p-ISSN : 2406-7733
e-ISSN : 2597-9922

PERANCANGAN SISTEM PENGAMANAN DOKUMEN *PDF* MENGGUNAKAN KOMBINASI ALGORITMA *DIFFIE-HELLMAN* DAN *RSA*

M. Ridho Saputra[1], Haida Dafitri[2], Siti Sundari[3]
Teknik Infomatika,Teknik dan Komputer, Universitas Harapan Medan
Jl. Rawa I Lr. Sedar IV, Medan
E-mail: pridho396@gmail.com[1], aida.stth@gmail.com[2], sundaristth@gmail.com

Abstrak - Keamanan data pada dokumen digital merupakan aspek penting, terutama pada dokumen *PDF* yang sering digunakan dalam berbagai transaksi bisnis dan pertukaran informasi sensitif. Penelitian ini bertujuan untuk merancang dan mengembangkan aplikasi pengamanan data pada dokumen *PDF* dengan menggunakan kombinasi algoritma *Diffie-Hellman* dan *RSA*. *Diffie-Hellman* digunakan untuk pertukaran kunci secara aman antara pengirim dan penerima, sedangkan algoritma *RSA* berperan dalam proses enkripsi dan dekripsi data. Penggunaan kombinasi kedua algoritma ini diharapkan mampu meningkatkan keamanan dengan meminimalisir risiko serangan pihak ketiga selama proses transmisi data. Pengujian dilakukan untuk memastikan tingkat keamanan, kecepatan enkripsi, serta efisiensi penggunaan sumber daya pada aplikasi. Hasil pengujian menunjukkan bahwa aplikasi ini mampu melindungi data dengan baik dan tetap efisien dalam penggunaannya. Dengan demikian, aplikasi ini dapat diimplementasikan sebagai solusi pengamanan data pada dokumen *PDF* untuk berbagai kebutuhan.

Kata Kunci: *Diffie-Hellman*, Enkripsi, Keamanan Data, *PDF*, *RSA*.

I. PENDAHULUAN

Dalam rangka meningkatkan keamanan dokumen *PDF*, pendekatan kriptografi menjadi sangat penting (Cheng & Chen, 2024). Salah satu algoritma kriptografi yang telah terbukti efektif adalah algoritma *Diffie-Hellman*, yang dikenal sebagai algoritma pertukaran kunci publik yang aman (Liander, 2022). Dengan menggunakan algoritma ini, dua entitas dapat secara aman menetapkan kunci bersama tanpa harus mengirimkan kunci rahasia secara langsung melalui kanal komunikasi yang tidak aman. Selain itu, algoritma *RSA* (Rivest et al., 2023) juga terkenal dalam dunia kriptografi karena kemampuannya untuk mengenkripsi dan mendekripsi data menggunakan pasangan kunci publik dan pribadi (Miller, 2023).

Penerapan kombinasi algoritma *Diffie-Hellman* dan *RSA* dalam konteks pengamanan dokumen *PDF* masih memerlukan penelitian lebih lanjut (Victor et al., 2023). Dengan menggunakan *Diffie-Hellman* untuk pertukaran kunci yang aman dan *RSA* untuk enkripsi data, dua lapisan keamanan dapat diterapkan untuk melindungi dokumen *PDF* dari ancaman. *RSA* menyediakan keamanan tambahan dengan memastikan bahwa data yang diterima dan dikirim tetap rahasia dan hanya dapat diakses oleh pihak yang berwenang.

Mengamankan dokumen *PDF* dengan menggunakan kombinasi algoritma *Diffie-Hellman* dan *RSA* dapat memberikan lapisan keamanan tambahan yang diperlukan untuk melindungi informasi sensitif (Patgiri, 2021). Oleh karena itu, penelitian ini bertujuan untuk mengembangkan sebuah aplikasi pengamanan data yang menggunakan kombinasi algoritma *Diffie-Hellman* dan *RSA* khususnya untuk dokumen *PDF*. Dengan menyediakan alat yang mudah digunakan untuk mengenkripsi dokumen *PDF*, diharapkan dapat meningkatkan kesadaran dan praktik keamanan data di kalangan pengguna (Nova Situmoran et al., 2023). Selain itu, aplikasi ini juga diharapkan dapat memberikan solusi yang efektif dalam melindungi dokumen *PDF* dari ancaman keamanan digital yang semakin kompleks.

Dengan demikian, penelitian ini tidak hanya akan memberikan kontribusi terhadap pemahaman kita tentang keamanan data dan kriptografi dalam konteks dokumen *PDF*, tetapi juga akan memberikan solusi praktis yang dapat membantu meningkatkan keamanan informasi sensitif di lingkungan digital (Victor et al., 2023).

II. TINJAUAN PUSTAKA

A. Sistem Keamanan Dokumen

Masalah keamanan data dan kerahasiaan data merupakan salah satu aspek penting dari suatu sistem informasi. Dalam hal ini, sangat terkait dengan betapa pentingnya informasi tersebut dikirim dan diterima oleh orang yang berkepentingan. Informasi akan tidak berguna lagi apabila di tengah jalan informasi itu disadap atau dibajak oleh orang yang tidak berhak (Singh & Ruj, 2020).

