Sistem *Mailtracking* Dengan *Digital Signature*

Wisnu Wendanto
Program Studi Sistem Komputer STMIK-AUB Surakarta
wwendanto9@gmail.com

Abstract - Mailing administration system is often called as mailtracking system is a system used for document management such as the disposition letter, official letters, filing, reporting, and so forth. The documents are in addition to functioning as a means of communication between personal, unit, also serves as formal evidence of activities that can be accounted for (legal aspects). On the other hand, the management and archiving is often a problem that is quite time-consuming and labor intensive. Document traffic can become very large and require a filing system that is also greater (aspect of efficiency).

This research aims to develop an application system mailtracking using the digital signature for data security are expected to support the process of routing and monitoring letter so as to create legitimacy and efficiency in the administration of the correspondence and facilitate in management of the environment regular mailing in STMIK-AUB Surakarta. In this research the data entered is sending one or more to generate the or more digital signature to the letter. Then the letter, the digital signature and the message digest are sent to one or more recipients. After that the recipient will verify if the result of the verification is valid, it means that the letter is valid and the sender of the letter is true. Conversely, if the results are not valid means of the letter or sending unauthorized and not the actual sender of the letter.

***Keyword** : mailtracking, data security, digital signature*

Abstrak - sistem administrasi Mailing sering disebut sebagai sistem pelacakan mail sistem yang digunakan untuk manajemen dokumen seperti surat disposisi, surat-surat resmi, pengajuan, pelaporan, dan sebagainya. Dokumen-dokumen yang selain berfungsi sebagai sarana komunikasi antara pribadi, unit, juga berfungsi sebagai bukti formal kegiatan yang dapat dipertanggungjawabkan (aspek hukum). Di sisi lain, manajemen dan pengarsipan sering merupakan masalah yang cukup memakan waktu dan tenaga. Dokumen lalu lintas dapat menjadi sangat besar dan memerlukan sistem pengarsipan yang juga lebih besar (aspek efisiensi).

Penelitian ini bertujuan untuk mengembangkan pelacakan email sistem aplikasi menggunakan tanda tangan digital untuk keamanan data diharapkan untuk mendukung proses routing dan pemantauan surat sehingga tercipta legitimasi dan efisiensi dalam administrasi surat menyurat dan memudahkan dalam pengelolaan lingkungan mailing biasa di STMIK-AUB Surakarta. Dalam penelitian ini data yang dimasukkan mengirimkan satu atau lebih untuk menghasilkan atau tanda tangan digital lebih untuk surat itu. Kemudian surat, tanda tangan digital dan message digest dikirim ke satu atau lebih penerima. Setelah itu penerima akan memverifikasi jika hasil verifikasi tersebut valid, itu berarti bahwa surat tersebut valid dan pengirim surat itu benar. Sebaliknya, jika hasilnya tidak berarti sah surat atau mengirim sah dan tidak pengirim sebenarnya dari surat itu.

Kata Kunci: mailtracking, keamanan data, tanda tangan digital

1. Pendahuluan

Pada masa kini, teknologi informasi sudah menjadi kebutuhan utama. Hampir semua orang menggunakan teknologi informasi dalam kehidupan mereka sehari-hari, baik untuk keperluan pendidikan, bisnis, hiburan, dan lain-lain. Pada saat ini penerapan teknologi informasi sudah mencakup di semua aspek kehidupan masyarakat, sebagai contoh adalah untuk sistem administrasi surat atau yang sering disebut *mailtracking system*. Administrasi surat terkait dengan dua hal, *legalitas* dan *effisiensi*. Baik itu dalam bentuk disposisi, surat dinas, pengarsipan, laporan, dan lain sebagainya. Dokumen-dokumen tersebut selain berfungsi sebagai sarana komunikasi antar personal, unit, juga berfungsi sebagai bukti formal kegiatan yang dapat dipertanggungjawabkan (*aspek legalitas*). Di sisi lain, pengelolaan dan pengarsipan sering menjadi permasalahan yang cukup menyita waktu dan tenaga. Lalu lintas dokumen dapat berkembang menjadi sangat besar sehingga memerlukan suatu sistem pengagendaan dan pengarsipan yang juga semakin besar (*aspek effisiensi*).

Pengelolaan dan pengarsipan surat masih banyak yang dilakukan secara manual karena data yang diproses masih berupa dokumen, maka akan beresiko dokumen hilang dan memakan waktu yang lama karena proses birokrasi yang ada di instansi itu sendiri. Karena itu perlu dibangun suatu sistem untuk pengelolaan dan pengarsipan berkas-berkas surat yang berupa aplikasi berbasis *multiuser* dengan menerapkan sistem keamanan data. Keamanan data dan informasi merupakan bagian yang sangat penting dari sebuah sistem dalam jaringan komputer terutama yang

terhubung ke internet. Saat pengiriman dokumen, seseorang bisa saja dengan ilegal mengubah isi dokumen itu tanpa diketahui pengirim atau penerima. Tanpa fasilitas keamanan yang baik, penerima dokumen tersebut dapat mencurigai adanya perubahan yang terjadi pada surat, sehingga legalitasnya surat dipertanyakan.

Salah satu cara untuk mencegahnya adalah dengan membuat suatu tanda khusus yang memastikan bahwa data tersebut adalah data yang benar. Untuk itu dapat digunakan salah satu teknologi keamanan jaringan yang disebut *Digital Signature*. *Digital Signature* diartikan sebagian orang sebagai bagian dari tanda tangan elektronik. Tanda tangan elektronik berarti sebuah suara elektronik, simbol, atau proses, secara logika dihubungkan dengan kontrak atau lainnya dan dieksekusi atau diadopsi oleh seseorang dengan maksud menandai rekaman tersebut. Sedangkan *digital signature* adalah sebuah tandatangan yang berbasiskan skema kriptografi. *Digital signature* memiliki tiga dari empat aspek keamanan yang dimiliki kriptografi yaitu: integritas data, otentikasi dan *non-repudiation*.

