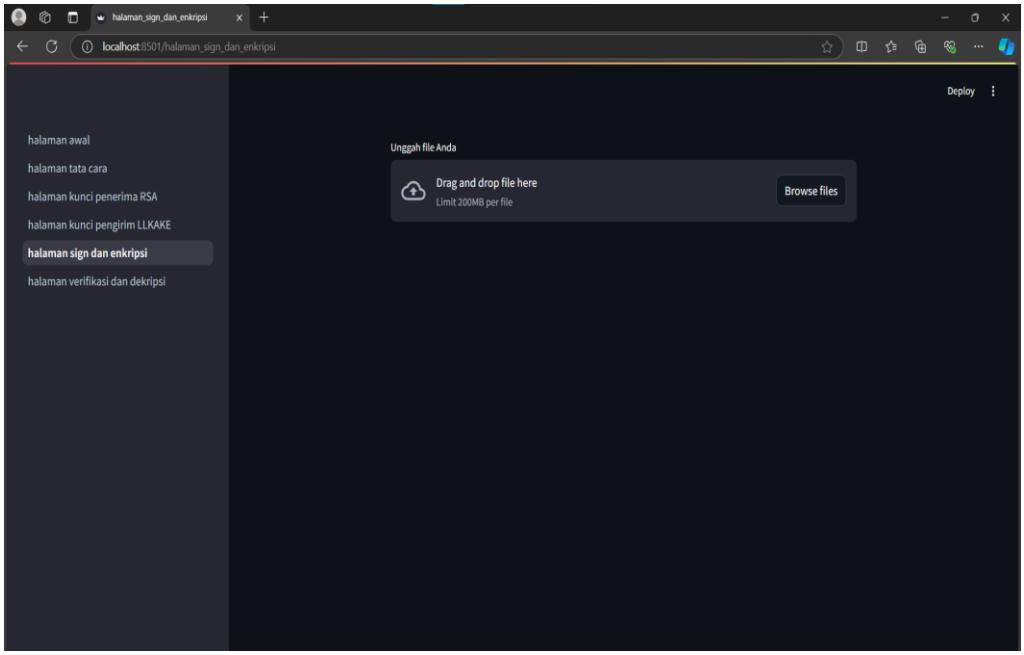
Merupakan laman pembangkit kunci untuk sign dan verification pada LLKAKE.



Gambar 4. 4 Pembuatan kunci *sign* dan verifikasi

4.1.5 Laman sign dan encryption

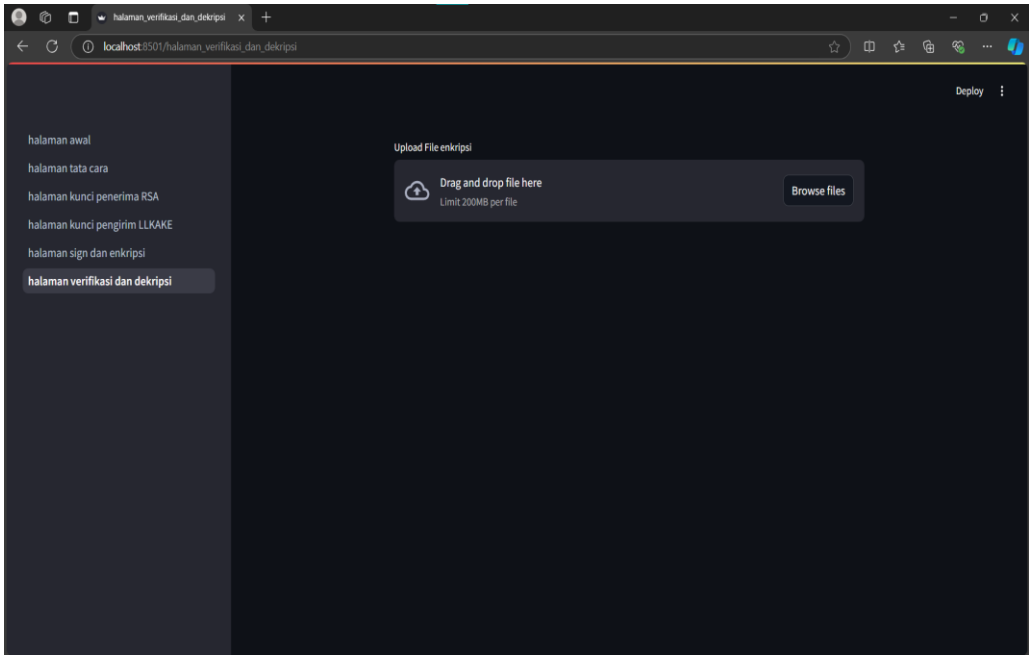
Merupakan laman melakukan sign dan encryption.



Gambar 4. 5 laman *sign* dan enkripsi

4.1.6 Laman *verification* dan *decryption*

Merupakan laman untuk melakukan *verification* dan *decryption.*



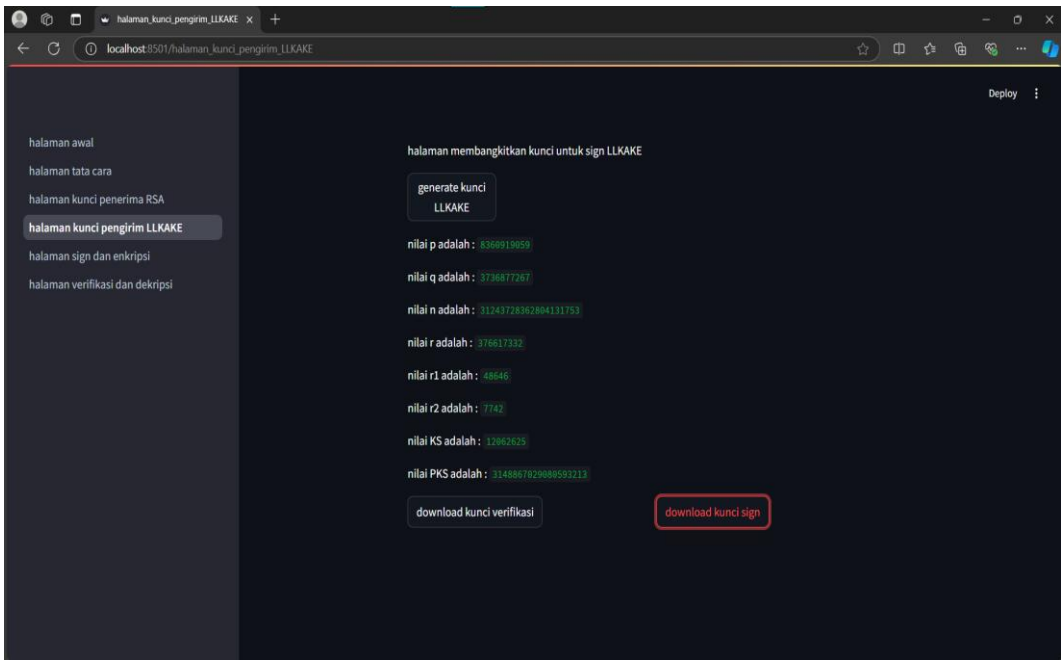
Gambar 4. 6 laman verifikasi dan dekripsi

