IMPLEMENTASI KRIPTOGRAFI DIGITAL SIGNATURE MENGGUNAKAN SECURE HASH ALGORITHM (SHA-1)

Yoriana Ade Putri

Evanita Veronica Manullang
eva.manullang@gmail.com

Program Studi Teknik Informatika
Fakultas Ilmu Komputer dan Manajemen
Universitas Sains dan Teknologi Jayapura

Abstraksi - *SHA-1 (Secure Hash Algorithm) merupakan keluarga fungsi hash satu arah, yang digunakan dalam DSA (Digital Signature Algorithm) untuk pembuatan tanda tangan digital. Yang dimaksud dengan tanda-tangan digital bukanlah tanda tangan yang di-digitasi dengan alat scanner, tetapi suatu nilai kriptografis yang bergantung pada pesan dan pengirim pesan. Dengan tanda-tangan digital, maka integritas data dapat terjamin, sehingga bisa mengurangi kasus kriminal yang terjadi belakangan ini, seperti pembajakan dan kasus pengakuan hasil karya atau plagiat. Contoh kasus krimanal lainnya yaitu saat pengiriman file, seseorang bisa saja dengan ilegal mengubah isi file itu tanpa diketahui oleh pengirim dan penerima. Penelitian ini bertujuan untuk membangun sebuah aplikasi yang mampu mengamankan file dokumen dengan memanfaatkan tanda-tangan digital menggunakan Secure Hash Algorithm (SHA-1), agar dapat membantu pihak yang ingin bertukar informasi merasa aman, karena percaya bahwa pesan yang diterima masih asli. Perancangan aplikasi tanda-tangan digital dibangun dengan menggunakan bahasa pemrograman Delphi7. Aplikasi yang dihasilkan berupa tanda-tangan digital yang akan dikirim bersamaan dengan file aslinya kepada penerima, kemudian digunakan oleh penerima untuk memverifikasi keaslian file.*

Kata kunci : SHA-1 (Secure Hash Algorithm), Tanda-tangan, Digital, Delphi7

I. PENDAHULUAN

Saat ini banyak kasus kriminal seperti pembajakan dan kasus pengakuan hasil karya atau plagiat. Contoh kasus kriminal lainnya yaitu saat pengiriman file, seseorang bisa saja dengan ilegal mengubah isi file itu tanpa diketahui oleh pengirim dan penerima. Untuk memperkuat keaslian terhadap file, maka digunakan tandatangan digital. Penerima pesan akan percaya bahwa pesan yang diterima masih asli karena dibubuhkan tanda tangan digital. Tandatangan Digital digunakan untuk menjamin integritas data, otentikasi dan nirpenyangkalan (tidak dapat disangkal). *SHA-1* (*Secure Hash Algorithm*) merupakan keluarga fungsi hash satu arah. *SHA-1* menerima masukan berupa pesan dengan ukuran maksimum 2^{64} bit (2.147.483.648 gigabyte) dan menghasilkan *message digest* (*MD*) dengan panjang 160 bit, yang kemudian digunakan dalam *DSA* (*Digital Signature Algorithm*) untuk pembuatan tanda tangan digital. *SHA* dikatakan aman karena tidak mungkin menemukan dua pesan yang berbeda yang menghasilkan *MD* yang sama, atau tidak mungkin menemukan pesan yang sama jika diberikan suatu nilai hash-nya. Berdasarkan penjabaran diatas, maka dirancang sebuah aplikasi yang dapat membantu mengamankan file seperti dokumen, yang diharapkan dapat membantu pihak yang ingin bertukar informasi merasa aman karena percaya bahwa pesan yang diterima masih asli.

II. TINJAUAN PUSTAKA

Dalam tinjauan pustaka berisi tentang perbandingan antara penelitian terdahulu dengan penelitian yang akan dilakukan oleh penulis, perbandingannya adalah sebagai berikut:

Nesya Rimanda (2011) jenis penelitian skripsi berjudul "Aplikasi Kriptografi *Digital Signature* Untuk Kerahasiaan Email Dengan Menggunakan Metode *RSA* dan *SHA-5*". Tools yang digunakan dalam perancangan sistem yaitu bahasa pemrograman *Visual Basic 6.0*. Hasil penelitian adalah proses pengamanan file text terjaga dan disertai dengan tanda tangan digital sebagai keabsahan atau otentifikasi untuk pembenaran data tersebut.