2. Kerangka Teori

2.1. Tinjauan Pustaka

Menurut Maharani, S dkk (2009) pada publikasi yang berjudul Implementasi Perangkat Lunak Penyandian Pesan Menggunakan Algoritma RSA, dalam penelitian tersebut merealisasikan suatu perangkat lunak enkripsi dan dekripsi pesan dengan menggunakan algoritma RSA. Pengiriman pesan lewat *email* memungkinkan pesan dapat dibajak oleh orang yang tidak berwenang. Pada masalah keamanan pesan, terdapat dua permasalahan utama yang mesti diperhatikan oleh pengguna yaitu masalah privasi *(privacy)* dan keautentikan *(authentication)*. Privasi mengandung arti bahwa pesan yang dikirimkan hanya dapat dimengerti informasinya oleh penerima yang sah, sedangkan keautentikan mencegah pihak ketiga untuk mengirimkan pesan yang salah atau mengubah pesan yang dikirimkan. Untuk mengatasi masalah keamanan pesan tersebut diperlukan keamanan pesan yang bisa menutupi kelemahan pesan tersebut. Penelitian ini menghasilkan perangkat lunak bantu penyandian pesan menggunakan algoritma RSA. Perangkat lunak ini juga terintegrasi dengan aplikasi pengiriman pesan, sehingga memudahkan para pengguna untuk penyandian pesan ketika akan dikirimkan melalui elektronik mail.

Menurut Shen, G dkk (2009), pada publikasi yang berjudul *on Fast Implementation of* RSA *With* JAVA. Penelitian ini menitikberatkan pada implementasi algoritma RSA untuk transaksi di *e-commerce* menggunakan JAVA. Metode RSA dalam implementasi diperlukan kecepatan sangat tinggi, dapat dikembangkan dengan bahasa C, dan kekurangan adalah bahwa kita harus menerapkan semua algoritma termasuk penggunaan integer besar, membuat bilangan prima besar, dan metode komputasi untuk menfaktorkan dua bilangan bulat yang besar. Dalam lingkungan *e-commerce*, sistem perdagangan harus dijalankan pada sistem operasi yang berbeda platform seperti Windows, Unix, dan Linux. Jika sistem ini dikembangkan dengan bahasa C, hal itu akan memunculkan masalah bahwa sistem tidak dapat berjalan di semua platform meskipun sistem dikembangkan dengan bahasa C mudah untuk diaplikasikan. Sebagai tipe data yang sama memiliki panjang yang berbeda pada platform yang berbeda, ini membuat sulit untuk diaplikasikan. Solusi memecahkan masalah lintas platform, bisa menggunakan bahasa java. Java adalah bahasa berorientasi objek, dan sistem dikembangkan dengan bahasa java dapat dijalankan pada semua platform. Meskipun efisiensi eksekusi bahasa java lebih lambat dibandingkan dengan bahasa C dan C + + , dikatakan bahwa kecepatan java hanya sepersepuluh dari C, kecepatan adalah sekitar sepertiga dari C dan C + +. Metode RSA, perlu kelas untuk mencari integer yang panjang, dan bahasa java menyediakan kelas BigInteger untuk mendefinisikan integer panjang . BigInteger berisi metode operasi dasar untuk penambahan, pengurangan, perkalian, dan pembagian, dan selain itu, juga menyediakan metode untuk menghasilkan prima besar, metode menghitung faktor bilangan, metode menghitung modul invers, dan sebagainya. Berdasarkan analisis tersebut, ada tiga kelas didalam metode RSA. Yang pertama adalah kelas RSA*Core* untuk implementasi algoritma inti, kelas yang kedua adalah *public* RSA yang merupakan *subclass* dari RSA*Core*, dan memiliki dua metode enkripsi dan verifikasi tanda tangan, yang kelas ketiga adalah *private* RSA yang merupakan *subclass* dari RSA*Core*, dan memiliki dua metode dekripsi dan tanda tangan. RSA*Core* kelas memiliki dua sifat yaitu N dan Kunci. Kunci publik maupun kunci privat adalah *array integer* dari presentasi *bit*-nya.

2.2. Algoritma RSA

Algoritma RSA dibuat oleh 3 orang peneliti dari MIT (*Massachussets Institute of Technology*) pada tahun 1976, yaitu Ron Rivest, Adi Shamir, dan Leonard Adleman. Algoritma RSA termasuk algoritma asimetri, yaitu algoritma yang memiliki 2 kunci, kunci publik dan kunci privat. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan

yang besar menjadi faktor-faktor prima. Pemfaktoran dilakukan untuk memperoleh kunci privat. Selama pemfaktoran bilangan besar menjadi faktor-faktor prima belum ditemukan algoritma yang bagus, maka selama itu pula keamanan RSA tetap terjamin (Delfs, Hans dkk, 2007).

2.2.1. Proses Pembuatan Kunci

Dalam membuat suatu sandi, RSA mempunyai cara kerja dalam membuat kunci publik dan kunci privat adalah sebagai berikut:

1. Pilih dua bilangan prima p dan q secara acak , p ≠ q. Bilangan ini harus cukup besar (minimal 100 digit).
2. Hitung $n = p \times q$. Bilangan n disebut parameter sekuriti.
3. Hitung $\Phi(n) = (p - 1)(q - 1)$.
4. Pilih bilangan bulat (integer) antara satu dan $\Phi(n)$ ($1 < e < \Phi(n)$) yang tidak mempunyai faktor pembagi dari $\Phi(n)$ atau gcd $(e, \Phi(n))$. Langkah ini dapat dihitung dengan algoritma *Euclidean*
5. Hitung $d = e^{-1} \bmod \Phi(n)$
6. (e, n) merupakan kunci publik
7. (d, n) merupakan kunci privat

Bilangan prima *p* dan *q* harus berupa bilangan yang sangat besar, disarankan *p* maupun *q* bilangan desimal 100 bit. Sehingga akan menghasilkan nilai *n* 200 bit. (Rivest). Hal ini dilakukan dengan tujuan mempersulit upaya menghitung kunci privat (*d*) dari kunci publik (*e* dan *n*) yang telah diketahui. Karena untuk mendapatkan nilai *d*, harus dicari nilai *p* dan *q* terlebih dahulu. Dan kedua nilai ini harus diperoleh dengan memfaktorkan *n*. Jika *n* terlalu kecil maka mudah untuk difaktorkan.