B. Kriptografi

Kriptografi merupakan kumpulan teknik untuk menyandikan data dan pesan sedemikian, sehingga data dan pesan tersebut dapat disimpan dan ditransmisikan dengan aman. Berikut ini beberapa

p-ISSN : 2406-7733
e-ISSN : 2597-9922

terminologi dasar dari kriptografi serta hal-hal yang berkaitan dengan terminologi tersebut (Gunawan, 2019), antara lain :

1. Kriptografi dapat digunakan untuk meningkatkan keamanan komunikasi meskipun komunikasi tersebut dilakukan dengan media komunikasi yang sangat tidak aman (misalnya internet). Juga dapat menggunakan kriptografi untuk melakukan enkripsi berkas-berkas sensitif, sehingga orang lain tidak dapat mengartikan data-data yang ada
2. Kriptografi dapat digunakan untuk memberikan jaminan integritas data serta menjaga kerahasiaan.
3. Dengan menggunakan kriptografi, maka sangat mungkin untuk memferivikasi asal data dan pesan yang ada menggunakan digital signature.
4. Pada saat menggunakan metoda kriptografi, hanya kunci sesi yang harus tetap dijaga kerahasiannya. Algoritma, ukuran kunci dan format berkas dapat dibaca oleh siapapun tanpa mempengaruhi keamanan.

C. Enkripsi dan Dekripsi

Pengguna dari enkripsi/dekripsi ini umurnya sama tua dengan pengembangan komunikasi pada awalnya. Pada masa perang, cipher sering disalah artikan dengan sandi, digunakan untuk menjaga musuh mencuri isi dari hubungan transmisi (secara teknik, sandi menggambarkan signal tanpa maksud tujuan untuk menyimpan, seperti sandi morse dan ASCII) (Desai & Choksi, 2020).

Enkripsi atau dekripsi sangat penting terutama dalam komunikasi tanpa kabel ini disebabkan karena komunikasi tanpa kabel sangat mudah disadap. Karena itu enkripsi atau dekripsi merupakan hal yang penting dalam setiap hal-hal yang sangat rahasia, seperti pembayaran kartu, atau percakapan rahasia antara satu departemen dengan departemen yang lainnya dalam suatu organisasi. Semakin baik pula keamanan yang akan didapat, namun juga semakin mahal nilai keamanannya.

D. Model-Model Kriptografi

Metode atau model kriptografi merupakan sebuah cara untuk diketahui oleh dua pribadi yang berkomunikasi. Berikut metode kriptografi :

1. Sistem *Diffie Hellman*
   Kunci atau algoritma pertukaran ini dittemukan oleh Whitfield Diffie dan Martin Hellman pada tahun 1976 dan sebelumnya ditemukan oleh Malcolm Williamson pada tahun 1974. Sistem ini dipakai untuk menyandikan pertukaran pesan antara dua pihak secara interaktif. Pada awalnya, masing-masing pihak mempunyai sebuah kunci rahasia yangn tidak diketahui pihak lawan bicara. Berdasar pada masing-masing kunci rahasia ini, kedua belah pihak dapat membuat sebuah kunci sesi (session *Key*) yang akan dipakai untuk pembicaraan selanjutnya.
2. *RSA*
   Singkatan dari huruf depan dari tiga orang yang menemukannya pada tahun 1977 di MIT, yaitu Ron Rivest, Adi Shamir, dan Len Adleman. Algoritma ini merupakan cara enkripsi publik yang sangat kuat saat ini. Algoritma *RSA* melibatkan seleksi digit angka prima dan mengalikan secara be*RSA*ma-sama untuk mendapatkan jumlah, yaitu n. Angka-angka ini dilewati algoritma matematis untuk menentukan kunci publik KU={e,n} dan kunci pribadi KR={d,n} yang secara metematis berhubungan.

