Aplikasi Tanda Tangan Digital dengan Algoritma Gost untuk Keamanan Pengiriman File Dokumen

Edunal Yoppi[1], Zakarias Situmorang[2]

[1,2,] *Fakultas Ilmu Komputer Universitas Katolik Santo Thomas Medan, Indonesia*

ARTICLE INFORMATION

Received: April, 18, 2021
Revised: April 22, 2021
Available online: April,28,2021

KEYWORDS

Tanda Tangan Digital, Algoritma GOST

CORRESPONDENCE

E-mail: edunalyoppi@gmail.com[1]
zakarias65@yahoo.com[2]

A B S T R A C T

Digital signature is a signature based on a cryptographic scheme. Digital signatures are created using public key cryptography. The GOST algorithm is a public key algorithm that can be used for digital signature systems. The working mechanism of the GOST algorithm is quite simple and easy to understand but robust. The safety of GOST lies in the difficulty of factoring large numbers into prime factors. The software for simulating the digital signature system will be built using the Java programming language. The software being built will explain the digital signature process which includes the key formation process, signature formation and the verification process..

PENDAHULUAN

Teknologi jaringan internet telah melakukan perubahan terhadap masyarakat dalam melakukan interaksi dan komunikasi. Bahkan internet telah menjadi media untuk mengirim dokumen bisnis pada email dan transaksi-transaksi bisnis lainnya. Kegiatan-kegiatan tersebut tentu saja akan menimbulkan resiko bilamana informasi yang sensitif dan berharga tersebut di akses oleh orang-orang yang tidak berhak[1], [2].

Internet adalah suatu public network yang tidak aman. Saat pengiriman dokumen, seseorang bisa saja dengan ilegal mengubah isi dokumen itu tanpa diketahui pengirim atau penerima. Tanpa fasilitas keamanan yang baik, penerima akan menerima dokumen tersebut tampa mencurigai adanya perubahan dalam dokumen. Untuk itu diperlukan suatu tanda tangan pada dokumen, sehingga penerima dapat merasa yakin dengan adanya tanda tangan oleh pengirim, dokumen atau informasi tesebut tidak ada yang memanipulasi saat dalam perjalanan. Tanda tangan yang akan dibubuhi dalam dokumen ini disebut tanda tangan digital. Sama halnya seperti fungsi tanda tangan di atas dokumen kertas, tidak dapat disangkal, dimanipulasi dan diakui keasliannya. Fungsi tanda tangan pada dokumen kertas juga diterapkan pada data digital[3].

Tanda tangan digital diartikan sebagian orang sebagai bagian dari tanda tangan elektronik yang berbasiskan skema kriptografi. Dan tanda tangan digital memiliki tiga dari empat aspek keamanan yang dimiliki kriptografi yaitu: integritas data, otentikasi dan antipenyangkalan[4], [5]. Berdasarkan latar belakang diatas, maka pada penelitian ini dibangun suatu aplikasi tanda tangan digital dengan algoritma gost untuk keamanan pengiriman file dokument. Aplikasi yang dibangun ini diharapkan dapat meningkatkan keamanan dalam hal pemalsuan sebuah dokumen sehingga otoritas dari sebuah dokumen tersebut dapat terjaga.
Perancangan aplikasi tanda tangan digital dengan algoritma gost untuk keamanan pengiriman file dokumen untuk menjaga otoritas dan keaslian dari sebuah dokumen yang dikirimkan melalui saluran internet.

BAHAN DAN METODE

2. 1 Dokumen

The georgia archives (2004), dokumen adalah informasi yang dikumpulkan dan bisa di akses serta digunakan. Adapun *The International Standard Organization* (ISO on Record Management-ISO 15489) mendefenisikan *record* (dokumen) sebagai informasi yang diciptakan, di terima, dan dikelola sebagai bukti maupun informasi yang oleh organisasi ataupun perorangan digunakan untuk memenuhi kewajiban hukum atau transaksi bisnis. Dokumen ini mempunyai awal dan akhir yang dapat berupa teks, data, peta digital, *spreadsheets*, *database*, gambar, dan data suara. Sedangkan arsip didefenisikan oleh Deserno dan Kynaston (2005) sebagai dokumen dalam semua media yang mempunyai nilai historis atau hukum sehingga disimpan secara permanen.

Faktor penting yang perlu diperhatikan adalah alasan di balik penyimpanan dokumen tersebut. Dalam manajemen arsip manual, di simpan berarti menempatkan dokumen dalam sistem kearsipan yang bisa ditemukan kemudian bila dibutuhkan. Namun dengan teknologi digital, hasil yang sama akan di capai dengan mentransfer dokumen elektronis dari proses administrasi manual ke dalam sistem penyimpanan.
Kennedy dan Schauder (1998) menjelaskan bahwa pada setiap dokumen dan arsip akan terdiri atas:

a. **Isi**, yaitu informasi yang terdapat pada arsip berupa ide atau konsep, fakta tentang suatu kejadian, orang, organisasi maupun aktivis lain yang di rekam dalam arsip tersebut.
b. **Struktur,** merupakan atribut fisik (ukuran dan gaya huruf, spasi, margin, dan lambang organisasi) dan logis (logika dibalik pembuatan dokumen tersebut) dari suatu arsip. Misalnya, struktur surat akan terdiri dari header (nama pengirim, tanggal, judul surat, dan penerima), tubuh surat (isi dari maksud dibuatnya surat) dan otentifikasi (tandatangan sipembuat surat).
c. **Konteks,** menjelaskan "mengapa" dari suatu arsip.