2.2.2. Proses Enkripsi Pesan

Proses enkripsi dilakukan oleh pihak pengirim, dalam hal ini adalah A. Seluruh perhitungan pemangkatan bilangan modulo dilakukan menggunakan metode *fast exponentiation*. Proses enkripsi RSA dijelaskan sebagai berikut :

1. Ambil kunci publik (*e*, *n*).
2. Pilih *plainteks M*, dengan $0 \le M \le n - 1$.
3. Hitung $\boldsymbol{C = M^e}$

Diperoleh *cipherteks C*, dan kirimkan kepada B.

2.2.3. Proses Dekripsi Pesan

Berikut ini adalah proses dekripsi RSA. Dilakukan oleh pihak penerima *cipherteks*, yaitu

1. Ambil kunci privat (*d, n*)
2. Hitung $\boldsymbol{M = C^d \bmod n}$.

2.3. Fungsi Hash

Suatu *hash function* adalah sebuah fungsi matematika, yang mengambil sebuah panjang *variable string input*, yang disebut pre-*image* dan mengkonversikannya ke sebuah *string output* dengan panjang yang tetap dan biasanya lebih kecil yang terdiri atas huruf dan angka yang terlihat acak (data biner yang ditulis dalam notasi heksadesimal), yang disebut *message digest* (Munir, 2004).

Fungsi hash satu arah *(one-way hash function)* adalah *hash function* yang bekerja satu arah, yaitu suatu *hash function* yang dapat menghitung *message digest* dari *pre-image*, tetapi sangat sukar untuk menghitung *pre-image* dari *message digest.* Sebuah fungsi hash satu arah, H(M) beroperasi pada suatu *pre-image* pesan M dengan panjang sembarang dan mengembalikan nilai hash h yang memiliki panjang tetap. Dalam notasi matematika fungsi hash satu arah dapat ditulis sebagai:

h = H(M), dengan h memiliki panjang b

Fungsi hash sangat berguna untuk menjaga integritas sebuah data. Sudah banyak algoritma *hash function* yang diciptakan, namun *hash function* yang umum digunakan saat ini adalah MD5 dan SHA (*Secure Hash Algorithm*).

2.3.1. Fungsi Hash SHA-1

Fungsi Hash (*hash function*) merupakan fungsi yang bersifat satu arah dimana jika dimasukkan data, maka akan menghasilkan sebuah "*checksum*" atau "*fingerprint*" dari kata tersebut. Sebuah pesan yang dilewatkan ke fungsi hash akan menghasilkan keluaran yang disebut *Message Authenticated Code* (MAC). Dilihat dari sisi matematik, *hash function* memetakan satu set data ke dalam sebuah set yang lebih kecil dan terbatas ukurannya. Fungsi hash satu arah mempunyai sifat sebagai berikut :

1. Diberikan M, harus mudah menghitung H(M) = h
2. Diberikan M, sangat sulit atau mustahil mendapatkan M sedemikian sehingga H(M) = h
3. Diberikan M, sangat sulit atau mustahil mendapatkan M' sedemikian sehingga H(M) = H(M'). Bila diperoleh pesan M' semacam ini maka disebut tabrakan (*collision*).
4. Sangat sulit atau mustahil mendapatkan dua pesan M dan M' sedemikian sehingga H(M) = H(M').

Sebuah fungsi hash satu arah H(M) beroperasi pada *pre-image* pesan M dengan panjang sembarang dan mengembalikan nilai hash h yang memiliki panjang tetap. Fungsi hash dikembangkan berdasarkan ide sebuah fungsi kompresi. Fungsi hash satu arah ini menghasilkan nilai hash berukuran n pada input sebesar b. Input tersebut berupa suatu fungsi kompresi blok pesan dan hasil blok pesan sebelumnya. Sehingga hash suatu blok M adalah :

$h_i = f(M_i, h_{i-1})$ …… (2.1)

dimana :
h_i = nilai hash saat ini
M_i = blok pesan saat ini
h_{i-1} = nilai hash blok pesan sebelumnya

SHA-1 adalah algoritma hash yang paling banyak digunakan publik (selanjutnya ditulis SHA). SHA merupakan keluarga fungsi hash satu arah. SHA menerima masukan berupa pesan dengan ukuran maksimum 2^{64} bit (2.147.483.648 gigabyte) dan menghasilkan *Message Digest* (MD) dengan panjang 160 bit. MD kemudian digunakan dalam RSA untuk menghitung tandatangan digital pesan tersebut. MD pesan yang sama dapat diperoleh oleh verifier ketika menerima pesan dari pengirim dengan cara memasukkan pesan tersebut pada fungsi SHA. SHA dikatakan aman karena secara matematis tidak mungkin menemukan dua pesan yang berbeda yang menghasilkan MD yang sama atau tidak mungkin menemukan pesan aslinya jika diberikan suatu nilai hash-nya.

Pesan M dengan panjang 1 bit dimana $0 \leq 1 \leq 2^{64}$. Algoritma pada SHA-1 menggunakan :

1. Pesan yang tersusun (*message schedule*) pada 80 dari 32 bit word.
2. Lima variabel kerja pada masing-masing 32 bit.
3. Nilai hash pada lima dari 32 bit word, hasil akhir adalah 160 bit MD.

Word pada *message schedule* diberi label W_0, W_1, …, W_{79}. Lima variabel kerja diberi label a, b, c, d dan e. Word dari nilai hash diberi label $H_0^{(i)}$, $H_1^{(i)}$, …, $H_4^{(i)}$ yang akan menampung nilai hash awal $H^{(0)}$ untuk diganti dengan nilai hash yang berurutan setelah blok pesan diproses $H^{(i)}$ dan berakhir dengan nilai hash final $H^{(N)}$. SHA-1 juga menggunakan *temporary* tunggal *word* T (FIPS PUB 183-3, 2008).