4.2 Pengunjian Sistem

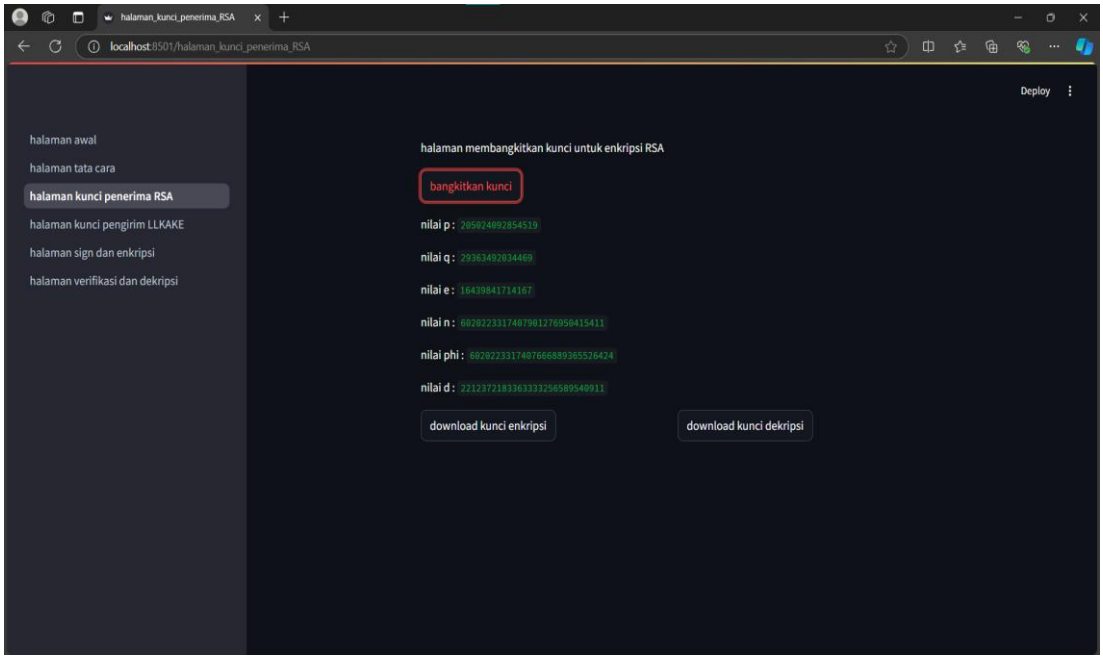
Sistem yang berhasil dikembangkan akan diuji guna memastikan bahwa sistem dapat melakukan teknik signcryption, serta mengembalikan file ke bentuk aslinya dan membuktikan kepemilikan file tersebut.

4.2.1 Pengujian Key generation

Melakukan pengujian terhadap laman key generation.



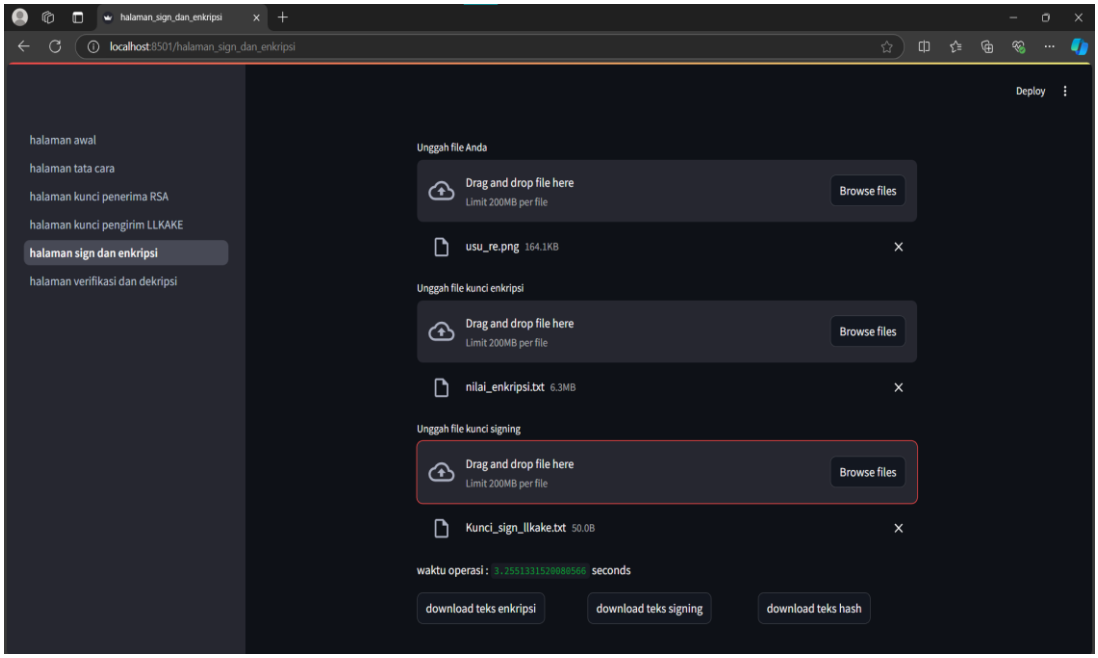
Gambar 4. 7 Pengujian pembuatan kunci LLKAKE