Sopina Tiur Maria Panggabean (2011) jenis penelitian skripsi berjudul "Aplikasi Enkripsi dan Dekripsi Menggunakan Kriptografi Twofish Untuk Mengamankan File dan Folder".Tools yang digunakan dalam perancangan sistem yaitu bahasa pemrograman *Visual Basic 6.0*. Hasil penelitian adalah isi file yang telah dienkripsi berasil teracak sehingga file terebut tidak bisa dimengerti, dan hasil dekripsi sama dengan file asli sebelum dienkripsi.

Evie Iriana Putri (2011) jenis penelitian skripsi berjudul "Identifikasi Tanda Tangan Dengan Menggunakan Metode *Learning Vector Quantization* (*LVQ*)".Tools yang digunakan dalam perancangan sistem yaitu bahasa pemrograman *Visual Basic 6.0*. Hasil penelitian adalah aplikasi dapat mencatat semua langkah-langkah proses perhitungan yang dilakukan, sehingga dapat membantu pembelajaran terhadap metode *LVQ* dalam implementasinya untuk mengenali pola tanda tangan.

Muhammad Reza Setiabudi (2012) jenis penelitian skripsi berjudul "Aplikasi Pengamanan Email Menggunakan Protokol *Digital Signature Alogorithm* (*DSA*) dan Metode Kriptografi *Government Standard* (*Gost*)". Tools yang digunakan dalam perancangan sistem yaitu bahasa pemrograman *Visual Basic 6.0*. Hasil penelitian adalah aplikasi dapat menjamin kerahasiaan dan keaslian pesan email serta identitas penulis pesan melalui kunci tanda tangan digital, serta menampilkan langkah-langkah detail dari proses *SHA-1, DSA* dan metode *GOST*, sehingga dapat membantu pelajaran terhadap metode kriptografi tersebut.

Ramlah (2013) jenis penelitian skripsi berjudul "Implementasi Algoritma Kriptografi El-Gamal Dalam Menjamin Keaslian dan Keutuhan Pesan".Tools yang digunakan dalam perancangan sistem yaitu bahasa pemrograman *Visual Basic 6.0*. Hasil penelitian adalah aplikasi menghasilkan kode tanda tangan digital El-Gamal dari pesan dan melakukan verifikasi terhadap pesan dan kode tanda tangan digital, sehingga aplikasi dapat digunakan sebagai aplikasi teks editor yang mampu memverifikasi keaslian dan keutuhan pesan serta keaslian identitas pengirim dengan menggunakan protokol El-Gamal.