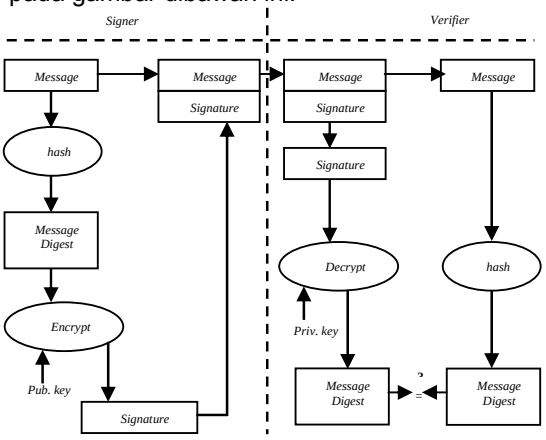
2.4. Tandatangan Digital

Tandatangan digital adalah mekanisme otentikasi yang mengijinkan pemilik pesan membubuhkan sebuah sandi pada pesannya yang bertindak sebagai tandatangan. Tandatangan dibentuk dengan mengambil nilai hash dari pesan dan mengenkripsi nilai hash pesan tersebut dengan kunci publik penerima pesan. Jika dalam proses pengiriman pesan saluran komunikasi yang digunakan sudah aman dan kunci hanya diketahui oleh pihak yang berhak, sekarang masalahnya siapa yang menjamin bahwa pesan yang dikirim memang berasal dari orang yang berhak. Atau bagaimana meyakinkan pihak yang akan menerima kiriman data bahwa memang benar-benar berasal dari pengirim aslinya. Untuk mengatasi validitas pengiriman tersebutlah digunakan teknologi Tandatangan Digital (Stallings, 2005).

Tandatangan digital berfungsi untuk melakukan validasi terhadap setiap data yang dikirim. Dalam pengiriman data, walaupun saluran komunikasi yang digunakan sudah sangat aman, tentu saja perlu diperhatikan validitasnya. Validitas tersebut berkaitan dengan pertanyaan apakah data yang sampai ke penerima dalam keadaan utuh dengan aslinya saat dikirim tanpa sedikitpun adanya gangguan-gangguan dari pihak lain. Tandatangan digital menggunakan algoritma yang disebut dengan istilah *hashing algorithm*. Fungsi tersebut akan menghasilkan sebuah kombinasi karakter yang unik yang disebut dengan *Message Digest*.

Keunikannya adalah jika di tengah perjalanan data mengalami modifikasi, penghapusan maupun di sadap diam-diam oleh *hacker* walaupun hanya 1 karakter saja, maka *message digest* yang berada si penerima akan berbeda dengan yang dikirimkan pada awalnya. Keunikan lainnya adalah *message digest* tersebut tidak bisa dikembalikan lagi ke dalam bentuk awal seperti sebelum disentuh dengan fungsi algoritma, sehingga disebutlah sebagai *one-way hash.*

Proses tandatangan digital dapat ditunjukkan pada gambar dibawah ini.



Gambar 1. Proses Tandatangan Digital (Munir, 2006)

Mekanisme kerja untuk menghasilkan tandatangan digital tersebut adalah sebagai berikut :

1. Proses *hashing algorithm* akan mengambil nilai hash dari pesan yang akan dikirim dan menghasilkan *message digest.* Kemudian *message digest* tersebut dienkripsi mengunakan kunci privat dan menghasilkan tandatangan digital.
2. Kemudian tandatangan digital tersebut dikirimkan bersama isi pesan tersebut.
3. Sesampainya di penerima, akan dilakukan proses *hashing algorithm* terhadap pesan tersebut seperti yang dilakukan saat

pengiriman. Dari proses tersebut menghasilkan *message digest* sekunder (MD').

4. Secara paralel *digital signature* yang diterima tadi langsung didekripsi oleh kunci publik. Hasil dekripsi tersebut akan memunculkan *message digest* yang serupa seperti *message digest* sebelum dienkripsi oleh pengirim pesan. *Message digest* disebut *message digest* primer (MD).
5. Proses selanjutnya adalah membandingkan *message digest* primer dengan *message digest* sekunder. Jika saja saat diperjalanan ada *hacker* yang mengubah isi pesan, maka *message digest* sekunder akan berbeda dengan *message digest* primer. Segera mekanisme tandatangan digital tersebut akan menyampaikan peringatan bahwa telah terjadi pengubahan isi pesan.

Tanda tangan digital mampu memenuhi tiga dari empat aspek keamanan kriptografi, yaitu aspek integritas data, otentikasi, dan antipenyangkalan.

Otentikasi dan integritas data dijelaskan sebagai berikut :

1. Apabila pesan M yang diterima sudah berubah, maka MD' yang dihasilkan dari fungsi hash berbeda dengan MD semula. Ini berarti pesan tidak asli lagi.
2. Apabila pesan M tidak berasal dari orang yang sebenarnya, maka *message digest* MD yang dihasilkan dari persamaan 3 berbeda dengan *message digest* MD' yang dihasilkan pada proses verifikasi (hal ini karena kunci publik yang digunakan oleh penerima pesan tidak berkoresponden dengan kunci rahasia pengirim).
3. Bila MD = MD', ini berarti pesan yang diterima adalah pesan yang asli dan orang yang mengirim adalah orang yang sebenarnya.

Pengirim pesan tidak dapat menyangkal pesan yang ia kirim. Andaikan pengirim menyangkal telah mengirim pesan, sangkalan dari pengirim dapat dibantah dengan cara : jika ia tidak mengirim pesan, berarti ia tidak mengenkripsi *message digest* dari pesan dengan kunci privatnya. Faktanya, kunci publik yang berkoresponden dengan kunci privat pengirim menghasilkan MD=MD' Ini berarti *message digest* memang benar dienkripsi oleh pengirim sebab hanya pengirim yang mengetahui kunci privatnya sendiri.