III. METODE PENELITIAN

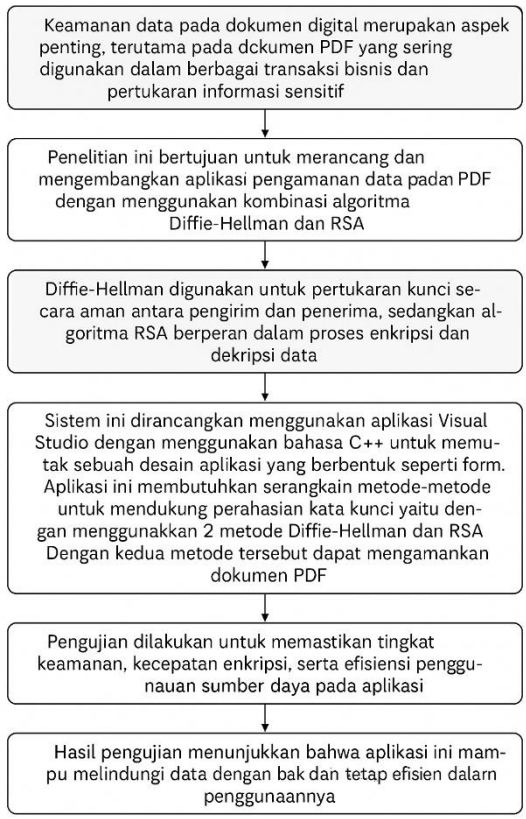
Sistem ini dirancangkan menggunakan aplikasi Visual Studio dengan menggunakan bahasa C++ untuk membuat sebuah desain aplikasi yang berbentuk seperti *form.* Aplikasi ini membutuhkan serangkain metode-metode untuk mendukung perahasian kata kunci yaitu dengan menggunakan 2 metode Diffie-helman dan RSA. Dengan kedua metode tersebut dapat mengamankan dokumen *PDF.*

Metode perancangan pada penelitian ini menggunakan kombinasi *Diffie-Hellman* dan *RSA* dipilih adalah karena kekuatannya dalam mengamankan proses enkripsi dan dekripsi data. *Diffie-Hellman* memungkinkan pertukaran kunci secara aman antara dua pihak, sedangkan *RSA* digunakan untuk enkripsi dan dekripsi data dengan kunci publik dan privat. Kombinasi ini memberikan lapisan keamanan ganda, yang membuat aplikasi ini lebih tahan terhadap serangan kriptografi, seperti serangan *man-in-the-middle* atau *brute force*. Dengan demikian, kebutuhan akan sistem yang dapat memberikan jaminan keamanan data yang kuat dan andal menjadi alasan penting dalam perancangan aplikasi ini.

Implementasi aplikasi ini melibatkan beberapa tahapan kunci. Pertama, saat pengirim ingin mengirimkan dokumen *PDF*, dia akan menggunakan *Diffie-Hellman* untuk menghasilkan kunci enkripsi simetris be*RSA*ma dengan penerima. Kemudian, dokumen *PDF* akan dienkripsi menggunakan kunci simetris tersebut. Setelah itu, pengirim menandatangani dokumen yang telah dienkripsi menggunakan kunci privat *RSA*. Dokumen terenkripsi beserta tanda tangannya dikirimkan ke penerima, yang kemudian menggunakan kunci simetris untuk mendekripsi dokumen dan kunci publik *RSA* pengirim untuk memverifikasi tanda tangannya. Dengan sistem ini, keamanan dokumen *PDF* selama transmisi dan penyimpanan dapat dijaga dengan baik.

p-ISSN : 2406-7733
e-ISSN : 2597-9922

Berikut adalah proses penjelasan tahap penelitian yang dijelaskan alur seperti gambar 1.

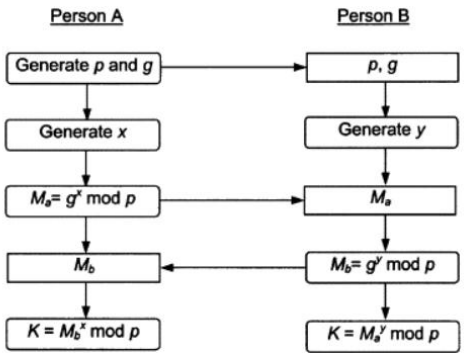


Gambar 1. Alur atau proses penelitian

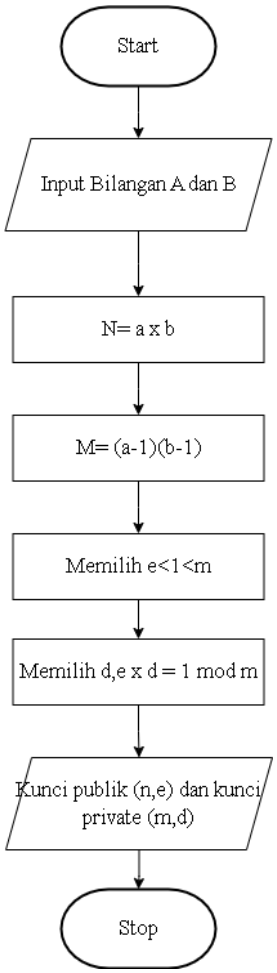
IV. HASIL DAN PEMBAHASAN

A. *Flowchart* Sistem

Dalam proses implementasi algoritma pada Aplikasi *Histogram Equalization* Pada *Real Time Cascade Classifier Face Tracking*, penulis merancang beberapa rancangan sebagai berikut.