2. 2 Kriptografi

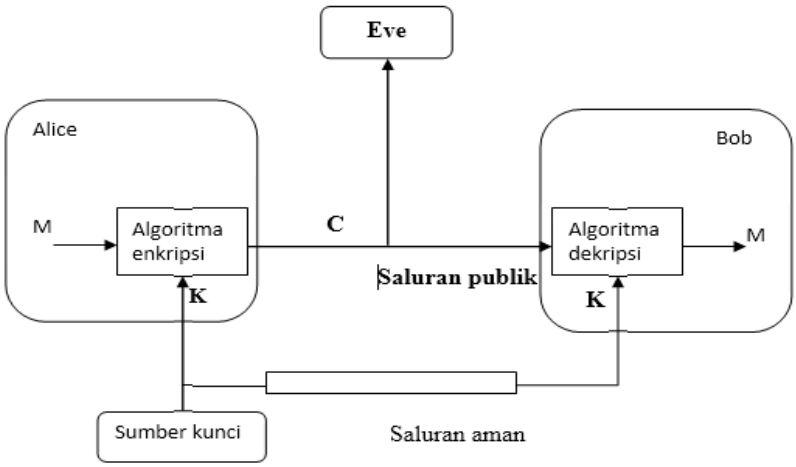
Kriptografi pada awalnya dijabarkan sebagai ilmu yang mempelajari bagaimana menyembunyikan pesan. Namun pada pengertian moderen kriptografi adalah ilmu yang berdasarkan pada teknik matimatika untuk berurusan dengan keamanan informasi seperti kerahasiaan, keutuhan dan otentikasi entitas. Jadi pengrtian kriptogradi moderen adalah tidak saja berurusan hanya dengan penyembunyian pesan namun lebih pada sekumpulan teknik yang menyediakan keamanaan informasi. Berikut ini adalah rangkuman beberapa mekanisme yang berkembang pada kriptografi moderen[6], [7].

a. Fungsi Hash. Fungsi hash adalah fungsi yang melakukan pemetaan pesan dengan panjang sembarang ke sebuah teks khusus yang di sebut *message digest* dengan panjang tetap. Fungsi hash umumnya di pakai sebagai nilai uji (*check value)* pada mekanisme keutuhan data.
b. Penyediaan dengan kunci simetrik (*symmetric key encipherment*). Penyediaan dengan kunci simetrik adalah penyediaan yang kunci enkripsi dan kunci dekripsi bernilai sama. Kunci pada penyandian simetrik diasumsikan bersifat rahasia hanya pihak yang melakukan enkripsi dan dekripsi yang mengetahui nilainya. Oleh karena itu penyediaan dengan kunci simetrik di sebut juga dengan penyediaan dengan kunci rahasia *secret key encrherment.*
c. Penyediaan dengan kunci asimetrik (*asymmetric key encipherment*). Penyediaan dengan kunci asimetrik atau sering di sebut dengan penyediaan kunci publik (*public key*) adalah penyediaan dengan kunci enkripsi dan dekripsi berbeda nilai. Kunci enkripsi yang juga di sebut kunci publik (*public key*) bersifat terbuka. Sedangkan, kunci dekripsi yang juga disebut kunci privat (*privat key*) bersifat tertutup atau rahasia.

2.3 Sistem Kriptografi

Sistem kriptografi terdiri dari 5 bagian yaitu [7], [8];

a. ***Plaintext***: Pesan atau data dalam bentuk aslinya yang dapat terbaca. *Plaintext* adalah masukan bagi algoritma enkripsi. Untuk selanjutnya digunakan istilah teks asli sebagai padanan kata *plaintext.*
b. ***Secret key***: *Secret key* yang juga merupakan masukan bagi algoritma enkripsi merupakan nilai yang bebas terhadap teks asli dan menentukan hasil keluaran algoritma enkripsi. Untuk selanjutnya digunakan istilah kunci rahasia sebagai padanan kata *secret key*.
c. ***Ciphertext***: *Ciphertext* adalah keluaran algoritma enkripsi. *Ciphertext* dapat di anggap sebagai pesan dalam bentuk tersembunyi. Algoritma enkripsi yang baik akan menghasilkan *ciphertext* yang terlihat acak. Untuk selanjutnya digunakan istilah teks sandi sebagai padanan kata *ciphertext.*
d. **Algoritma Enkripsi:** Algoritma enkripsi memiliki 2 masukan yaitu teks asli dan kunci rahasia. Algoritma enkripsi melakukan transformasi terhadap teks asli sehingga menghasilkan teks sandi.
e. **Algoritma Dekripsi:** Algoritma dekripsi memiliki 2 masukan yaitu teks sandi dan kunci rahasia. Algoritma dekripsi memulihkan kembali teks sandi menjadi teks asli bila kunci rahasia yang di pakai algoritma dekripsi sama dengan kunci rahasia yang di pakai algoritma enkripsi.



