Tanda-Tangan Digital, Antara Ide dan Implementasi

Donny Kurnia, Agus Hilman Majid, dan Satria Buana

Departemen Teknik Informatika
Institut Teknologi Bandung
Jalan Ganesha 10 Bandung 40132

E-mail : if10021@students.if.itb.ac.id, if10064@students.if.itb.ac.id, if10087@students.if.itb.ac.id

Abstrak

Tanda-tangan telah lama digunakan untuk otentikasi dokumen kertas. Berkembangnya teknologi informasi, memunculkan ide untuk mengadaptasi tanda-tangan untuk dokumen-dokumen digital, yaitu tanda-tangan digital. Salah satu implementasinya yaitu dengan fungsi *hash* dan algoritma kunci-publik, dimana hanya pengguna yang memiliki kunci privat yang dapat menandatangani dokumen. Berbeda dengan tanda-tangan pada dokumen kertas yang selalu sama untuk semua dokumen, tanda-tangan digital berbeda-beda antara satu dokumen dengan dokumen lainnya, bergantung pada isi dokumen yang akan ditandatangani. Ide tanda-tangan digital ini sebenarnya cukup baik, apalagi secara matematis kekuatan algoritma kriptografi yang digunakan tidak diragukan. Tetapi sayangnya ada satu hal yang terlupa, bahwa pada dunia komputasi manipulasi terhadap program dan sabotase terhadap komputer pengguna bukanlah hal yang sulit dilakukan. Seseorang bisa saja menyabotase komputer orang lain untuk menandatangani dokumen tanpa sepengetahuan orang yang bersangkutan. Dengan kata lain, tanda-tangan digital hanya memberikan otentikasi antara dokumen dengan komputer, tetapi tidak memberikan otentikasi keterkaitan antara komputer dengan pemilik kunci privat yang sah. Jika seseorang berada di pengadilan dan ditanya tentang tanda-tangan digital miliknya pada sebuah dokumen, dia dapat saja mengatakan bahwa ia tidak pernah menandatangani dokumen tersebut, dan ketika saksi ahli dihadirkan ia akan menjelaskan bahwa mungkin saja dokumen diberi tanda-tangan digital tanpa sepengetahuan si pemilik kunci privat.

***Kata kunci**:tanda-tangan digital, fungsi hash, algoritma kunci-publik*

1. Pendahuluan

Tanda-tangan telah lama digunakan untuk membuktikan otentikasi dokumen kertas (misalnya surat piagam, ijazah, buku, karya seni, dan sebagainya). Seseorang yang tanda-tangannya tertera pada suatu dokumen kertas tidak akan dapat menyangkal bahwa bukan ia yang menandatangani dokumen tersebut. Jika demikian, maka dapat didatangkan 'ahli tanda-tangan' untuk memastikan keaslian tanda-tangan tersebut.

Dengan berkembangnya teknologi, maka bentuk dokumen tidak lagi selalu 'nyata' seperti halnya dokumen kertas, tetapi muncul juga dokumen dalam bentuk digital. Terinspirasi dengan kegunaan tanda-tangan, maka para ahli komputer memikirkan cara untuk menerapkan tanda-tangan pada dokumen digital. Tanda-tangan digital memiliki karakteristik yang sedikit berbeda dengan tanda-tangan pada dokumen kertas. Jika tanda-tangan pada dokumen kertas selalu sama untuk semua dokumen, maka tidak demikian halnya dengan tanda-tangan digital. Tanda-tangan digital akan berbeda-

beda dari satu dokumen dengan dokumen lainnya, ini karena tanda-tangan digital adalah suatu nilai kriptografis yang nilainya bergantung pada dokumen yang akan ditandatangani.

2. Fungsi *Hash* Satu Arah dan Algoritma Kunci-Publik

Fungsi *hash* adalah fungsi yang menerima masukan *string* dengan panjang sembarang dan mengkonversinya menjadi *string* keluaran dengan panjang tetap.