Gambar 4. 8 Pengujian pembuatan kunci RSA

4.2.2 Proses uji coba *sign* dan *encryption*

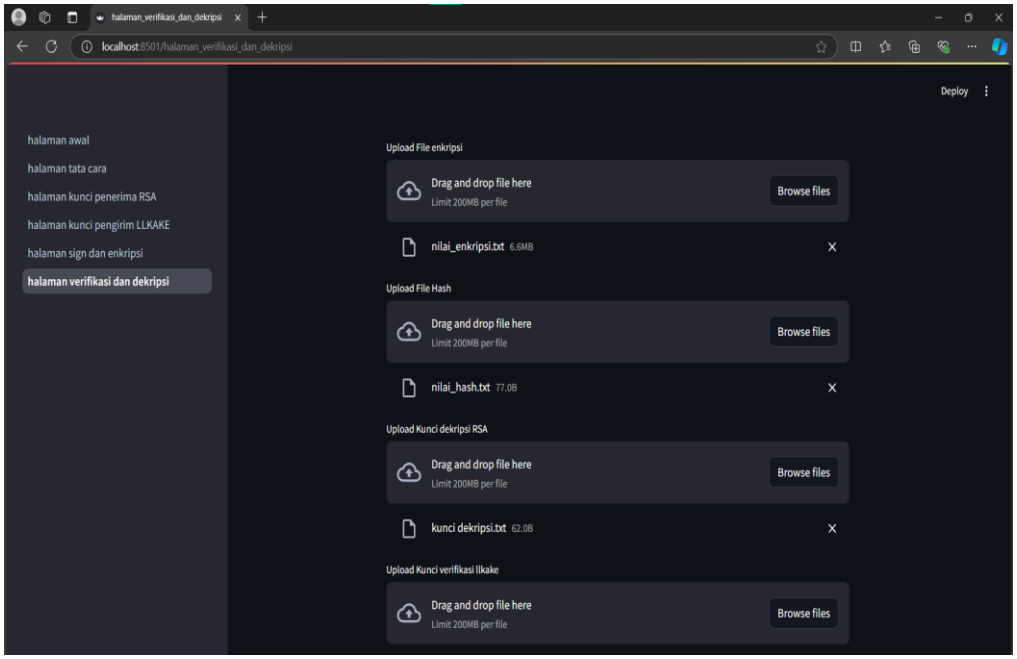
Melakukan tahap uji coba pada metode sign dan encryption.



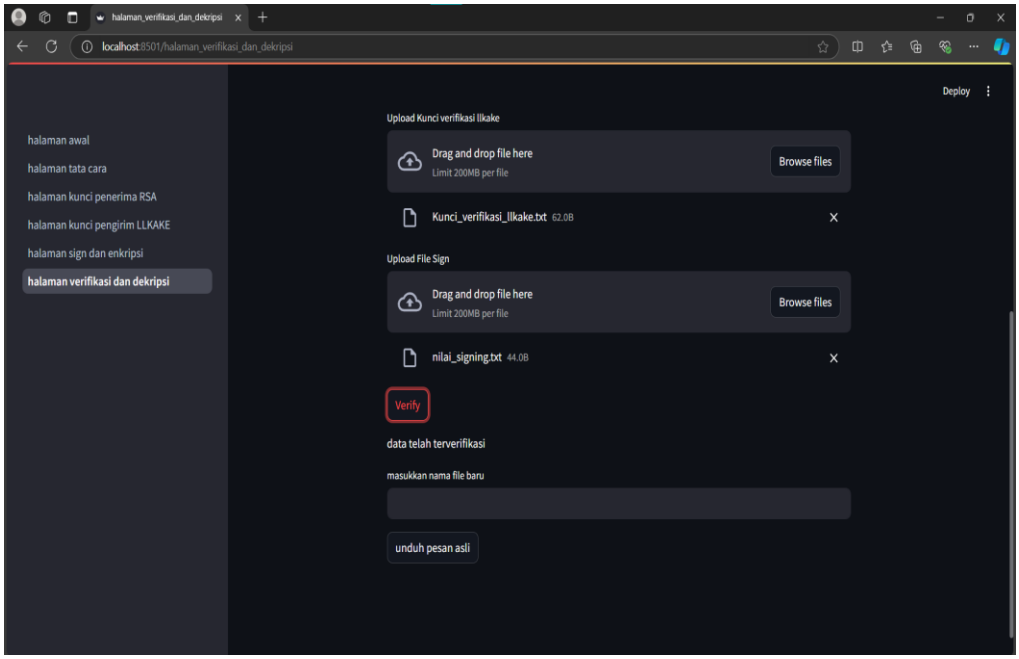
Gambar 4. 9 Pengujian tanda tangan dan enkripsi

4.2.3 Proses uji coba *verification* dan *decryption*

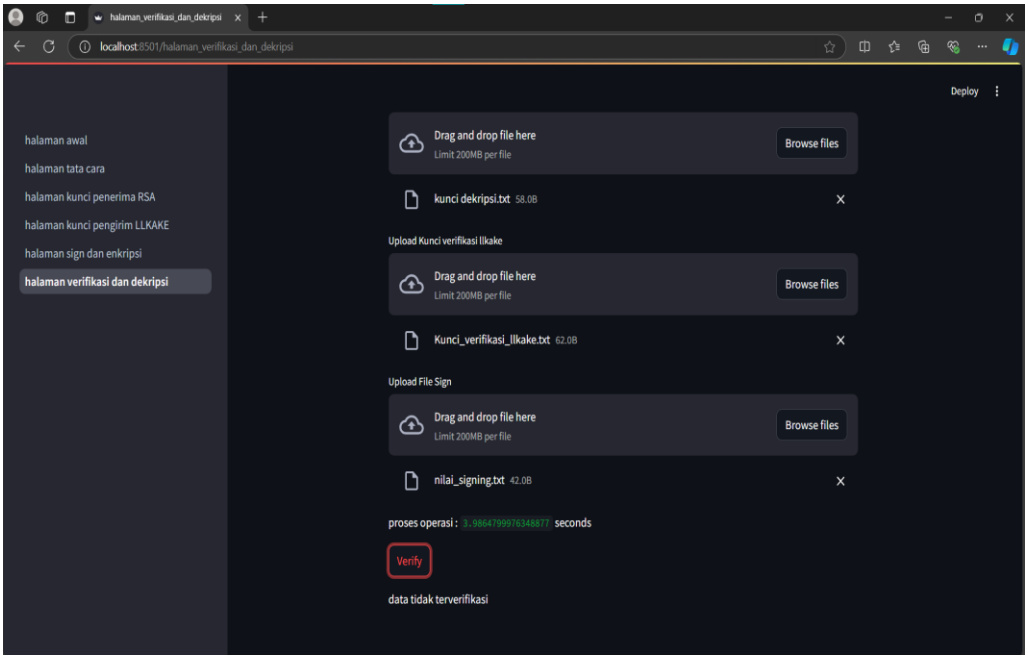
Melakukan uji coba terhadap metode verification and decryption.