Gambar 1. Sistem Kriptografi Konvensional[6]

Sistem enkripsi harus memenuhi kaidah *correctness* yaitu untuk setiap $K \in K$ dengan K adalah himpunan kunci dan terdapat teks sandi hasil enkripsi teks asli m, c = *e*K (m) maka harus berlaku *d*K (c) = untuk semua kemungkinan teks asli.
Pada gambar II.1 kunci rahasia dibangkitkan oleh pembangkit kunci dan di kirim melalui saluran aman ke pihak penyandi (*encryptor*) maupun pemulih sandi (*decryptor*). Teks sandi di kirim melalui saluran umum sehingga ada pihak ketiga yang dapat membaca teks sandi itu.

2.4. Algoritma Tanda Tangan Digital Gost

Sejak ditemukannya oleh Diffie dan Hellman pada tahun 1973, banyak usulan untuk merealisasikan tanda tangan digital yang paling populer diantaranya RSA, ElGamal, Schnorr dan Fiat shamir. Beberapa organisasi mengusulkan standar tanda tangan yang berbeda-beda, sebagai contoh; spesifikasi ISO/IEC 9796 untuk tanda tangan digital, American National Standard x930-199x, RSA dan skema tanda tangan ElGamal[8]–[10].

Masyarakat perbankan Perancis telah menstadarisasi RSA. *National Institute of Standardization Technology* (NIST) mengusulkan suatu modifikasi ElGamal dan Schnorr sebagai standar tanda tangan digital (*digital signature Standard* (DSS)) dan skema tanda tangan didasarkan pada kurva elips yang sampai saat ini menjadi standar *Institute of Electrical and Electronics Engineers* (IEEE). Pada tahun 1999 federasi Rusia mengeluarkan standar tanda tangan digital GOST. Standar ini melingkupi sistem penghasil kunci, penghasil tanda tangan dan verifikasi. Standar ini juga menggunakan fungsi hash yang distandarisasi di bawah referensi GOST, fungsi hash didasarkan pada klasifikasi farsial, blok cipher GOST. Prosedur penghasil tanda tangan dan verifikasi menggunakan metode DSA yang dimodifikasi[11].

Parameter Dalam Algoritma GOST

1. Parameter Gost
   Parameter GOST dapat digambarkan dalam perhitungan sebagai berikut:
   a. P adalah bilangan prima
   b. Q adalah bilangan prima, dan merupakan faktor dari P-1
   c. A adalah bilangan yang lebih kecil dari p-1 sehingga A^Q mod P=1
   d. X adalah bilangan yang lebih kecil dari q di mana 0<X<Q
   e. Y = A^X mod P
   f. M adalah dokumen yang akan di beri tanda tangan.
   g. K = bilangan bulat yang dibangkitkan bilangan random di mana 0< K <Q

2. Pembangkit Sepasang Kunci
   Algortima pembangkit kunci sama dengan sistem kriptografi GOST, yaitu menghasilkan kunci publik dan kunci privat. Implementasi skema digital signature GOST. Fungsi pembangkit kunci menggunakan objek GOST, yaitu objek yang mempresentasikan sistem kriptografi asimetrik GOST dengan mengembalikan fungsi pembangkit kunci.
   a. Pilih bilangan prima P dan Q, di mana (P-1) mod Q = 0
   b. Hitung a di mana A<P-1 sehingga A^Q mod P = 1
   c. Tentukan kunci privat X<Q
   d. Hitung kunci publik y = A^X mod p

3. Pemberian Tanda Tangan Digital (*signing*)
   Algoritma sign menerima sebuah masukan pesan m, kunci privat dan kunci publik GOST. Algoritma sign menggunakan perhitungan eksponsial modular untuk mendapatkan signature r dan s.
   Algoritma sign diimplementasikan sebagai sebuah fungsi sign dengan masukan sebuah pesan m, kunci privat GOST dan kunci publik GOST. Fungsi sign mengembalikan larik byte yang merupakan signature skema digital signature GOST.
   a. Ubah pesan m menjadi message digest dengan fungsi SHA-1 menghasilkan SHA (m)
   b. Tentukan bilangan acak K<Q
   c. Diperoleh R dan S yang merupakan tanda tangan dari dokumen M dengan rumusan sebagai berikut:
   R = (A^K mod P) mod Q
   S = (x^r + k (H(M)) mod Q
   jika tandatangan yang dihasilkan benar maka nilai R dan atau S tidak mungkin 0.
   d. Di kirim dokumen beserta tanda tangan R dan S

4. Verifikasi Tanda Tangan Digital (*verify*)
   Pada algoritma ini penerima mendapatkan (M, R dan S) dari pengirim pesan. Penerima memverifikasi pesan (M, R dan S) dengan menjalankan algoritma Verify yang diberikan oleh algoritma sign.
   Algoritma Verify digital signature GOST diimplementasikan sebagai fungsi verify, fungsi verify menerima masukan dari pesan m dalam larik byte, signature r dan s dalam larik byte dan sebuah objek publik kunci GOST. Fungsi verify mengembalikan nilai boolean di terima.
   Sebelum diverifikasi, harus dipastikan tersedia kunci publik pengirim (Y), nilai P-Q dan A beserta pesan yang bertandatangan R dan S. Verifer memeriksa terlebih dahulu apakah 0<R<Q dan 0<S<Q kemudian menghitung:
   V = H $(M)^{Q-2}$ mod Q
   Z1 = (sv) mod Q
   Z2 = ((Q-R * V) mod Q
   U = ((A^{z1} * Y^{z2}) mod P) mod Q
   Jika hasil U = R ⟶ *ferivied*