3. Metodologi Penelitian

3.1. Jalan Penelitian

Implementasi *digital signature* pada sistem keamanan data *mailtracking* dengan studi kasus administrasi surat di lingkungan STMIK-AUB Surakarta. Secara umum gambaran sistem adalah :

1. Surat masuk diterima dan dicatat oleh administrasi umum.
2. Dari administrasi umum surat akan di *post* kan kepada pimpinan.
3. Pimpinan membuka surat masuk dan langsung memberikan disposisi yang berupa instruksi kepada administrasi umum, jika pimpinan sudah melakukan disposisi maka status monitoring akan muncul status "sudah didisposisi".
4. Administrasi umum membuka surat disposisi dari pimpinan, lalu jika surat disposisi tersebut sudah dibuka dan dibaca maka status monitoring di account pimpinan akan muncul status "sudah dilaksanakan".
5. Untuk surat keluar, administrasi umum bisa membuat surat keluar setelah menerima instruksi dari pimpinan, dan untuk nomor surat dibuat otomatisasi dari sistem. Setelah di inputkan surat keluar akan dikirimkan ke penerima surat. Pada proses pengiriman surat tersebut proses hash dan pembuatan *digital signature* di *generate* oleh sistem sesuai dengan siapa pengirim dan penerima surat, selanjutnya data surat, *message digest*, dan *digital signature* akan tersimpan di *database*.
6. Pada penerima surat akan menerima surat yang dikirim tersebut lalu akan melakukan verifikasi surat menggunakan kunci privat penerima untuk membuktikan otentikasi pengirim surat dan mengambil surat tersebut.

3.2. Pengembangan Sistem Dengan Metode Sequential Model

Penelitian ini adalah untuk melakukan implementasi *digital signature* pada sistem keamanan data *mailtracking*. Pada penelitian ini dibangun suatu sistem dalam bentuk perangkat lunak *mailtracking* yang menerapkan *digital signature* untuk pengamanan data. Sedangkan metode yang digunakan untuk pembuatan perangkat lunak tersebut adalah metode *sequential linier* yaitu suatu metode pengembangan sistem yang bersifat sistematis dan terdiri dari 5 tahap yang saling terkait dan mempengaruhi, tahapan tersebut meliputi :

1. Analisa Kebutuhan
   Tahap analisa kebutuhan dilakukan pengumpulan informasi proses administrasi surat, proses tandatangan digital dan kebutuhan *user* secara lengkap. Hal ini dapat diperoleh dengan menyusun *scope of work software* yang akan dibuat beserta perangkat lunak yang dibutuhkan. Pada penelitian ini *scope of work software* yang dibuat meliputi :

a. Proses pendataan user berdasarkan level hak akses.
b. Proses pembangkitan kunci privat dan kunci publik.
c. Proses penyimpanan kunci privat dan kunci publik ke *database*.
d. Proses pendataan surat yang bersifat eksternal (berasal dari luar institusi).
e. Proses disposisi surat masuk.
f. Proses pembuatan surat (surat tugas, surat keputusan, undangan).
g. Proses pembuatan *digital signature.*
h. Proses hash yang akan menghasilkan *message digest*.
i. Proses kirim surat.
j. Proses penyimpanan data surat, *message digest* dan *digital signature* ke *database*.
k. Proses verifikasi *digital signature*.

Sedangkan kebutuhan perangkat lunak bahasa pemrograman yang dibutuhkan pada penelitian ini adalah yang memiliki kemampuan sebagai berikut :

a. Dapat melakukan proses komputasi dengan cepat, termasuk penanganan *input* dan *output*.
b. Memiliki ketelitian yang tinggi.
c. Bahasa pemrograman yang bersifat *open source* serta didukung oleh vendor dan komunitas sehingga memudahkan pengembangan sistem di kemudian hari.

Dari kebutuhan kemampuan tersebut bahasa pemrograman PHP dapat memenuhi semua kriteria tersebut, sehingga digunakan untuk pembuatan sistem pada penelitian ini.

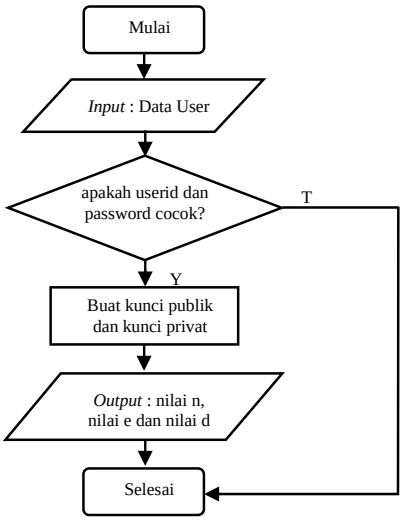
2. Desain (*design*)
   Desain atau perancangan sistem merupakan tahap penyusunan proses, data, aliran proses dan hubungan antar data yang paling optimal untuk menjalankan proses bisnis dan memenuhi kebutuhan-kebutuhan perusahaan sesuai dengan hasil analisa kebutuhan. Dokumentasi yang dihasilkan dari tahap desain sistem ini yaitu *Flow Chart* dan *Data Flow Diagram*, *Entity Relationship Diagram* (ERD) dan desain *interface*.
3. Pengkodean (*Coding*)
   Penulisan kode program atau *coding* merupakan tahap penerjemahan desain sistem yang telah dibuat ke dalam bentuk perintah-perintah yang dimengerti komputer. Pada penelitian ini *coding* dilakukan secara modular sesuai dengan pembagian blok-blok program pada tahap perancangan sistem, dengan menggunakan PHP dan MySQL.
4. Pengujian Program
   Pengujian program dilakukan untuk memastikan bahwa aplikasi yang dibuat telah sesuai dengan desainnya dan semua fungsi dapat dipergunakan dengan baik tanpa ada kesalahan. Pengujian program dilakukan untuk memastikan bahwa kerangka/skenario pengujian program dibuat dengan lengkap meliputi semua proses, kebutuhan dan pengendalian yang ada di dalam dokumen analisa kebutuhan dan desain sistem. Pada penelitian ini pengujian program dilakukan pada proses pembuatan dan pengiriman surat yang dilengkapi, pengiriman surat lebih dari 1 penerima, pembuatan dan verifikasi *digital signature*.
5. Penerapan Program
   Penerapan program merupakan tahap dimana peneliti menerapkan atau mengimplementasikan yang telah dibuat dan diuji ke obyek penelitian. Pada penelitian ini studi kasus pada administrasi surat di lingkungan STMIK-AUB Surakarta.