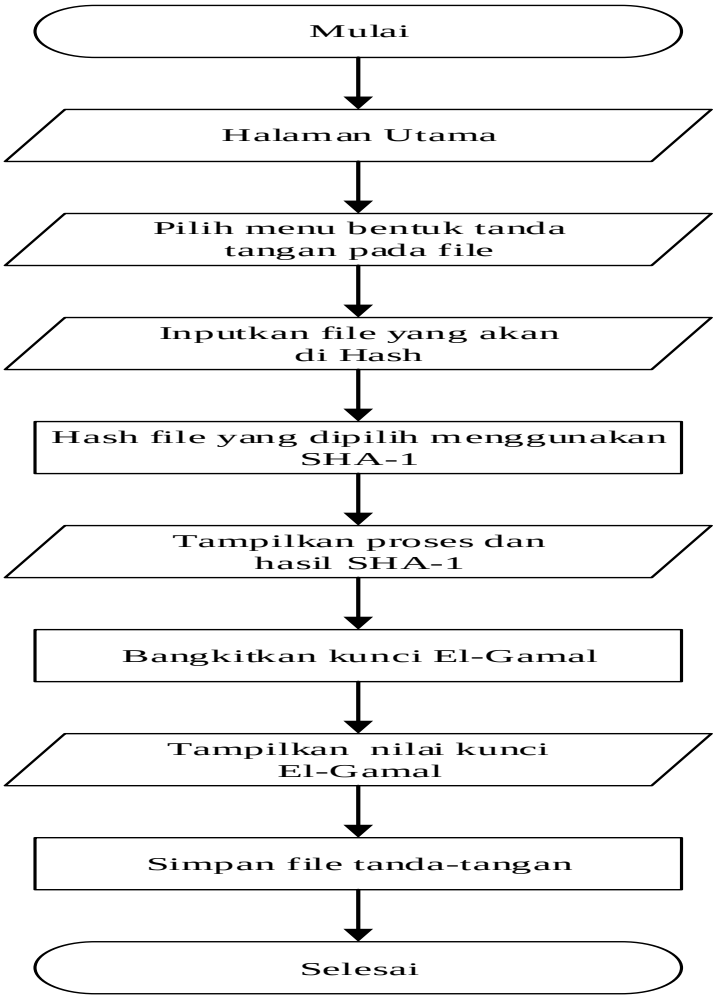
III. HASIL DAN PEMBAHASAN

A. Flowchart

Berikut ini merupakan diagram alir sistem aplikasi tanda tangan digital mnggunakan *SHA-1.*

a. Flowchart untuk pengirim

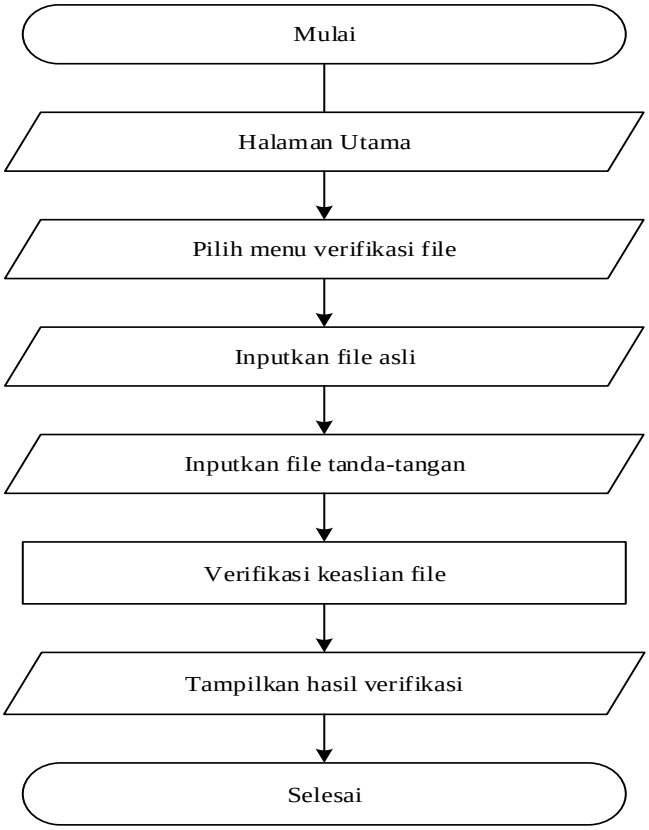
Berikut ini merupakan diagram alir sistem untuk pengirim yang menjelaskan proses *hashing* hingga penandatanganan file, seperti yang terlihat pada Gambar 1.



Gambar 1. Flowchart untuk pengirim

b. Flowchart untuk penerima

Berikut ini merupakan diagram alir sistem untuk penerima yang menjelaskan proses verifikasi pesan, seperti yang terlihat pada Gambar 2.



Gambar 2. Flowchart untuk penerima

B. Hasil

Berikut merupakan hasil implementasi sistem dari halaman utama hingga hasil verifikasi.