Contoh Kasus Algoritma Gost

Berikut contoh kasus pembuatan kunci, tanda tangan digital, dan verifikasi menggunakan algoritma gost.

a. Pembuatan kunci privat dan kunci publik dengan algoritma gost

1. cari nilai P dan Q yang merupakan bilangan prima yang dilakukan secara acak.
   - P – 1= Q * cari
   - (P-1) mod Q = 0

- Cari = $\frac{P-1}{Q} = \frac{509-1}{127}$ = 4 (bilangan bulat)

2. Mencari nilai A.
   Nilai A diperoleh dengan syarat:
   - A < P-1 (nilai A harus lebih kecil dari P-1)
   - A^Q mod P = 1

   Contoh diberikan nilai A = 23 → memenuhi syarat <P-1
   → 23^{127} mod 509 = 1 → memenuhi syarat A^Q mod P = 1
3. Mencari nilai X, di mana nilai X merupakan bilangan acak dengan syarat:
   - X harus lebih besar dari 0 (X<0)
   - X harus lebih kecil dari Q (X<Q)

   Dimisalkan X = 89 → (89 memenuhi syarat 1 dan 2)
4. Mencari nilai Y dengan rumus Y = A^x mod P, di mana nilai A, X, P telah ditemukan maka:
   Y = A^x mod P
   Y = 23^{89} mod 509 = 293

   Setelah itu ditentukan kunci privat dan kunci publik yaitu:
   - Kunci privat (P, Q, A, X)
   - Kunci publik (P, Q, A, Y)

b. Pembuatan tanda tangan digital menggunakan algoritma Gost

Berdasarkan kunci privat yang telah diperoleh sebagai pembentuk tanda tangan digital, makan dapat dilanjutkan pada proses pembuatan tanda tangan digital dengan contoh berikut:

Dimisalkan diperoleh nilai hash (M) dari suatu dokumen adalah 4321 lalu dicari nilai k.

- Cari nilai K secara acak di mana nilai k<Q

Contoh, diberikan nilai K = 121

- Cari nilai R dan S
  - Untuk mencari R dirumuskan, R = (A^k mod P) mod Q
    = (23^{121} mod 509) mod 127
    = 103
  - Untuk mencari S dirumuskan, S = (X *R) + K ((H(m)) mod Q
    = (89*103) + 121 (4321) mod 127
    = 532008 mod 127
    = 5

c. Verifikasi menggunakan algoritma gost

algoritma verifikasi pada algoritma gost dirumuskan sebagai berikut:

- V = $H(M)^{Q-2}$ mod Q
  V = 4321^{125} mod 127
  V = 85

- Z1 = (S * V) mod Q
  Z1 = (5 * 85) mod 127
  Z1 = 44
- Z2 = ((Q – R) * V) mod Q
  Z2 = ((127-103) * 85) mod 127
  Z2 = 8
- U = (($A^{z1} * Y^{z2}$) mod P) mod Q
  U = (($23^{44} * 293^8$) mod 509) mod 127
  U = 103

- R = 103 dan U = 103
  Terbukti bahwa nilai R dan U adalah sama, maka dinyatakan data valid.