Gambar 2. *Diffie-Hellman Key Exchange*



Gambar 3. *Flowchart* Pembangkitan Kunci *RSA*

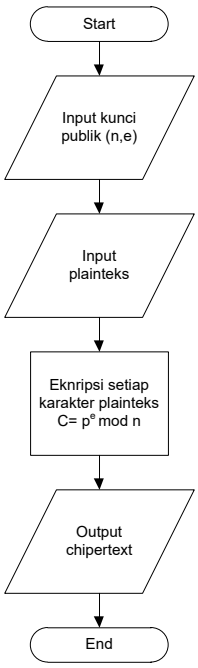
Pada *flowchart* gambar 1 dan *flowchart* gambar 3 menampilkan gambaran proses dari pembangkitan kunci. Pertama, *Diffie-Hellman* digunakan untuk melakukan pertukaran kunci secara aman. Proses ini dimulai dengan kedua pihak, Alice dan Bob, memilih bilangan prima p dan basis g yang diketahui bersama. Alice dan Bob kemudian masing-masing memilih kunci privat secara rahasia dan menghitung kunci publik mereka. Setelah saling bertukar kunci publik, kedua pihak menghitung kunci be*RSA*ma yang digunakan untuk enkripsi simetris. Kunci be*RSA*ma ini memungkinkan mereka untuk menghasilkan kunci simetris yang akan digunakan dalam proses enkripsi dan dekripsi data.

Setelah kunci bersama dihasilkan dari Diffie-Hellman, *RSA* digunakan untuk enkripsi data yang lebih kompleks dan aman. Dalam tahap ini, *RSA* mengandalkan pasangan kunci publik dan privat.

Dengan menggabungkan *Diffie-Hellman* dan *RSA* dalam satu sistem, keamanan sistem secara keseluruhan ditingkatkan. *Diffie-Hellman* memastikan bahwa kunci simetris yang digunakan dalam enkripsi data dihasilkan secara aman, sementara *RSA* memberikan lapisan tambahan

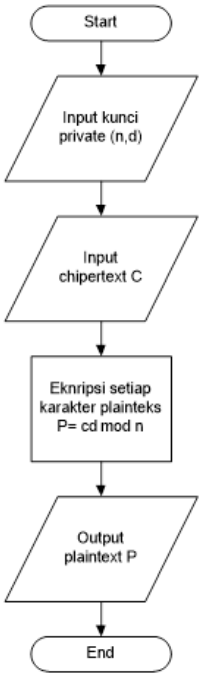
p-ISSN : 2406-7733
e-ISSN : 2597-9922

perlindungan melalui enkripsi asimetris. Penjadwalan kunci ini memastikan bahwa data yang ditransmisikan tetap aman, terlepas dari potensi ancaman dalam saluran komunikasi yang tidak aman.



Gambar 4. *Flowchart* Enkripsi Algoritma *RSA*

Flowchart di atas menjelaskan proses pembuatan kunci publik dan kunci privat menggunakan algoritma *RSA*. Langkah pertama adalah memilih dua bilangan prima, a dan b. Setelah itu, dihitung nilai N sebagai hasil perkalian a dan b, serta nilai M sebagai hasil kali dari (a-1) dan (b-1).



Gambar 5. *Flowchart* Dekripsi Algoritma *RSA*

Flowchart di atas menunjukkan proses dekripsi dalam kriptografi *RSA*. Dimulai dengan memasukkan kunci privat (n, d) yang sudah dihasilkan sebelumnya. Setelah itu, *ciphertext* C yang akan didekripsi dimasukkan.

Proses dekripsi dilakukan dengan mengaplikasikan rumus P=C mod n pada setiap karakter *ciphertext*, di mana P adalah plaintext yang dihasilkan. Setelah semua karakter diproses, *plaintext* P dikeluarkan sebagai hasil akhir, dan proses dekripsi selesai.

B. Implementasi Sistem

Pada Program Pengamanan Data Pada Dokumen *PDF* Menggunakan kombinasi *Diffie-Hellman* dan *RSA* terdapat beberapa interface atau antarmuka yang di desain untuk mempermudah user atau pemakai dalam menggunakan atau menjalankan program ini. Adapun *interface* atau antarmuka adalah sebagai berikut :

1. Halaman Utama

   Halaman utama aplikasi ini menampilkan antarmuka yang sederhana dan intuitif, dimulai dengan judul "Pengamanan *PDF* Menggunakan *Diffie Hellman* dan *RSA*" yang langsung memberikan informasi mengenai tujuan utama aplikasi, yaitu untuk mengamankan pesan teks dalam *file PDF* menggunakan dua metode kriptografi terkenal: *Diffie-Hellman* dan *RSA*.

   Di tengah halaman, terdapat ilustrasi yang menggambarkan sebuah kunci, gembok, *file PDF*, dan komputer. Gambar ini mencerminkan fungsi aplikasi yang berfokus pada keamanan, terutama dalam konteks pengamanan data melalui enkripsi.