3.3. Perancangan Sistem Mailtracking dengan Digital Signature

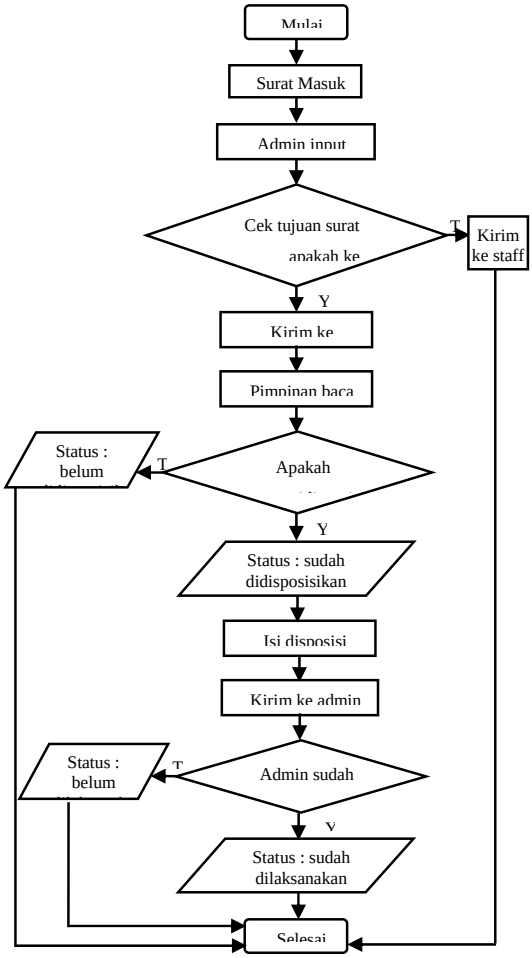
Tahap perancangan sistem *mailtracking* dengan sistem keamanan data menggunakan *digital signature* dan algoritma RSA meliputi penyusunan *Flow Chart*, *Data Flow Diagram*, *Entity Relationship Diagram*, dan perancangan *interface* sebagai deskripsi dari tahap penyusunan proses, data, aliran proses dan hubungan antar data *input* dan *output* yang ada di dalam sistem *mailtracking*.

3.3.1. Flow Chart Sistem Mailtracking

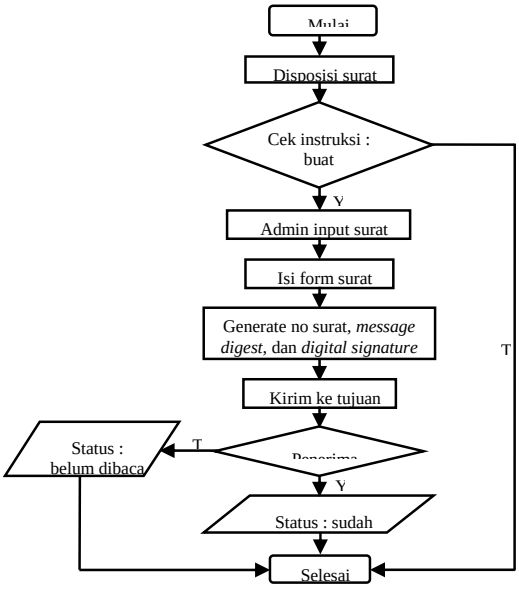
Flow Chart pada sistem *mailtracking* dengan sistem keamanan data *digital signature* dan algoritma RSA disusun sesuai tahapan proses yaitu terdiri dari Flow Chart Proses Pembangkitan Kunci, Flow Chart Proses Surat Masuk, Flow Chart Proses Kirim Surat, dan Flow Chart Verifikasi. Masing-masing bisa dilihat pada gambar 2, 3, 4, dan 5.



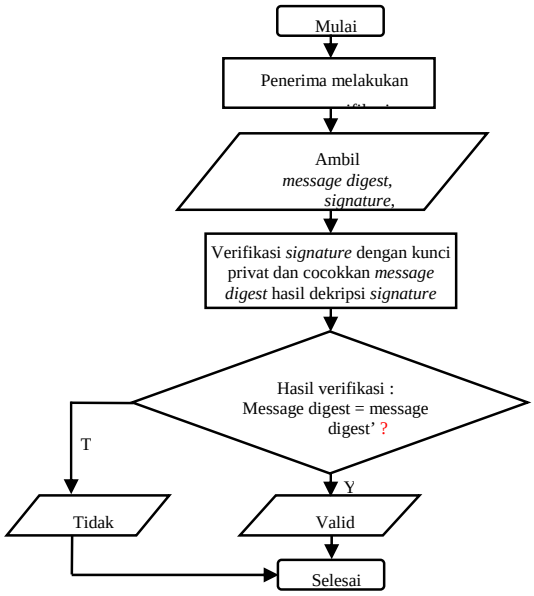
Gambar 2. Flow Chart Proses Pembuatan Kunci



Gambar 3. Flow Chart Proses Surat Masuk



Gambar 4. Flow Chart Proses Surat Keluar



Gambar 5. Flow Chart Proses Verifikasi

4. Hasil dan Pembahasan

4.1. Hasil Penelitian

Hasil penelitian ini adalah perangkat lunak administrasi surat atau sistem *mailtracking* berbasis *multiuser* yang menerapkan *digital signature* untuk keamanan data dan otentikasi pengirim surat menggunakan algoritma RSA. Perangkat lunak ini terdiri dari beberapa proses yaitu proses pendataan data master yang dilakukan oleh *user* admin, proses pembuatan kunci yang dilakukan oleh *user* progdi dan *user* dosen, proses pendataan surat masuk yang dilakukan oleh *user* admin, proses disposisi surat yang dilakukan oleh *user* pimpinan, proses kirim surat yang dilakukan oleh *user* admin, proses terima surat, dan proses verifikasi surat yang dilakukan oleh *user* progdi dan *user* dosen.