HASIL DAN PEMBAHASAN

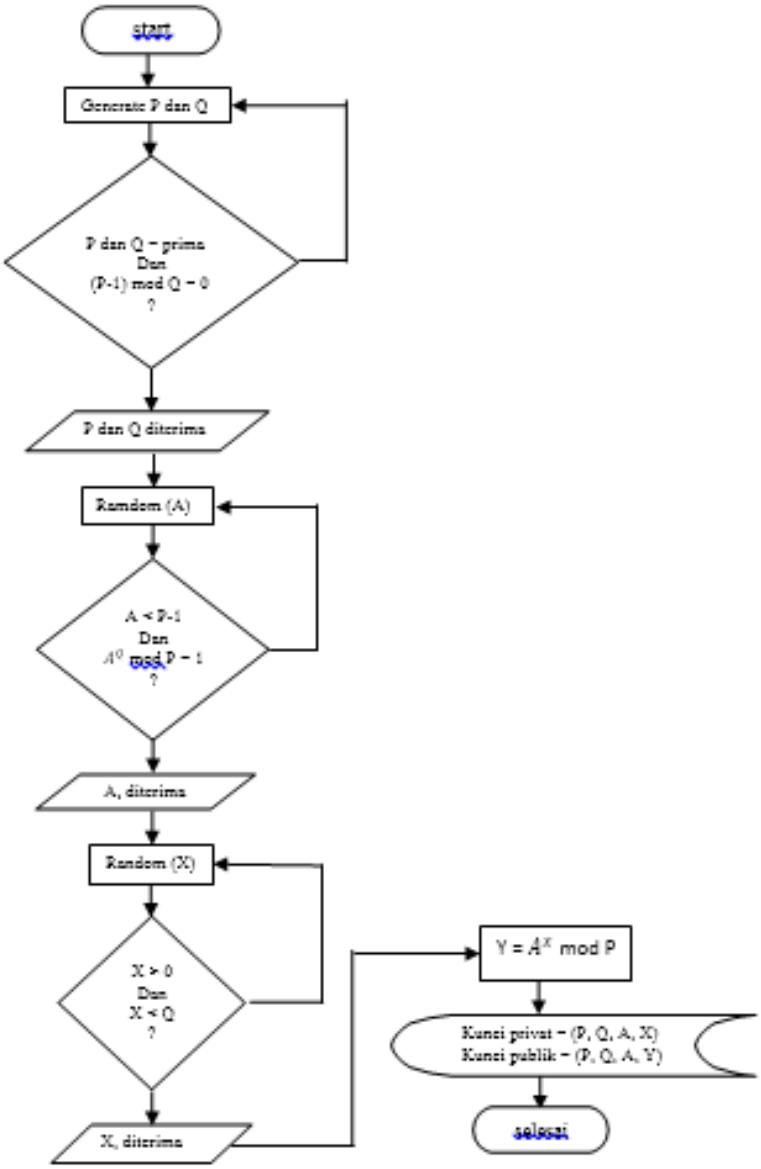
3.1. Flowchart Proses Pembentukan Kunci

Sepasang kunci diperlukan untuk membangkitkan nilai *digital signature* pada pesan dan memverifikasi keabsahan pesan. Kunci – kunci tersebut adalah kunci privat dan kunci publik. Tampa kunci-kunci tersebut pembentukan dan verifikasi digital signature tidak dapat dilakukan. Prosedur pembangkitan sepasang kunci pada algoritma GOST akan dijelaskan sebagai berikut:

a. Pilih bilangan prima p dan q, dimana (p-1) mod q=0
b. Hitung a dimana a<p-1 sehingga a^q mod p-1
c. Tentukan kunci privat x<q
d. Hitung kunci publik y=A^x mod p

Seperti yang ditunjukkan pada pada gambar 2. pada prosedur pembagkit sepasang kunci menghasilkan ;

a. Kunci publik dinyatakan sebagai (P, Q, A, Y)
b. Kunci privat dinyatakan sebagai (P, Q, A, X)



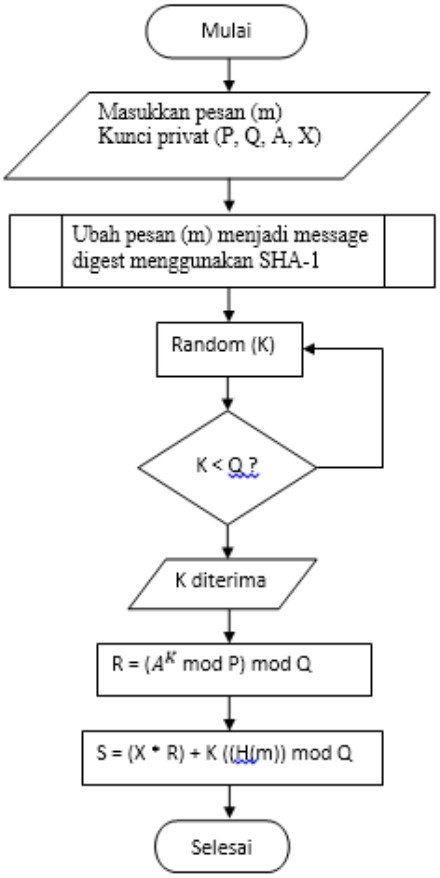
Gambar 2.. Flowchart Pembentukan Kunci Privat dan Kunci Publik

3.2. Flowchart Proses Pembuatan Tanda Tangan Digital

Tanda tanga digital (*digital signature*) merupakan protokol yang dijalankan untuk dapat memverifikasi keaslian dan keutuhan dari pesan yang akan dikirimkan dalam satu saluran komunikasi.
Prosedur pembuatan tanda tangan digital pada algoritma GOST meliputi:

1. Masukkan pesan dan kunci privat
2. Ubah pesan m menjadi message digest dengan fungsi SHA menghasilkan SHA(m) 160 bit
3. Tentukan bilangan acak k dimana k<q
4. Tandatangan dari pesan m adalah bilangan r dan s yang di dapat dari perhitungan:
   r = $(a^k$ mod p) mod q
   s = $(x^r$ + k (H(m)) mod q
   jika tanda tangan yang dihasilkan benar maka nilai r dan atau s tidak mungkin 0.
5. Diperoleh tanda tangan r dan s

Prosedur pembuatan tanda tanagn digital digambarkan pada gambar 3 berikut.