   Di sisi kanan layar, terdapat beberapa tombol navigasi yang memungkinkan pengguna untuk mengakses fitur-fitur utama aplikasi. Tombol pertama, "Pembangkitan Kunci", digunakan untuk menghasilkan kunci kriptografi yang diperlukan untuk proses enkripsi dan dekripsi. Selanjutnya, tombol "Enkripsi" berfungsi untuk mengamankan pesan teks dalam *PDF* dengan cara mengenkripsinya. Sebaliknya, tombol "Dekripsi" memungkinkan pengguna untuk mengembalikan pesan terenkripsi menjadi teks asli. Terdapat juga tombol "About" yang menyediakan informasi tambahan tentang aplikasi atau pengembangnya, dan tombol "Keluar" yang berfungsi untuk menutup aplikasi.

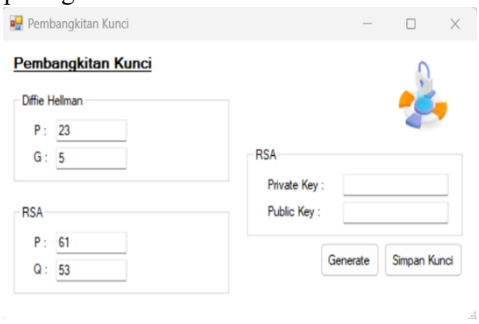
   Di bagian bawah halaman, terdapat deskripsi singkat yang menyatakan bahwa aplikasi ini berfungsi untuk mengamankan pesan teks pada *file PDF* menggunakan metode kriptografi *Diffie-Hellman* dan *RSA*,

p-ISSN : 2406-7733
e-ISSN : 2597-9922

memberikan gambaran umum mengenai manfaat utama aplikasi bagi pengguna.. Adapun tampilan dari halaman login admin dapat dilihat pada gambar berikut :

Gambar 6. Tampilan Utama Program

2. Menu Pembangkitan Kunci
Halaman Pembangkitan Kunci adalah halaman yang dikhususkan untuk membuat atau mengenerate kunci *Public* dan *Private* pengirim dan penerima. Adapun tampilan dari halaman Pembangkitan Kunci dapat dilihat pada gambar 4.2



Gambar 7. Halaman Pembangkitan Kunci

Halaman ini adalah bagian dari aplikasi yang menampilkan fitur Pembangkitan Kunci. Fitur ini digunakan untuk menghasilkan kunci kriptografi yang akan digunakan dalam proses enkripsi dan dekripsi menggunakan metode *Diffie-Hellman* dan *RSA*.

Pada bagian kiri atas, terdapat area untuk metode *Diffie-Hellman*. Terdapat dua nilai yang perlu dimasukkan, yaitu:

- P: Sebuah bilangan prima yang berperan sebagai modulus.
- G: Sebuah bilangan yang berfungsi sebagai basis atau generator.

Bagian kiri bawah menampilkan input untuk metode *RSA*. Dua nilai yang dibutuhkan adalah:

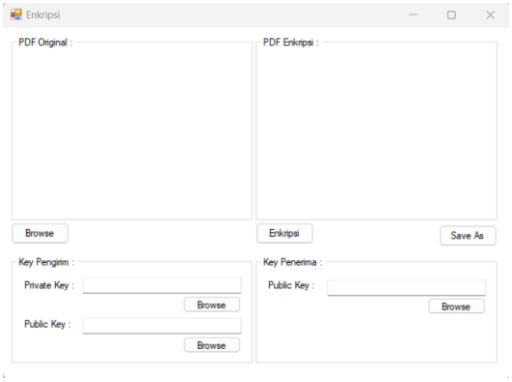
- P: Nilai bilangan prima pertama.
- Q: Nilai bilangan prima kedua.

Di sisi kanan, ada bagian untuk *RSA Key Generation*, di mana pengguna dapat menghasilkan kunci privat dan publik:

- *Private Key*: Kolom ini akan menampilkan kunci privat setelah dihasilkan.
- *Public Key*: Kolom ini akan menampilkan kunci publik setelah dihasilkan.

Ada dua tombol utama yang berfungsi untuk proses:

a. Generate: Tombol ini digunakan untuk menghasilkan kunci privat dan publik berdasarkan nilai yang dimasukkan.
b. Simpan Kunci: Setelah kunci dihasilkan, tombol ini memungkinkan pengguna menyimpan kunci yang telah dihasilkan untuk digunakan dalam proses enkripsi dan dekripsi.



Gambar 8. Halaman Enkripsi

Tampilan aplikasi yang terlihat dalam gambar adalah sebuah antarmuka dari sebuah aplikasi enkripsi *PDF* yang dirancang untuk mengamankan *file* dengan metode kriptografi. Aplikasi ini memiliki beberapa komponen penting yang terbagi secara jelas untuk memudahkan pengguna dalam menjalankan proses enkripsi dan dekripsi *file PDF*.

Di bagian atas antarmuka, terdapat dua area besar yang ditujukan untuk mengelola *file PDF*. Di sebelah kiri, area bertuliskan "*PDF* Original" merupakan tempat pengguna dapat memilih dan menampilkan *file PDF* yang ingin dienkripsi. Pengguna dapat menggunakan tombol "*Browse*" yang berada di bawahnya untuk mencari *file PDF* di komputer mereka. Setelah *file* dipilih, *file* tersebut akan muncul di area ini sebagai *PDF* original yang akan dienkripsi.