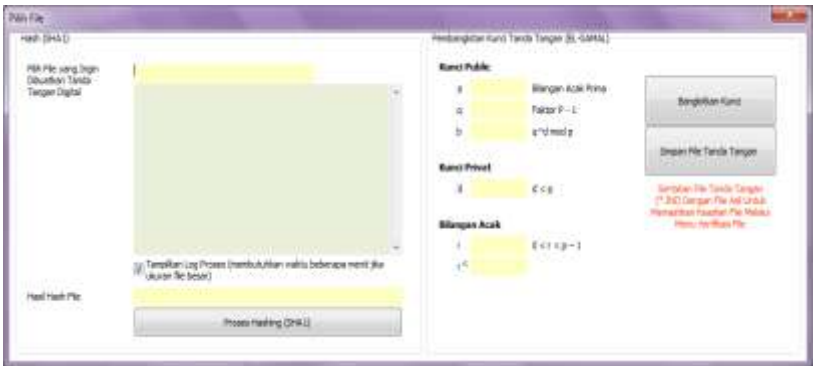
1. Form Halaman Utama

Form halaman utama merupakan tampilan awal dari aplikasi Implementasi Kriptografi Digital Signature Menggunakan Secure Hash Algorithm (SHA-1) ketika pertama kali dibuka seperti pada Gambar 3.

Gambar 3 Form Halaman Utama

2. Form Pilih File

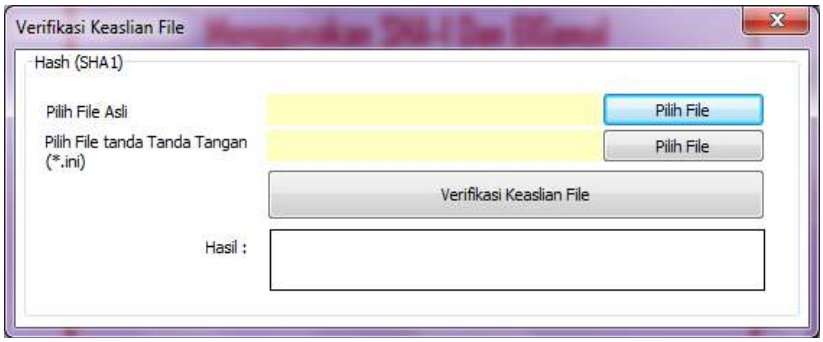
Form pilih file merupakan form yang akan digunakan oleh pengirim pesan untuk menghashing file menggunakan SHA-1, kemudian membangkitkat kunci ElGamal. Di dalam menu pilih file ini akan ditampilkan proses dari pencarian nilai SHA-1 pada file yang dipilih, seperti yang terlihat pada Gambar 4.



Gambar 4 Form Pilih File

3. Form Verifikasi

Form verifikasi merupakan form yang bertujuan untuk memverifikasi file. Pada form ini penerima akan menginputkan file asli dan file tanda tangan, kemudian memverifikasinya. Hasil dari verifikasi akan menampilkan pesan bahwa verfikasi berhasil dan pesan masih asli, atau file gagal diverifikasi karena tidak sesuai dengan file asli, seperti yang terlihat pada Gambar 5.



Gambar 5 Form Verifikasi

C. Pembahasan

1. Halaman Utama

Gambar 6 Halaman Utama

Hasil pengujiannya adalah sebagai berikut:

Halaman utama hanya menampilkan menu pembentukan tanda-tangan yang digunakan oleh pengirim, menu verifikasi yang digunakan oleh penerima untuk memverifikasi pesan yang diterima dari pengirim, serta menu tutup alikasi untuk keluar atau mengakhiri aplikasi.