Gambar 3.. Flowchat Prosedur Pembentukan Tanda Tangan Digital

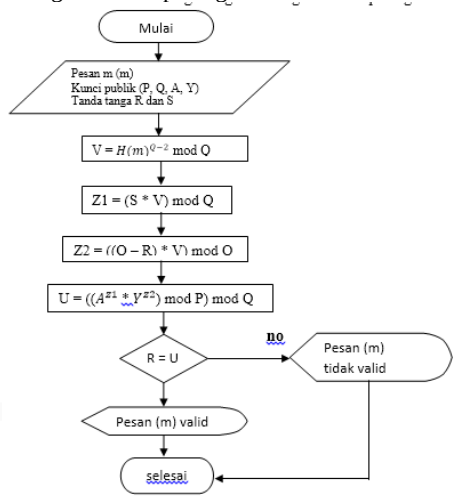
4. Flowchart proses verifikasi

Proses verifikasi adalah proses memastikan keaslian si pengirim. Artinya apakanh yang mengirimkan itu asli dari orang yang bersangkutan.

Prosedur verifikasi keabsahan tanda tangan digital pada algortima GOST meliputi:

1. Masukkan pesan yang sudah di tanda tangani kunci publik
2. Hitung
   v = H $(m)^{q-2}$ mod q
   z1 = (sv) mod q
   z2 = ((q-r * v) mod q
   u = $((a^{z1} * y^{z2})$ mod p) mod q
3. Jika nilai R = U maka data valid

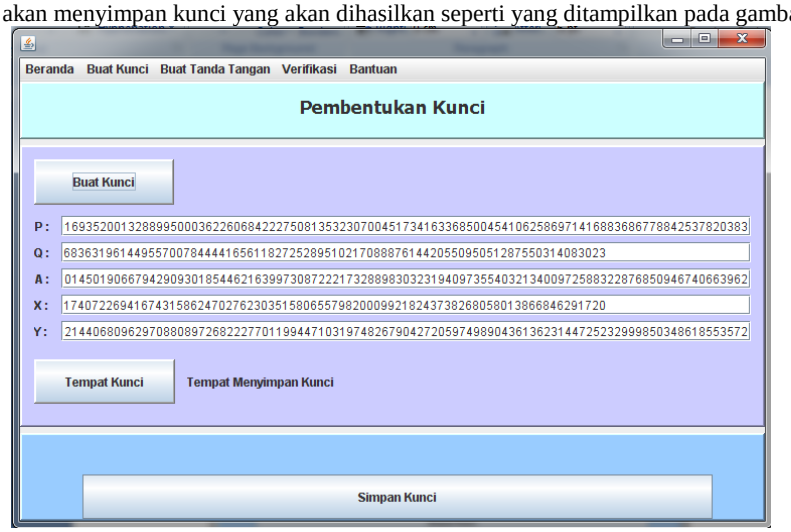
Prosedur verifikasi tanda tangan digital ini digambarkan pada gambar 4 berikut.



Gambar 4. Flowchart Verifikasi Tanda Tangan Digital

IMPLEMENTASI SISTEM

Pada halaman tampilan pembuatan kunci kita juga masih bisa mengakses ke halaman lainnya seperti beranda, buat kunci, buat tanda tangan, verifikasi dan bantuan. Proses pembuatan kunci dapat dilakukan dengan cara meng klik button buat kunci, maka secara otomatis akan diperoleh nilai P, Q, A, X, Y. Setelah itu diberikan juga sebuah button dengan nama tempat kunci yang difungsikan sebagai opsi dimana kita akan menyimpan kunci yang akan dihasilkan seperti yang ditampilkan pada gambar 5 berikut.



Gambar 5..Pembentukan Kunci

Pada form tampilan pembuatan kunci diatas ditampilkan hasil dari nilai dari P, Q, A, X, Y yang secara sederhana dapat di peroleh dengan rumusan dan ketentuan sebagai berikut:

h. P adalah bilangan prima
i. Q adalah bilangan prima dan merupakan faktor dari p-1
j. A adalah bilangan yang lebih kecil dari p-1 sehingga a^q mod p=1
k. X adalah bilangan yang lebih kecil dari q di mana 0<x<q
l. Y = a^x mod p

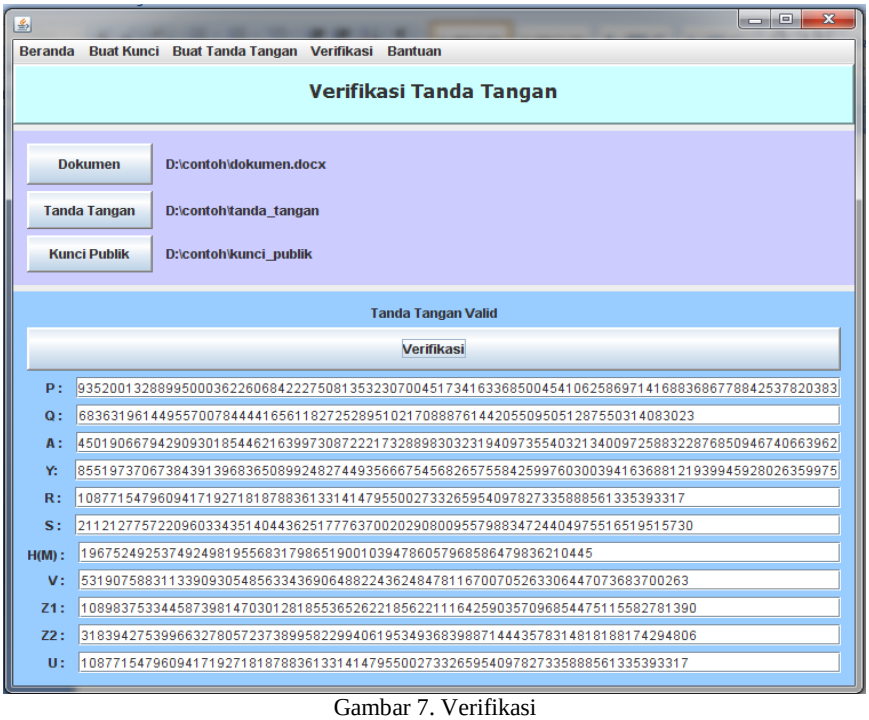
Pada proses penyimpanan kunci, kunci yang disimpan dapat diletakkan kedalam memori penyimpanan yang ada dikomputer atau memori penyimpanan lainnya. Dalam hal ini dicontohkan kunci tersebut akan disimpan di dalam drive D dengan nama folder "contoh" seperti yang ditunjukkan gambar 6 berikut