Gambar 4. 10 Uji coba verifikasi dan dekripsi



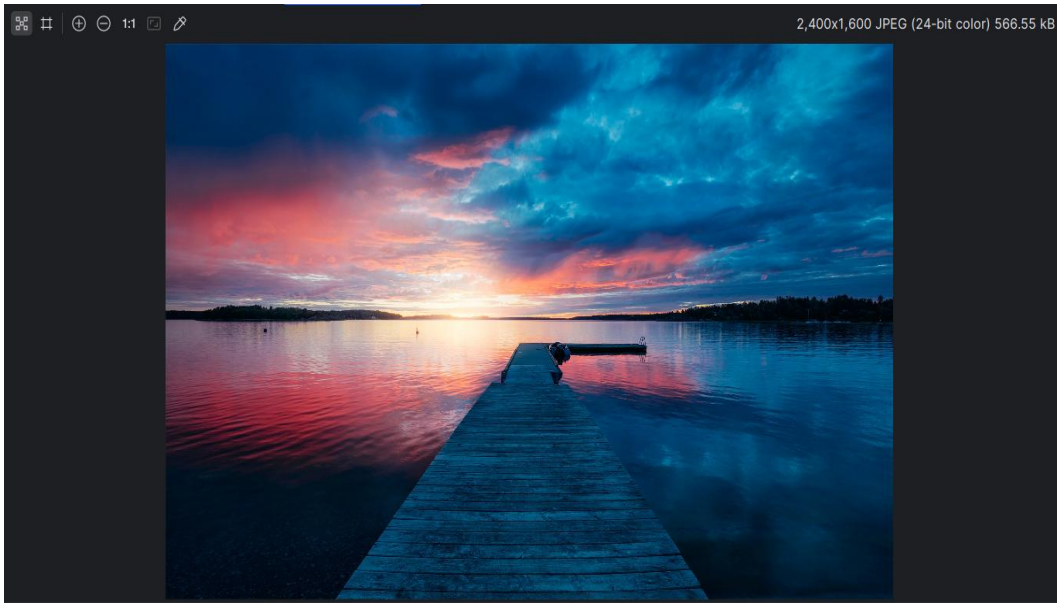
Gambar 4. 11 Hasil uji coba verifikasi dan dekripsi



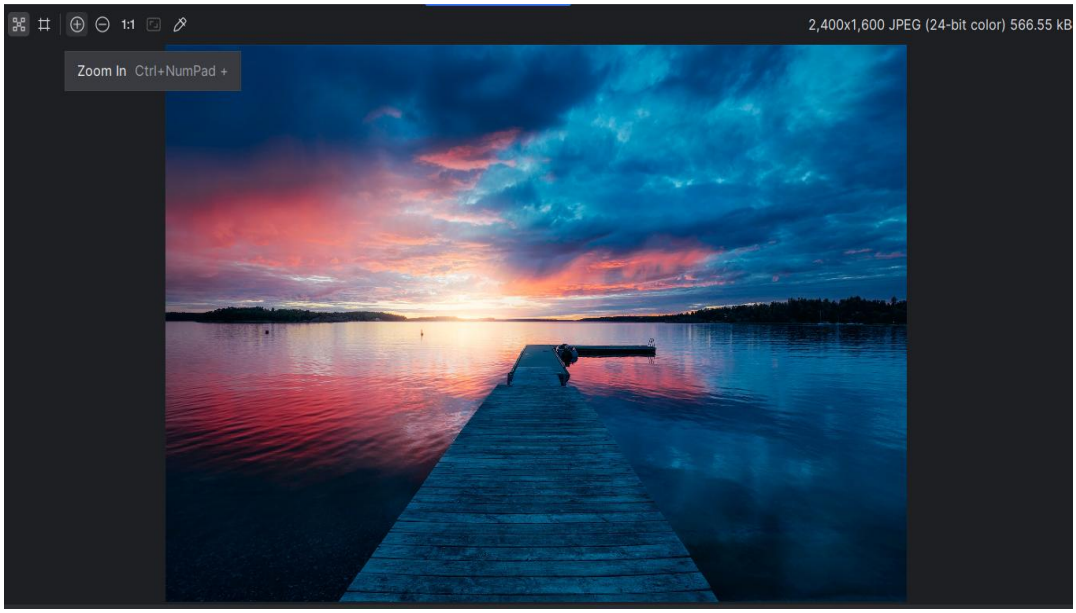
Gambar 4. 12 Hasil uji coba gagal

4.2.4 Hasil uji coba dokumen

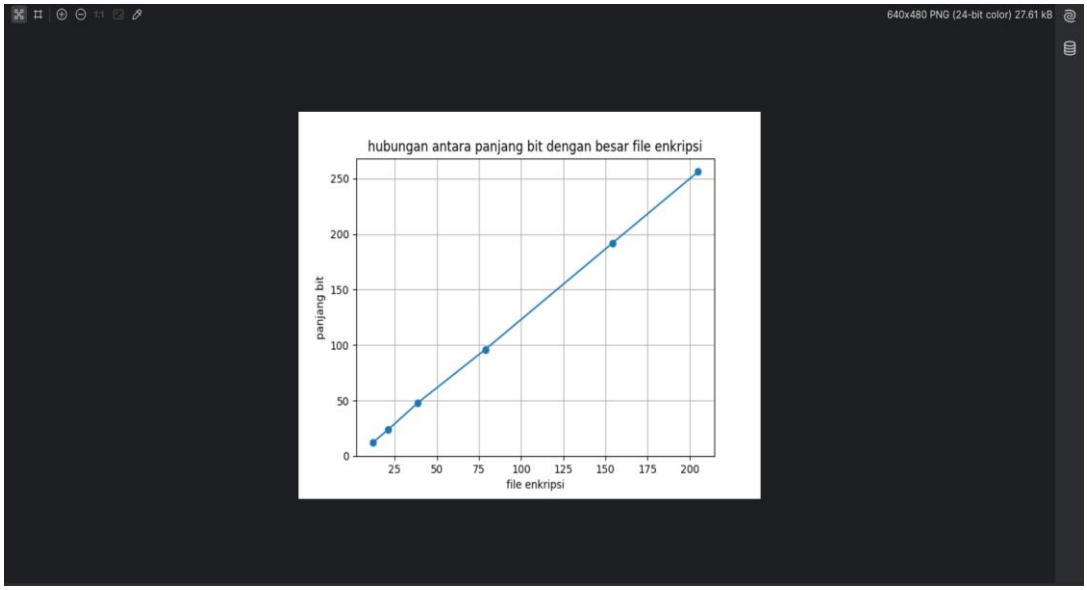
Memberikan hasil dokumen asli dengan dokumen yang sudah di dekripsi.