2. Halaman Pilih File

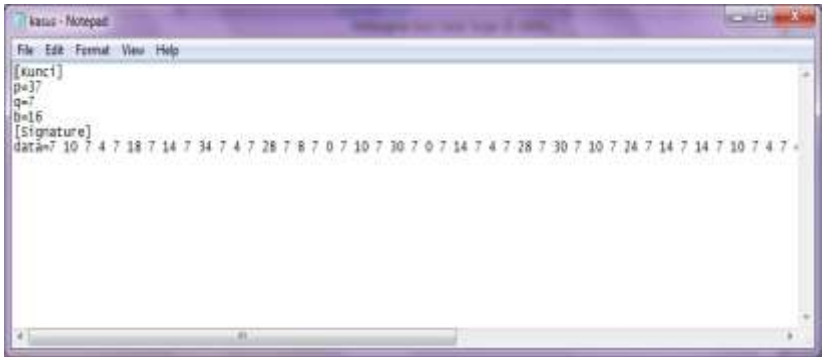
Halaman pilih file adalah halaman yang digunakan oleh pengirim untuk menghashing file serta membangkitkan kunci ElGamal yang kemudian akan dikirim oleh pengirim kepada penerima untuk diverifikasi.

Gambar 7 Pengujian halaman pilih file (proses hashing)

Hasil pengujiannya adalah sebagai berikut:

Pada proses ini, pengirim akan menginputkan file terlebih dahulu, kemudian melakukan *hashing*, dan sistem akan menampilkan proses serta hasil *hashing* dari file.Untuk proses pembangkitan kunci tanda-tangan (ElGamal), pengirim membangkitkan kunci, maka sistem akan menampilkan nilai kunci privat dan kunci publik yang akan digunakan oleh penerima untuk memverifikasi file. Pengirim kemudian menyimpan file yang sudah ditandatangani, dan sistem menampilkan informasi “kunci berhasil dibangkitkan.

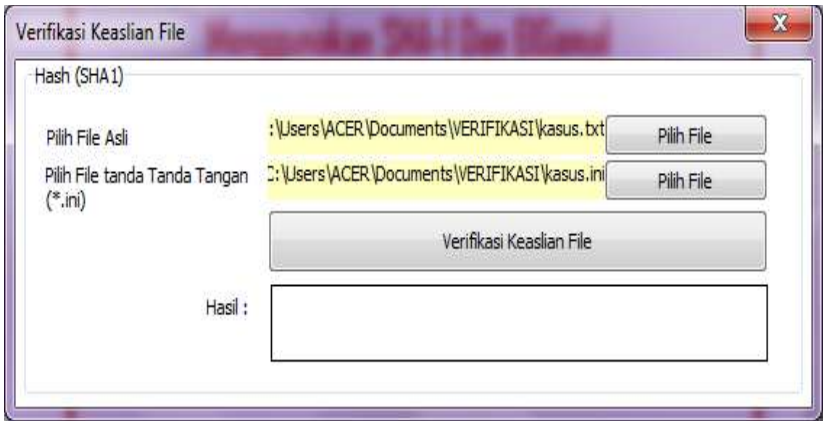
Misalnya, yang ditanda-tangani adalah file dengan nama kasus, pengirim menyimpan file yang sudah ditanda-tangani, kemudian sistem menampilkan informasi “File tanda-tangan berhasil tersimpan dengan nama kasus.ini pada lokasi yang sama dengan file yang anda pilih”. File yang tersimpan berisi kunci publik dan tanda tangan pesan, seperti yang terlihat pada Gambar 8.



Gambar 9 Tampilan file tanda-tangan

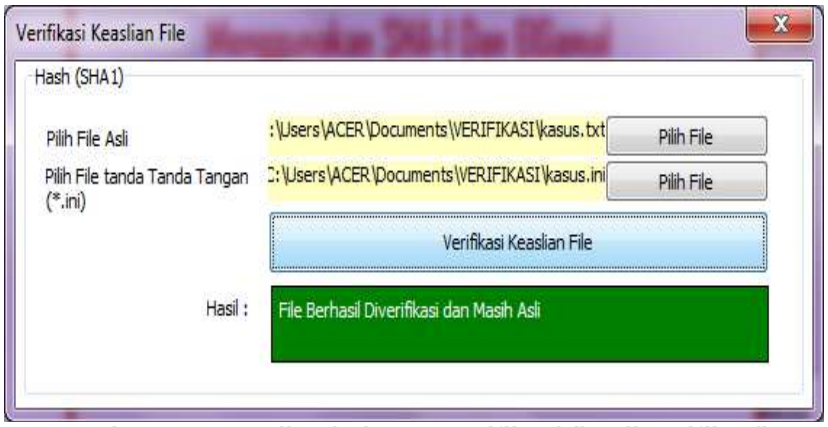
3. Halaman Verifikasi

Pada proses ini pihak penerima mula-mula menginputkan file asli dan file tanda-tangan yang diterima dari pengirim. Seperti yang terlihat pada Gambar 10.