1. Klik Buat Kunci

Gambar 6. Klik Buat Kunci

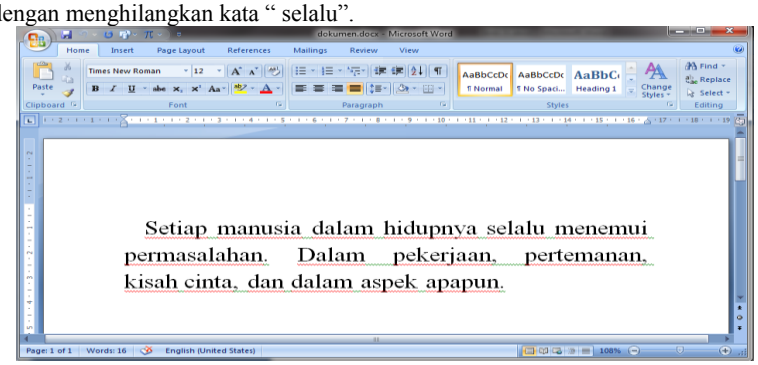
2. Verifikasi

Pada halaman tampilan verifikasi kita juga masih bisa mengakses ke halaman lainnya seperti beranda, buat kunci, buat tanda tangan, verifikasi dan bantuan. pada tampilan halaman verifikasi disediakan button dengan nama dokumen, tanda tangan dan kunci publik dimana kita akan memilih dokumen yang akan di verifikasi dan memilih tanda tangan dan kunci publik yang sudah disimpan di dalam komputer. Setelah dokumen,tanda tangan, dan kunci publik sudah dipilih, kita dapat langsung memverifikasi dokumen tersebut dengan cara mengklik pada button verifikasi. Apabila dokumen yang akan diverifikasi belum mengalamai perubahan, maka akan muncul pesan "tanda tangan valid" begitupun sebaliknya seperti yang digambarkan pada gambar 7 berikut.

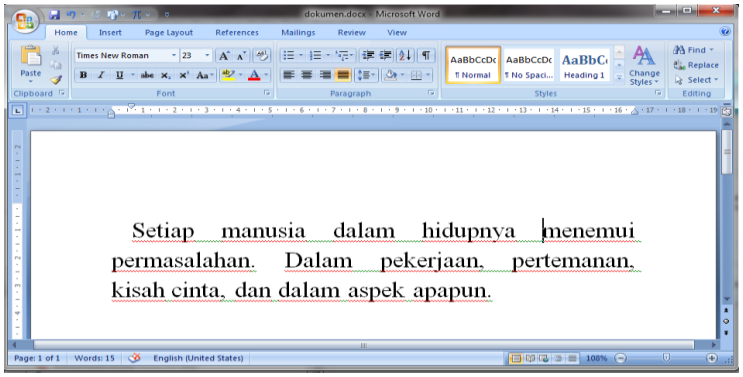


Gambar 7. Verifikasi

Proses verifikasi merupakan proses validasi dari dokumen menggunakan tanda tangan dan kunci publik dari pengirim dokumen. Pada proses ini dapat diketahui apakah suatu dokumen yang telah di beri tanda tangan digital telah mengalami perubahan isi pada saat proses pengiriman maupun proses lain sebelum sampai di tangan penerima.
Untuk membuktikannya dapat kita melakukan perubahan dokumen seperti yang ditunjukan pada gambar dokumen pertama menjadi gambar dokumen kedua dengan menghilangkan kata " selalu".