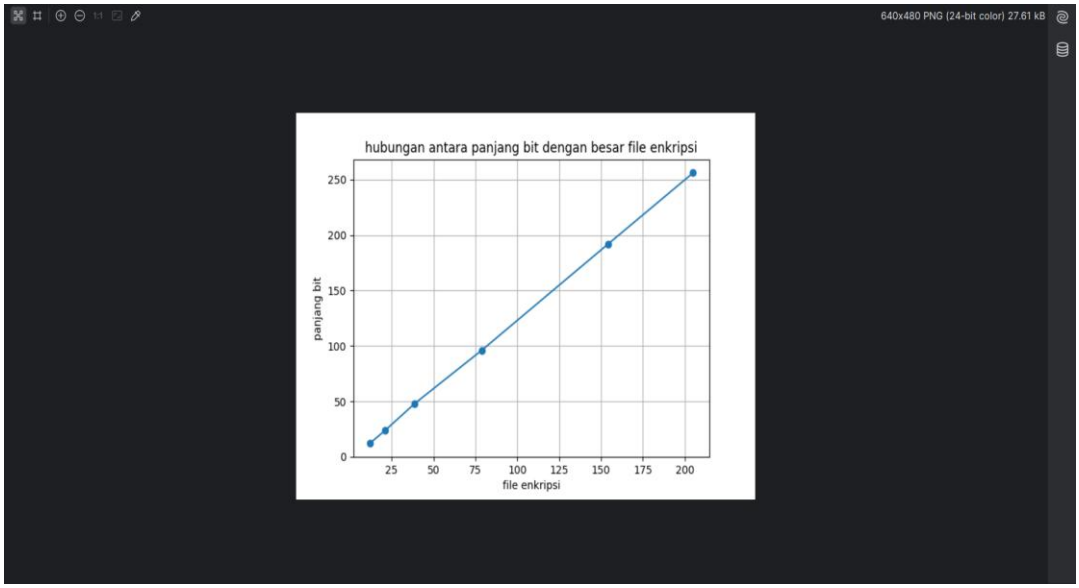
Gambar 4. 13 dokumen asli



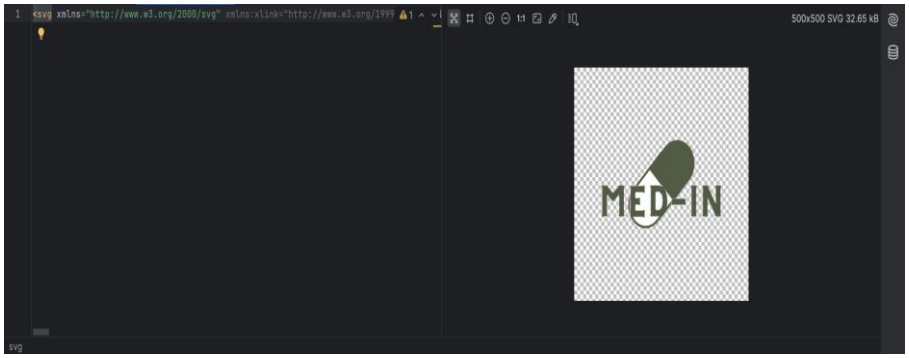
Gambar 4. 14 hasil dokumen dekripsi



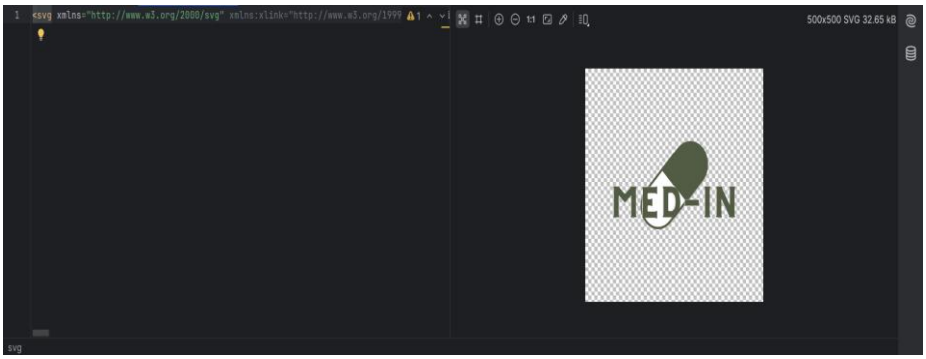
Gambar 4. 15 Gambar sebelum di enkripsi dalam format PNG