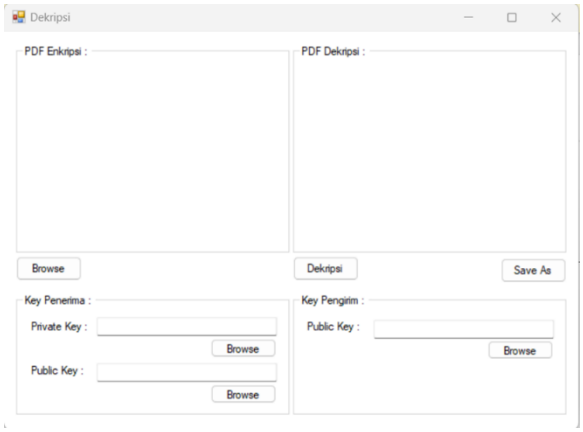
Di sebelah kanan dari area ini, terdapat area "*PDF* Enkripsi" yang akan menampilkan hasil dari proses enkripsi. Setelah *file PDF* di bagian "*PDF* Original" dipilih dan kunci enkripsi diatur, pengguna dapat menekan tombol "Enkripsi" untuk memulai proses enkripsi. Hasil dari proses ini akan ditampilkan di area "*PDF* Enkripsi". Selain itu, ada tombol "*Save As*" yang berfungsi untuk menyimpan *file PDF* yang sudah terenkripsi ke dalam

p-ISSN : 2406-7733
e-ISSN : 2597-9922

komputer, memberikan pengguna kontrol untuk menyimpan *file* terenkripsi sesuai keinginan.

Di bagian bawah antarmuka, aplikasi ini memiliki dua bagian yang terfokus pada pengelolaan kunci enkripsi: "*Key* Pengirim" dan "*Key* Penerima". Pada bagian "*Key* Pengirim", terdapat dua kolom utama, yaitu "*Private Key*" dan "*Public Key*". Pengguna perlu memuat kunci privat (*Private Key*) dan kunci publik (*Public Key*) dari pengirim melalui tombol "*Browse*" yang disediakan. Kunci ini digunakan dalam proses enkripsi agar hanya pihak yang memiliki kunci yang sesuai dapat membuka *file PDF* yang telah dienkripsi.

Di sebelah kanan, terdapat bagian "*Key* Penerima", yang hanya terdiri dari satu kolom untuk memuat "*Public Key*" penerima. Pengguna dapat menggunakan tombol "*Browse*" untuk memilih kunci publik penerima yang akan digunakan dalam proses enkripsi. Kunci ini memastikan bahwa *file PDF* hanya bisa didekripsi oleh penerima yang memiliki kunci privat yang benar.



Gambar 9. Halaman Dekripsi

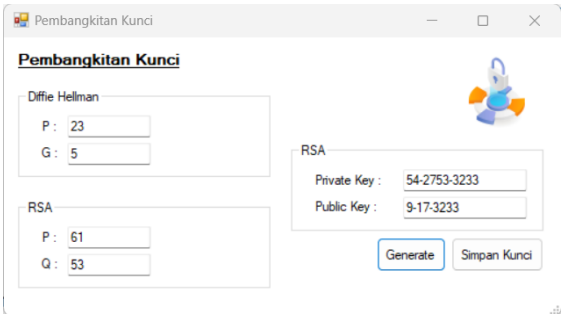
Tampilan aplikasi pada gambar ini adalah antarmuka untuk sebuah aplikasi dekripsi *PDF* menggunakan metode kriptografi, dengan struktur yang mirip seperti aplikasi enkripsi yang telah dijelaskan sebelumnya. Namun, kali ini fokusnya adalah untuk membuka *file PDF* yang telah terenkripsi. Berikut adalah narasi penjelasan dari masing-masing komponen:

Di bagian atas, terdapat dua area besar yang mewakili proses dekripsi *PDF*. Di sebelah kiri, area bertuliskan "*PDF* Enkripsi" digunakan untuk memilih *file PDF* yang telah terenkripsi. Pengguna bisa menggunakan tombol "*Browse*" di bawahnya untuk memilih *file PDF* yang ingin didekripsi dari perangkat mereka.

Di sebelah kanan dari area ini, terdapat area "*PDF* Dekripsi" yang akan menampilkan hasil dari proses dekripsi. Setelah *file PDF* terenkripsi dipilih dan kunci-kunci kriptografi yang diperlukan diatur, pengguna bisa menekan tombol "Dekripsi" untuk memulai proses. Hasil dekripsi, yaitu *file PDF* yang kembali ke bentuk aslinya, akan muncul di sini. Selain itu, ada tombol "*Save As*" yang memungkinkan pengguna menyimpan hasil dekripsi ke perangkat mereka.