4.2. Implementasi

Lingkungan implementasi perangkat lunak ini membutuhkan beberapa aplikasi yang dibutuhkan diantaranya adalah pada *server* membutuhkan webserver, MySQL sebagai DBMS, dan phpMyAdmin sebagai *database manager*, sedangkan pada *client* membutuhkan aplikasi *web browser* seperti internet explorer, mozila, google chrome dan lain-lain. Pada simulasi sistem *mailtracking* pada penelitian ini menggunakan server localhost yaitu appserv version 2.6.0 dimana didalamnya aplikasi tersebut sudah ada aplikasi webserver, MySQL, dan phpMyAdmin.

Pembahasan penelitian ini dilakukan dengan melakukan simulasi terhadap administrasi surat atau sistem *mailtracking* yang akan diberi sistem keamanan menggunakan *digital signature*.

Simulasi dilakukan dari proses pendataan surat masuk, proses disposisi, proses pembuatan kunci publik dan kunci privat, proses kirim surat yang dikasih *digital signature*, dan proses verifikasi yang dilakukan oleh penerima surat.

Simulasi pada proses kirim surat akan dilakukan terhadap satu pengirim atau n pengirim ke satu penerima atau n penerima, dimana n=1,2,3,…. Pada proses ini akan terbentuk *message digest*, dan *digital signature* yang melekat pada data surat. Sedangkan simulasi pada proses verifikasi dilakukan dengan kunci dan atau *digital signature* yang sah atau sebenarnya juga dilakukan dengan kunci dan atau *digital signature* yang rusak atau tidak sebenarnya. Pada pembahasan penelitian ini juga dilakukan analisa waktu eksekusi pembuatan kunci, waktu enkripsi, dan waktu dekripsi pada saat proses verifikasi.

4.3. Pembahasan

Aplikasi sistem *mailtracking* dengan *digital signature* menggunakan algoritma RSA ini pada dasarnya memiliki empat buah proses yaitu pembangkitan kunci, proses pengiriman surat, proses enkripsi dan proses dekripsi. Di dalam program ini terdapat enam *function* yang secara esensial mengaplikasikan algoritma RSA. Masing-masing *function* tersebut akan dijelaskan sebagai berikut.

A. *Function* Bilangan Random

Function bilangan random/gmp_random ini digunakan membuat kunci yang lebih panjang. *Function* ini digunakan untuk mencari dua bilangan random dari 0 sampai 1 x limb, dan bilangan random ini akan digunakan dalam proses pencarian bilangan prima.

B. *Function* Bilangan Prima

Function bilangan prima/gmp_nextprime ini digunakan untuk memilih dua buah bilangan prima secara acak dari hasil pencarian bilangan random yang telah dilakukan oleh *function* bilangan random. *Function* ini memiliki dua buah variabel yaitu rand1 dan rand2 yang digunakan untuk menampung indeks masing-masing bilangan acak yang terpilih. Variabel rand1 memiliki bilangan acak yang nilainya ditentukan oleh hasil perkalian fungsi Math.random dengan nilai maksimal dari bilangan yang telah dibangkitkan pada *function* bilangan random dan hasil perkalian tersebut dibulatkan ke atas dengan menggunakan fungsi gmp_strval. Pencarian nilai rand2 sama caranya dengan variabel rand1.

C. *Function Totien*t

Function totient/phi ini digunakan untuk menghitung totient/phi. Fungsi yang digunakan adalah gmp_mul untuk proses perkalian dan gmp_sub untuk proses pengurangan.

D. *Function* Kunci Publik

Function ini untuk mencari kunci publik atau e, dimana e merupakan coprime dari totient. Dikatakan coprime dari totient jika gcd/fpb dari e dan totient = 1 maka di dalam *function* ini dibuat perulangan yang di dalamnya terdapat proses penelusuran nilai variabel gcd dimana nilai totient dan e digunakan sebagai parameter kunci publik. Perulangan ini membutuhkan *function* totient dalam pemfaktoran dari bilangan prima. Dimana perulangan akan berakhir ketika nilai dari variabel gcd sudah sama dengan 1 setelah itu *function* akan mengembalikan nilai variabel totient.

E. *Function* Kunci Privat

Function ini digunakan untuk mencari kunci privat yang akan digunakan dalam proses dekripsi plainteks. Di dalam *function* ini dibuat perulangan yang memproses penelusuran nilai variabel x sampai didapatkan nilai hasil bagi x dengan e berupa bilangan bulat dan didapat nilai d sama dengan hasil bagi tersebut dimana nilai x dalam perulangan didapat dari hasil tambah 1 dengan hasil kali totient dengan nilai iterasi yang sekarang.

F. *Function* Pow dan Mod

Function ini digunakan dalam proses enkripsi dan mengacu pada rumus enkripsi yaitu $C = M^e \bmod n$. *Function* ini memiliki tiga buah variabel sebagai parameter, diantaranya strlen untuk menampung nilai ASCII dari tiap karakter dalam plainteks yang akan dienkripsi, pow untuk menyimpan hasil eksponensial dan mod untuk menyimpan hasil modulasi. Untuk dapat memperoleh nilai yang besar dari hasil enkripsi tanpa kehilangan presisinya maka digunakanlah *class* BigInteger untuk menampung bilangan integer yang besar.