Gambar 4. 16 Gambar dokumen setelah melakukan proses dekripsi



Gambar 4. 17 Gambar sebelum di enkripsi dalam forma SVG



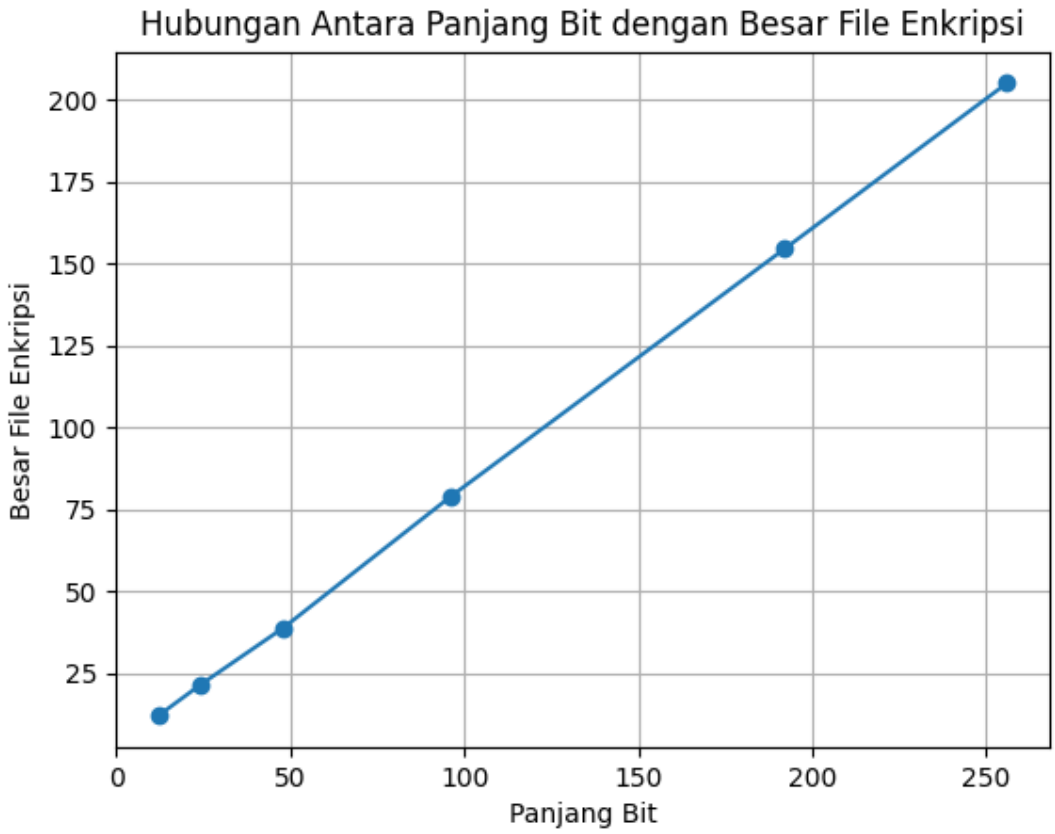
Gambar 4. 18 Gambar sesudah di dekripsi dengan format SVG

4.2.5 Aspek *ratio* dokumen

Melakukan perbandingan terhadap dokumen awal dengan dokumen yang telah di enkripsi.

Tabel 4. 1 perbandingan dokumen enkripsi

Dokumen awal	Panjang bit kunci	Dokumen setelah di enkripsi	Aspek rasio
160 KB	12	1,9 MB	11,875
160 KB	24	3,4 MB	21,25
160 KB	48	6,2 MB	38,75
160 KB	96	12,6 MB	78,75
160 KB	192	24,7 MB	154,375
160 KB	256	32,8 MB	205



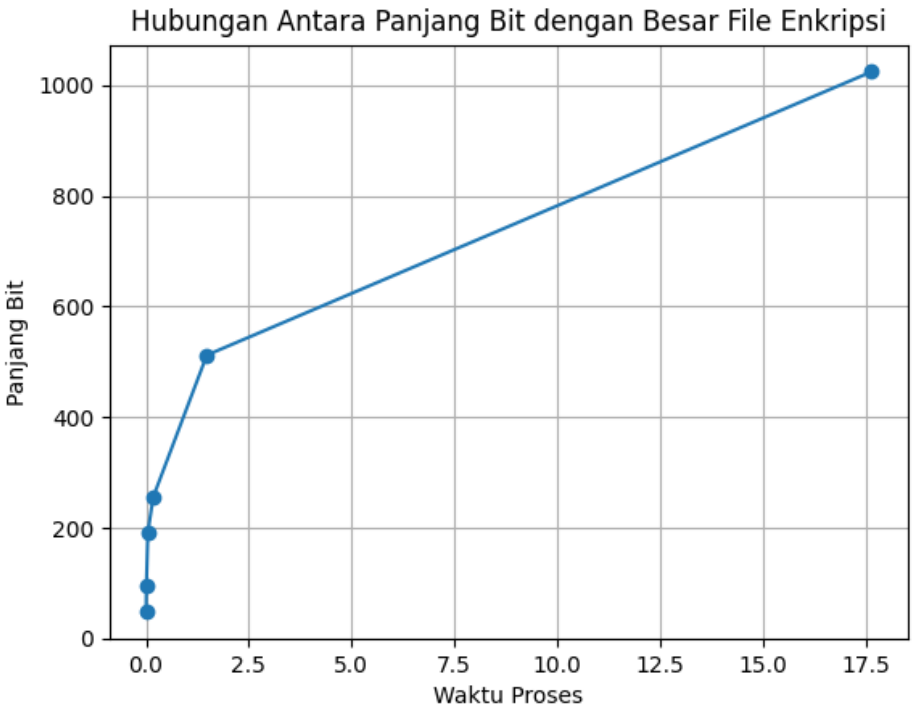
Gambar 4. 19 Grafik perbandingan kunci dengan file enkripsi

4.2.6 Proses waktu pembangkitan kunci

Melakukan perbandingan waktu pembangkitan kunci dengan panjang kunci.

Tabel 4. 2 Proses pembangkitan kunci RSA

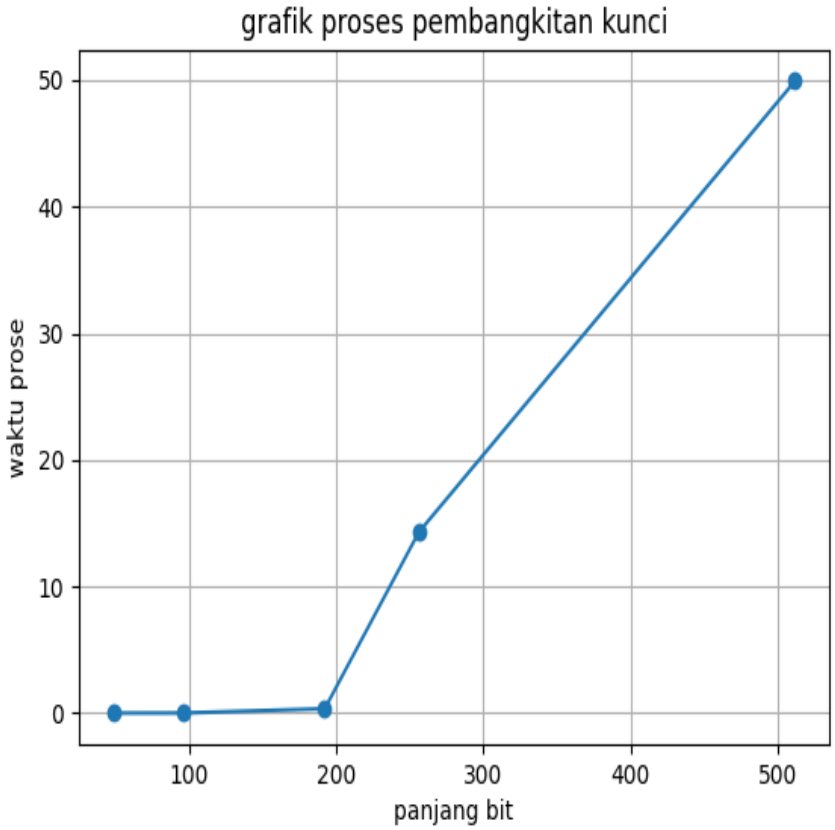
Panjang kunci (bit)	Panjang kunci enkrispi	Panjang kunci dekripsi	Waktu pembangkitan kunci (detik)
48	48	96	0.0010154247283935547
96	96	192	0.009267091751098633
192	192	384	0.045647382736206055
256	256	512	0.17699956893920898
512	512	1024	1.4790003299713135
1024	1024	2048	17.64358425140381