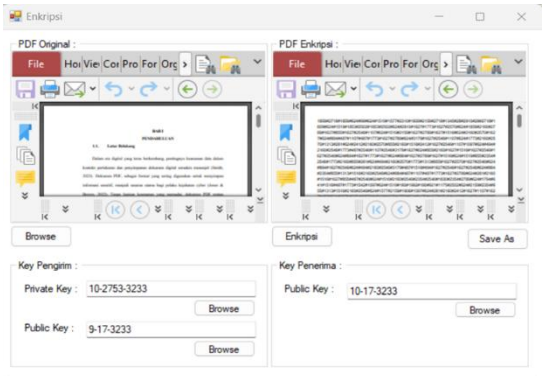
Di bagian bawah, terdapat dua bagian yang penting untuk proses dekripsi, yaitu "*Key* Penerima" dan "*Key* Pengirim". Pada bagian "*Key* Penerima", terdapat dua kolom untuk memuat kunci yang dimiliki oleh penerima, yaitu "*Private Key*" dan "*Public Key*". Pengguna dapat menggunakan tombol "*Browse*" untuk mengunggah kunci privat dan publik yang diperlukan untuk membuka *file* yang terenkripsi.

Sementara itu, di sebelah kanan, terdapat bagian "*Key* Pengirim" yang hanya berisi satu kolom untuk "*Public Key*" pengirim. Pengguna perlu memuat kunci publik pengirim melalui tombol "*Browse*". Kunci ini berperan penting dalam memvalidasi bahwa *file* yang terenkripsi benar-benar berasal dari pengirim yang sah.



Gambar 10. Proses Pembangkitan Kunci Penerima dan Pengirim

Pada bagian ini dapat dilihat bahwa sistem men-*generate* kunci *Private*, dan kunci *Public* perlu diingat bahwa kunci ini akan degenerate sebanyak dua kali yaitu untuk penerima dan pengirim. Untuk keperluan itu maka kunci akan disimpan pada dua folder yang berbeda dengan masing-masing folder untuk pengirim, dan penerima. Hal ini dikarenakan tata cara penggunaan kunci *Public* dan *Private* yang berbeda antara *RSA* dengan Diffie Hellman.



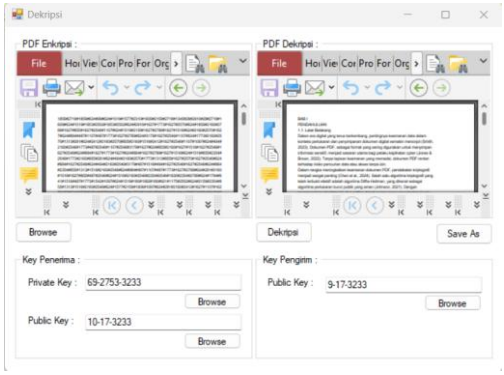
Gambar 11. Proses Enkripsi *File PDF*

p-ISSN : 2406-7733
e-ISSN : 2597-9922

Gambar yang ditampilkan menunjukkan proses enkripsi sebuah *file PDF* dalam sebuah aplikasi enkripsi dokumen. Berikut adalah narasi lengkap mengenai bagaimana proses enkripsi berlangsung dalam aplikasi ini berdasarkan gambar yang ada:
Pada bagian kiri antarmuka, terdapat area *PDF* Original, yang menampilkan *file PDF* asli sebelum proses enkripsi dilakukan. Dalam contoh ini, *file PDF* yang dipilih berisi dokumen dengan format teks yang menampilkan bab "PENDAHULUAN". Pengguna memilih *file PDF* ini menggunakan tombol "*Browse*", yang berada di bawah area ini.

Di sebelah kanan, area *PDF* Enkripsi menunjukkan hasil dari proses enkripsi. Setelah *file PDF* asli dipilih, pengguna bisa memulai proses enkripsi dengan menekan tombol "Enkripsi". Pada gambar ini, terlihat hasil enkripsi berupa teks terenkripsi yang berbentuk karakter-karakter acak dan tidak terbaca.

Proses enkripsi ini memastikan bahwa data dalam *file PDF* hanya bisa dibuka oleh penerima yang memiliki kunci kriptografi yang sesuai, sehingga keamanan dokumen tetap terjaga.



Gambar 12. Proses Dekripsi *File PDF*

Gambar ini menampilkan proses dekripsi *file PDF* yang telah terenkripsi. Berikut adalah narasi lengkap mengenai proses dekripsi yang sedang terjadi berdasarkan gambar yang ditampilkan: Antarmuka aplikasi terdiri dari dua bagian utama untuk menampilkan *file PDF* sebelum dan sesudah proses dekripsi. Di bagian kiri atas, terdapat area yang bertuliskan *PDF* Enkripsi yang menunjukkan *file PDF* yang telah terenkripsi dalam bentuk teks terenkripsi. Di sebelah kanan, terdapat area *PDF* Dekripsi, yang akan menampilkan hasil dari proses dekripsi, yakni *file PDF* asli yang dapat dibaca setelah proses dekripsi berhasil dilakukan.

V. KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan proses perancangan Program Pengamanan Data Pada Dokumen *PDF* Menggunakan kombinasi *Diffie-Hellman* dan *RSA*, maka dapat diambil beberapa kesimpulan yaitu:

1. Peningkatan keamanan dokumen *PDF* dengan kombinasi algoritma program yang dirancang berhasil mengimplementasikan kombinasi metode *Diffie-Hellman* untuk pertukaran kunci yang aman dan *RSA* untuk enkripsi/dekripsi dokumen *PDF*. Kombinasi kedua algoritma ini memberikan lapisan keamanan yang kuat, di mana *Diffie-Hellman* memungkinkan pengiriman kunci secara aman, dan *RSA* digunakan untuk mengenkripsi konten dokumen, sehingga mencegah akses tidak sah.
2. Integrasi mekanisme kriptografi untuk perlindungan data melalui penggunaan kunci publik dan kunci privat, program ini memastikan bahwa hanya pihak yang memiliki akses ke kunci privat yang dapat mendekripsi dokumen *PDF* yang dienkripsi. Dengan demikian, integritas dan kerahasiaan dokumen terjamin, bahkan jika *file* terenkripsi diakses oleh pihak ketiga. Hal ini juga meminimalkan risiko pemalsuan dokumen.
3. Efektivitas pengamanan dokumen dalam aplikasi nyata uji coba program menunjukkan bahwa metode ini dapat diimplementasikan secara efektif dalam pengamanan dokumen *PDF* yang sering digunakan dalam konteks legal, akademis, dan bisnis. Pengguna dapat dengan mudah mengenkripsi dan mendekripsi dokumen tanpa kehilangan keaslian dokumen, serta melindungi informasi sensitif dari potensi ancaman keamanan siber.

Saran

Saran saran yang penulis kemukakan diharapkan dapat lebih meningkatkan hasil yang telah didapatkan. Berikut ini beberapa saran yang di sampaikan oleh penulis adalah.

1. Optimisasi waktu proses enkripsi dan dekripsi meskipun keamanan dokumen terjamin, proses enkripsi dan dekripsi yang melibatkan algoritma *RSA* dapat memakan waktu lebih lama pada *file PDF* berukuran besar. Pengoptimalan algoritma atau penggunaan pendekatan hybrid seperti AES untuk enkripsi simetris data, sementara *RSA* digunakan untuk mengamankan kunci, dapat meningkatkan efisiensi waktu.
2. Pengembangan fitur otentikasi tambahan selain pengamanan data, disarankan untuk menambahkan fitur otentikasi digital menggunakan tanda tangan elektronik (digital signature) yang terintegrasi dengan sistem. Hal ini akan memperkuat verifikasi identitas pengirim dan memastikan keaslian dokumen, serta memberikan lapisan keamanan tambahan dalam aplikasi.

p-ISSN : 2406-7733
e-ISSN : 2597-9922

DAFTAR PUSTAKA

Cheng, D., & Chen, Y. (2024). *Symmetric multiple-image encryption algorithm using Chen ' s Hyper-Chaotic System*. 0–19.

Desai, H., & Choksi, R. (2020). *Introduction to Cryptography For History Cryptography*.

Gunawan, I. (2019). *Keamanan Data: Teori dan Implementasi*.

Liander, G. V. (2022). Penggunaan Algoritma Elliptic Curve Diffie Hellman dan AES 256 pada Implementasi End-to-End Encryption WhatsApp. *Sumber*, *18219075*. https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi-dan-Koding/2021-2022/Makalah2022/Makalah-II4031-Kripto-2022 (14).pdf

Miller, V. S. (2023). Elliptic Curves and their use in Cryptography. *Communications*, *November 2023*, 1–14.

Nova Situmoran, J., Nainggolan, E., Saputra Loi, A., Wendi Hutablian, A., & Franjein Hutasoit, A. (2023). Security Analysis of Diffie-Hellman Algorithm in Cryptographic Key Exchange. *Jurnal Teknik Indonesia*, *2*(01), 1–7. https://doi.org/10.58471/juti.v2i01.654

Patgiri, R. (2021). privateDH : An Enhanced Diffie-Hellman Key-Exchange Protocol using RSA and AES Algorithm. *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03â•fi05, 2021, Woodstock, NY*, *1*(1), 1–6.

Rivest, R. L., Shamir, A., & Adleman, L. (2023). A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM*, *21*(2), 120–126. https://doi.org/10.1145/359340.359342

Singh, R. G., & Ruj, S. (2020). *A Technical Look At The Indian Personal Data Protection Bill*. 1–43. http://arxiv.org/abs/2005.13812

Victor, M., Praveenraj, D. D. W., Sasirekha, R., Alkhayyat, A., & Shakhzoda, A. (2023). Cryptography: Advances in Secure Communication and Data Protection. *E3S Web of Conferences*, *399*. https://doi.org/10.1051/e3sconf/202339907010