Gambar 10 Pengujian halaman verifikasi (menginput file tanda-tangan)

Pada proses ini penerima memverifikasi file yang sudah diinputkan. Jika file masih asli, maka file berhasil diverifikasi. Jika file sudah diubah oleh pihak ketiga, maka file gagal di verifikasi.



Gambar 11 Pengujian halaman verifikasi (hasil verifikasi)

IV. PENUTUP

a. Kesimpulan

Berdasarkan hasil implementasi pada aplikasi kriptografi *digital signature* menggunakan *Secure Hash Algorithm* (*SHA-1*) maka penulis mengambil beberapa kesimpulan sebagai berikut :

1. Aplikasi *digital signature* menggunakan *Secure Hash Algorithm* (*SHA-1*) ini, bukan hanya menanda tangani file dokumen saja tetapi file gambar (*JPG), suara (*wma) juga bisa ditanda-tangani.
2. Dapat menampilkan proses *hashing*, namun untuk file yang berukuran lebih dari 200kb akan memakan waktu yang cukup lama untuk proses *hashingnya*.
3. Hasil *hashing* (*SHA-1*) akan berubah jika, pada pesan ditambahkan titik (.) atau spasi, dan disisipi virus.
4. Sebesar apapun file yang dipilih nilai *hash* tetap 160 bit.
5. File yang sudah ditanda-tangani dan disimpan oleh pengirim, akan tersimpan pada lokasi yang sama pada file yang dipilih.

6. Jika pihak ketiga mengubah isi file yang dikirim oleh pihak pengirim kepada penerima tanpa sepengetahuan kedua belah pihak, maka proses verifikasi gagal dan menampilkan pesan yang berbunyi “file gagal diverifikasi karena tidak sesuai dengan file asli atau sudah diubah”.
7. Untuk memverifikasi pesan pihak penerima harus memasukkan menginputkan file asli dan file tanda-tangan (*.ini).
8. Pihak penerima tidak bisa melakukan verifikasi pada file yang proses *hashingnya* menggunakan aplikasi lain karena tidak memiliki file tanda-tangan.

b. Saran

Adapun saran yang dapat digunakan untuk mengembangkan aplikasi yang telah dibuat yaitu Aplikasi yang dibuat sudah berjalan dengan baik untuk proses pencarian nilai *SHA-1* hingga proses verifikasi. Namun, ketika melakukan proses hashing pada file yang berukuran besar (200kb), maka proses pencarian nilai *SHA-1*akan menjadi lambat. Oleh karena itu, kedepannya diharapkan untuk bisa lebih mengembangkan aplikasi ini, terutama untuk file-file yang berukuran besar.

V. DAFTAR PUSTAKA

Adi Nugroho, 2005, *Analisis Dan Perancangan Sistem Informasi Dengan Metodologi Berorientasi Objek*, Cetakan Pertama (Edisi Revisi), Penerbit Informatika Bandung.

Bahri K.S, Sjachriyanto W, *Teknik Pemrograman Delphi*, Cetakan pertama November 2008 (Edisi Revisi), Penerbit Informatika Bandung.

Rifki Sadikin, 2012, *Kriptografi Untuk Keamanan Jaringan*, Penerbit Andi Publisher,Yogyakarta: ANDI, No. ISBN, 9789792931280.

Rinaldi Munir, 2006, *Kiptografi*, Cetakan pertama, Informatika Bandung.

Yakub, 2012, *Pengantar Sistem Informasi*, Penerbit Graha Ilmu, Yogyakarta.

Yuni Sugiarti, 2013, *Analisis Dan Perancangan UML* (*Unified Modeling Language*), EdisiPertama, Graha Ilmu,Yogyakarta.