Gambar 4. 20 Grafik proses pembuatan kunci RSA

Tabel 4. 3 Pembangkitan kunci LLKAKE

Pembangkitan kunci LLKAKE (bit)	Waktu proses (detik)
48	0.009058475494384766
96	0.01399993896484375
192	0.3523836135864258
256	14.290706396102905
512	49.92369055747986



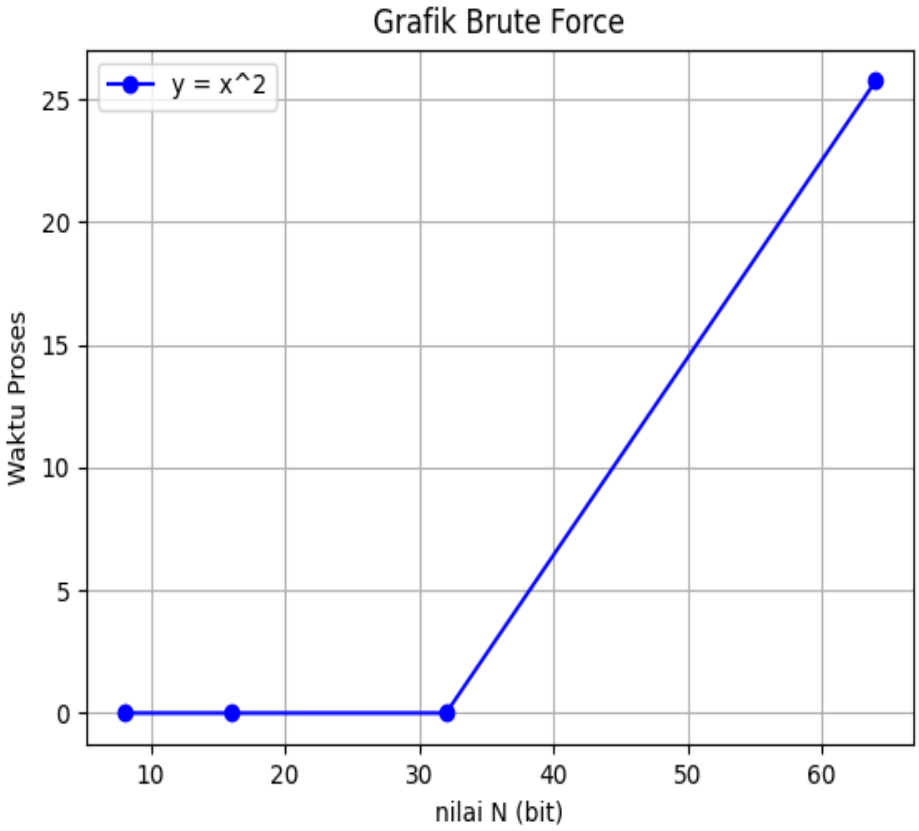
Gambar 4. 21 grafik pembangkitan kunci LLKAKE

4.2.7 Analisis penyerangan kunci menggunakan *Brute Force* dan faktorisasi *fermat*

Pada bagian ini peneliti akan melakukan analisis penyerangan kunci dengan menggunakan *Brute Force* dan juga faktorisasi *fermat.*

Tabel 4. 4 Tabel proses *Brute Force*

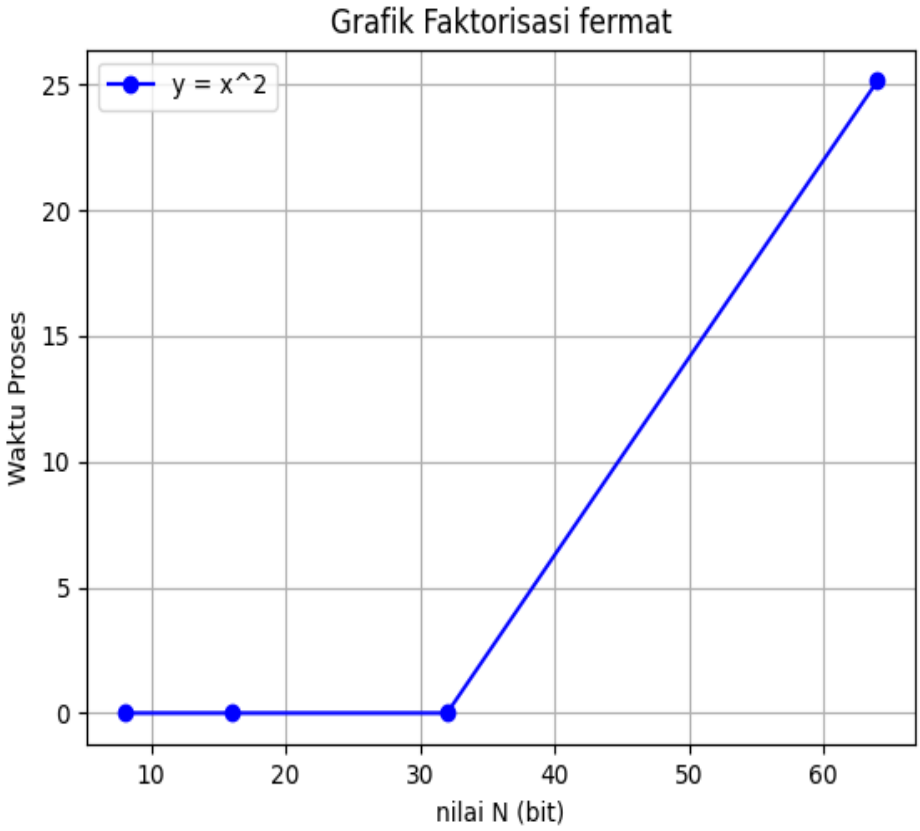
Nilai n	Nilai p	Nilai q	Waktu proses *Brute Force (detik)*
8*bit*	4*bit*	4*bit*	0.00003266334533691406
16*bit*	8*bit*	8*bit*	0.00003910064697265625
32*bit*	*16bit*	*16bit*	0.0007946491241455078
64bit	*32bit*	*32bit*	25.73798966407776