5. Kesimpulan

Aplikasi Sistem *mailtracking* dengan *digital signature* dapat digunakan untuk *routing* surat dan monitoring surat sehingga dapat tercipta legalitas dan efisiensi administrasi surat-menyurat serta memudahkan dalam memanajemen persuratan secara teratur. Sistem ini dapat mendukung proses pembuatan kunci secara dinamis, dari satu *user* atau pengguna dapat membangkitkan kunci lebih dari satu kali. Hal ini menunjukkan konsep baru penggunaan kunci untuk satu kali pakai. Selain itu sistem ini juga dapat mendukung proses pengiriman surat sebanyak n penerima, dengan melakukan proses penandatangan surat dilakukan dengan menggunakan kunci publik sebanyak n penerima. Implementasi *digital signature* untuk sistem keamanan *mailtracking* pada penelitian ini digunakan untuk otentikasi data pengirim atau penandatangan surat, dan didasarkan atas satu surat jika ditandatangani oleh n pengirim maka harus dilakukan verifikasi

sebanyak n kali. Jika semua n verifikasi bernilai valid berarti telah menembus n lapis keamanan dalam hal verifikasi. Sebaliknya jika salah satu atau lebih hasil verifikasi tidak valid dan atau gagal maka penerima dapat mengetahui jika surat yang diterima sudah tidak otentik dan atau salah satu atau lebih dari penandatangan surat adalah bukan orang yang sebenarnya menandatangani surat tersebut.

Untuk lebih mempercepat proses dekripsi RSA pada sistem *mailtracking* dikembangkan menggunakan salah satu metoda yang aman untuk mempercepat proses dekripsi RSA yaitu menggunakan *Chinese Remainder Theorem* (CRT). Sedangkan analisis untuk operasi eksponensialnya menggunakan metoda *Montgomery*. Pada sistem *mailtracking* dapat juga dikembangkan untuk fasilitas mengirim surat melalui email, menggunakan *function mail* yang ada di php dengan ditambah *module sendmail* supaya *function mail* tersebut berjalan dengan baik.Pada keamanan data file surat yang dikirim akan lebih aman jika ada proses hash dan enkripsi nama file, sehingga file surat yang diterima bisa dipertanggung jawabkan keasliannya.

Daftar Pustaka

[1] Jaafar, A.M dkk. 2010. *Visual digital Signature Scheme : A New Approach. IAENG International Journal of Computer Science. Volume 37 Issue 4*.

[2] Ariyus, D. 2008. Pengantar Ilmu Kriptografi, Yogyakarta: ANDI.

[3] Junaedi, D dkk. 2010. Prototype Aplikasi Pengolahan Surat Perintah Tugas Interen Berbasis Web di PT. PLN (Persero) Penyaluran dan Pusat Pengatur Beban Jawa Bali Region Jawa Barat Unit Pelayanan Transmisi Cirebon. Seminar Nasional Informatika (semnasIF), ISSN : 1979-2328.

[4] Delfs, Hans dkk. 2007. *Introduction to Cryptography, Principles and Applications Second Edition.* Switzerland : EPFL.

[5] Fatansyah. 1999. *Basis Data*. Informatika : Bandung.

[6] FIPS PUB 183-3 (2008). *Secure Hash Standard*, *Federal Information Processing Standards Publication* 180, U. S. Dept. of Commerce / National Institute of Standards and Technology.

[7] Shen, G dkk. 2009. *Research on Fast Implementation of* RSA *With* JAVA. *International Symposium on Web Information Systems and applications* (WISA), ISBN 978-952-5726-00-8 (Print), 978-952-5726-01-5 (CD-ROM), pp. 186-189.

[8] Jogiyanto, H.M. 1997. Sistem Informasi Berbasis Komputer Edisi ke-2. BPFE Yogyakarta : Yogyakarta.

[9] Kendall, K. E. dan Julie E.K. 2003. *Analisis dan Perancangan Sistem. Edisi Terjemahan*. PT Intan Sejati : Klaten.

[10] Kurniawan Yusuf. 2004. Kriptografi Keamanan Internet dan Jaringan Komunikasi. Bandung : Informatika.

[11] Menezes, Oorschot, Vanstone. 1996. *Handbook of Applied Cryptography*, CRC Press.

[12] Munir, Rinaldi. 2006. Kryptografi, Bandung: Informatika.

[13] M. Sreerama Murty dkk. 2011. *Digital Signature and Watermark Methods for Image Authentication using Criptography Analysis. Signal and Image Processing : An International Journal (SIPIJ). Volume 2 Issue 2 Page 170-179*.

[14] Maharani, S dkk. 2009. Implementasi Perangkat Lunak Penyandian Pesan Menggunakan Algoritma RSA. Jurnal Informatika Mulawarman, Vol. 4 No. 1.

[15] Stallings, William. 2005. *Cryptography and Network Security Principles and Practise*. New Jersey: Prentice Hall.

[16] Rizvi, S.S. dkk. 2010. *Combining Private and Public Key Encryption Techniques for Providing Extreme Secure Environment for an Academic Institution Application*. *International Journal of Network Security & Its Application* (IJNSA), Vol. 2 No. 1.

[17] PHPClasses. 2011. PHP *Classes Repository*.[online]. Tersedia : http://www.phpclasses.org/package/4121-PHP-Encrypt-and-decrypt-data-with-RSA-public-keys.html [5 Oktober 2011].

[18] Pressman, Roger S. 2002. Rekayasa Perangkat Lunak Pendekatan Praktisi (Edisi Bahasa Indonesia) Buku Dua. Yogyakarta: ANDI.

[19] Prasetyo, D.D. 2003. *Belajar Sendiri Administrasi Database Server. MySQL*. Elex Media Komputindo : Jakarta.

[20] *Erlina Cahya Setianingrum, Bambang Eka Purnama,* Sistem Pengaman Brankas Dengan Menggunakan Handphone Berbasis Mikrokontroler AT89S51, Seruni 2013 - Seminar Riset Unggulan Nasional Informatika dan Komputer

[21] *Slamet Riyadi, Bambang Eka Purnama,* Sistem Pengendalian Keamanan Pintu Rumah Berbasis SMS (Short Message Service) Menggunakan Mikrokontroler Atmega 8535, Vol 2, No 4 (2013): IJNS Oktober 2013