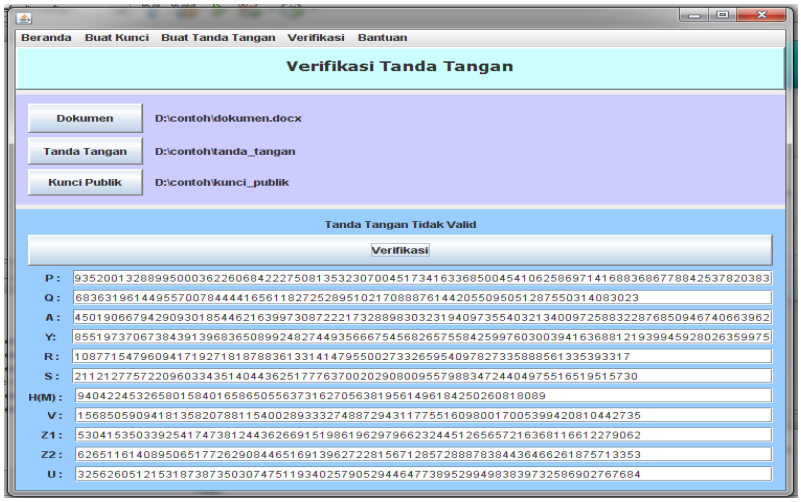


Gambar 8 Gambar Dokumen Pertama



Gambar 9 Dokumen Kedua

Setelah dilakukan perubahan pada dokumen, maka dilanjutkan dengan melakukan verifikasi ulang pada dokumen yang telah diubah dengan cara menjalankan form verifikasi melalui aplikasi tanda tangan digital. Proses verifikasi ulang tersebut dapat dilihat pada gambar 10 berikut



Gambar 10. Proses Verifikasi Ulang

Pada proses verifikasi ulang yang pada gambar 10 terlihat bahwa tanda tangan yang dibuat pada dokumen yang diverifikasi tidak valid karna telah terjadi perubahan isi pada dokumen.

Salah satu syarat dalam algoritma ini menyatakan sebuah dokumen valid apa bila nilai yang dihasilkan pada R sama dengan nilai yang dihasilkan pada U. Pada gambar 10 jelas terlihat bahwa nilai $R \neq U$.

R = 10877154796094171927181878836133141479550027332659540978273358885613353933 17

U = 3256260512153187387350307475119340257905294464773895299498383 9732586902767684

KESIMPULAN

Berdasarkan pembahasan mengenai aplikasi tanda tangan digital dengan algoritma gost untuk keamanan pengiriman file dokumen dapat diambil kesimpulan:

1. Pada implementasi tanda tangan digital dengan metode gost maka dapat memenuhi kebutuhan keamana e-dokumen dalam hal:
2. Otentikasi pesan, mewujudkan layanan otentikasi pesan yang memungkinkan dokumen hanya dapat di verifikasi menggunakan kunci publik pasangan kunci privat yang di buat oleh si pembuat tanda tangan.
3. Keutuhan atau keotentikan (*integrity*) dokumen yang dikirimkan dapat dijamin keutuhannya dengan hash SHA-1 dari dokumen tersebut.
4. Penyangkalan (*non-repudiaton*), memungkinkan *verifier* dapat membuktikan bahwa si pengirim dokumen merupakan pengirim dokumen yang sebenarnya.

DAFTAR PUSTAKA

[1] T. Limbong, "Peran dan Fungsi Komputer dalam Mendukung Prestasi Akademik Mahasiswa STMIK Budi Darma Medan," *Inf. dan Teknol. Ilm.*, vol. 3, no. 1, pp. 138–143, 2014.

[2] H. M. Jogiyanto, *Analisis dan Desain (Sistem Informasi Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis)*. Yogyakarta: Penerbit Andi, 2017.

[3] Komunikasi, "17 Dampak Negatif dari Internet - PakarKomunikasi.com," 2019. https://pakarkomunikasi.com/dampak-negatif-dari-internet (accessed Sep. 17, 2019).

[4] F. Nurhasanah and R. Sulaiman, "PEMBUATAN TANDA TANGAN DIGITAL MENGGUNAKAN DIGITAL SIGNATURE ALGORITHM," May 2013. Accessed: Apr. 17, 2021. [Online]. Available: https://jurnalmahasiswa.unesa.ac.id/index.php/mathunesa/article/view/2555.

[5] A. M. Andalan, "Kedudukan Tanda Tangan Elektronik dalam Transaksi Teknologi Finansial," *Jurist-Diction*, vol. 2, no. 6, pp. 1931–1950, 2019.

[6] R. Sadikin, *Kriptografi untuk keamanan jaringan*. Penerbit ANDI, 2012.

[7] R. Munir, "Kriptografi," *Inform. Bandung*, 2006.

[8] T. Limbong *et al.*, "The implementation of computer based instruction model on Gost Algorithm Cryptography Learning," in *IOP Conference Series: Materials Science and Engineering*, 2018, vol. 420, no. 1, p. 12094.

[9] W. Dwiono and T. Hartanto, "Penerapan Algoritma Kriptografi ElGamal Untuk Pengaman File Citra," *J. EECCIS*, vol. 4, no. 1, pp. 8–11, 2010.

[10] J. K. Azhar and S. Yuliany, "Implementasi Algoritma RSA (Rivest, Shamir dan Adleman) untuk Enkripsi dan Dekripsi File .pdf," no. December, 2019.

[11] T. Arianti and B. Nadeak, "Perancangan Aplikasi Pembelajaran Kriptografi Algoritma GOST dengan Menggunakan Metode Computer Based Instruction ," Medan, Jan. 2019. Accessed: Apr. 20, 2021. [Online]. Available: http://ejournal.ust.ac.id/index.php/KAKIFIKOM/article/view/626.