Gambar 4. 22 Grafik proses *Brute Force*

Tabel 4. 5 Tabel prose Faktorisasi *Fermat*

Nilai n	Nilai p	Nilai q	Waktu proses Faktorisasi *Fermat(detik)*
8*bit*	4*bit*	4*bit*	0.000009059906005859375
16*bit*	8*bit*	8*bit*	0.000011444091796875
32*bit*	*16bit*	*16bit*	0.00002956390380859375
64bit	*32bit*	*32bit*	25.14949083328247



Gambar 4. 23 Grafik Faktorisasi *Fermat*

Dapat disimpulkan bahwa algoritma Faktorisasi *Fermat* lebih cepat melakukan penyerangan dari pada algoritma *Brute Force*.

BAB 5
PENUTUP

5.1 Kesimpulan

Setelah tahap implementasi sudah dilaksanakan pada proses penelitian *signcryption* menggunakan algoritma LLKAKE dan RSA dan pengujian yang dilakukan pada web, disimpulkan bahwa:

1. Algoritma RSA terbukti dapat melakukan proses kriptografi, yaitu *key generation*, enkripsi, dan dekripsi dan terbukti dapat melakukan proses enkripsi pada file.
2. Algoritma LLKAKE terbukti dapat melakukan proses *digital signature*, yaitu *key generation, signing, verificartion.*
3. Algoritma RSA memiliki waktu komputasi yang terbilang cukup lama dibanding dengan kriptografi simetris, tetapi RSA memiliki efektivitas yang tinggi dalam mengamankan sebuah dokumen digital, yang dapat membuat pihak asing tidak dapat membaca dokumen tersebut.
4. Pembangkitan p dan q terbilang cukup kompleks sehingga membuat proses komputasi menjadi jauh lebih lama seiring dengan bertambah nya jumlah bit pada setiap variabel.
5. Algoritma LLKAKE memiliki pembangkitan nilai kunci publik yang aman dengan cara menambahkan sebuah variabel jebakan pada perhitungannya, yaitu variabel r.
6. Algoritma LLKAKE dan RSA memiliki kelemahan pada pembangkitan kunci yang harus menggunakan bilangan prima, menggunakan bilangan prima tersebut dapat dikatakan aman dan juga sekaligus kompleks yang membuat komputasi akan melambat.

5.2 Saran

1. Untuk menutupi kelemahan algoritma RSA pada proses enkripsi, pengelompokkan beberapa karakter menjadi satu buah blok. Gabungkan setiap kode ASCII setiap karakter yang ada dalam blok.
2. Disarankan untuk melakukan studi lebih lanjut guna memperoleh pemahaman lebih mendalam menyebabkan lambatnya waktu eksekusi untuk uji keprimaan

suatu angka dengan menggunakan algoritma *fermat little theorem*.

3. Disarankan untuk melakukan studi lebih lanjut untuk memperoleh pemahaman lebih mendalam menyebabkan lambatnya waktu pembangkitan kunci algoritma LLKAKE.
4. Karena proses *signcryption* dilakukan secara bersamaan maka pembuatan kunci harus diperkecil untuk mempermudah proses komputasi.

DAFTAR PUSTAKA

Lalem, F., Laouid, A., Kara, M., Al-Khalidi, M., & Eleyan, A. (2023). A novel digital signature scheme for advanced asymmetric encryption techniques. *Applied Sciences*, *13*(8), 5172.

Mohamad, M. S. A., Din, R., & Ahmad, J. I. (2021). Research trends review on RSA scheme of asymmetric cryptography techniques. *Bulletin of Electrical Engineering and Informatics*, *10*(1), 487-492.

Arief, A., & Saputra, R. (2016). Implementasi Kriptografi Kunci Publik dengan Algoritma RSA-CRT pada Aplikasi Instant Messaging. *Scientific Journal of Informatics*, *3*(1), 46-54.

Ginting, C. L., Budiman, M. A., & Nasution, S. (2024). Signcryption with Matrix Modification of RSA Digital Signature Scheme and Cayley-Purser Algorithm. *Data Science: Journal of Computing and Applied Informatics*, *8*(1), 14-24.

Wang, H. (2023). The Application of Fermat's Little Theorem in Cryptography. *Science and Technology of Engineering, Chemistry and Environmental Protection*, *1*(4).

Ramesh, V. P., & Makeshwari, M. (2022). A prime primitive root p of 2 p+ 1 is a Sophie Germain prime. *The American Mathematical Monthly*, *129*(6), 538-538.

Dubner, H. (1996). Large Sophie Germain primes. *Mathematics of computation*, *65*(213), 393-396.

Iliev, A., & Kyurkchiev, N. (2018, November). The faster extended Euclidean algorithm. In *Collection of scientific works from conference* (pp. 21-26).

Diko, E., & Ibraimi, M. (2023). RSA & EXTENDED EUCLIDEAN ALGORITHM WITH EXAMPLES OF EXPONENTIAL RSA CIPHERS, RSA EXAMPLE SOLUTION WITH EXTENDED EUCLIDEAN ALGORITHM. *International Scientific Journal*

Vision, *8*(1), 161-175.

Poulakis, D. (2020). An application of Euclidean algorithm in cryptanalysis of RSA. *Elemente Der Mathematik*, *75*(3), 114-120.

Zhou, J., Hu, J., & Chen, P. (2010, December). Extended Euclid algorithm and its application in RSA. In *The 2nd International Conference on Information Science and Engineering* (pp. 2079-2081). IEEE.

Luo, Z. J., Liu, R., & Mehta, A. (2023). Understanding the RSA algorithm. *arXiv preprint arXiv:2308.02785*.

Malone-Lee, J. (2002). Identity-based signcryption. *Cryptology ePrint Archive*.

Bao, F., & Deng, R. H. (1998, February). A signcryption scheme with signature directly verifiable by public key. In *International workshop on public key cryptography* (pp. 55-59). Berlin, Heidelberg: Springer Berlin Heidelberg.