Algoritma kunci-publik adalah algoritma kriptografi asimetri dimana kunci yang digunakan untuk mengenkripsi pesan berbeda dengan kunci yang digunakan untuk mendekripsi pesan. Ada dua jenis kunci pada algoritma kunci-publik, yaitu kunci privat dan kunci publik. Setiap orang yang ingin menggunakan algoritma kunci-publik harus memiliki sepasang kunci tersebut. Kunci publik tidak bersifat rahasia sehingga semua orang boleh mengetahuinya, sementara kunci privat bersifat rahasia dan tidak boleh diketahui oleh orang lain.

Umumnya kunci publik digunakan untuk mengenkripsi pesan, sementara kunci privat untuk mendekripsi pesan. Seseorang (katakanlah A) yang ingin mengirim pesan ke pihak lain (katakanlah B), akan mengenkripsi pesan yang ingin dikirimkannya dengan menggunakan kunci publik si B. Pesan terenkripsi ini mungkin saja ‘disadap’ oleh pihak lain yang tidak berhak, tetapi karena hanya si B yang mengetahui kunci privat-nya, maka hanya si B yang dapat mengetahui isi pesannya.

Berbeda dengan mekanisme pada pengiriman pesan biasa, penggunaan algoritma kunci-publik pada tanda-tangan digital justru menggunakan kunci privat untuk proses enkripsi dan kunci publik untuk proses dekripsi.

Cara yang umum digunakan untuk ‘membangkitkan’ tanda-tangan digital yaitu dengan kombinasi fungsi *hash* satu arah dan algoritma kunci-publik. Fungsi *hash* akan menghasilkan pesan-ringkas (*message digest*) dari dokumen, kemudian pesan-ringkas ini akan dienkripsi dengan menggunakan kunci privat orang yang akan menandatangani dokumen tersebut. Hasil enkripsi pesan-ringkas inilah yang disebut sebagai tanda-tangan digital. Tanda-tangan digital akan disertakan (*embedded*) pada dokumen digital terkait.

Verifikasi terhadap dokumen digital dilakukan dengan menggunakan kunci-publik orang yang menandatangani. Mula-mula dokumen dipisahkan menjadi dua bagian, yaitu bagian isi dan tanda-tangan digital. Tanda-tangan digital kemudian didekripsi dengan menggunakan kunci publik si ‘pemilik’ dokumen untuk mendapat nilai *hash*-nya. Nilai *hash* ini kemudian dibandingkan dengan nilai *hash* dokumen (bagian isinya). Jika sama, berarti benar dokumen tersebut telah ditandatangani secara digital oleh orang yang memiliki kunci privat terkait.

3. Keabsahan Tanda-tangan Digital

Sejak ditemukan pada tahun 1970-an, tanda-tangan digital diharapkan dapat berperan sebagaimana tanda-tangan pada dokumen kertas. Perkembangan teknologi yang pesat menjadikan tanda-tangan digital sebagai komponen penting dalam bisnis di *cyberspace* saat ini. Bahkan di beberapa negara maju, tanda-tangan digital telah memiliki kekuatan hukum.

Di saat tanda-tangan digital mulai lazim digunakan dan bahkan berkekuatan hukum, sebagian ahli di bidang informatika justru mulai mempertanyakan keabsahan tanda-tangan digital. Bukan karena lemahnya kekuatan matematis dari algoritma yang digunakan, tapi lebih kepada banyaknya celah-celah kelemahan pada saat implementasi tanda-tangan digital.

Misalkan, Alice (nama ini akan digunakan seterusnya untuk menyederhanakan penjelasan) memiliki kunci privat dimana hanya ia yang mengetahuinya. Saat ia ingin ‘menandatangani’ dokumen, ia akan menghitung nilai *hash* dari dokumen tersebut. Kemudian ia akan melakukan kalkulasi matematis terhadap nilai *hash* tersebut dengan menggunakan kunci privat-nya (mengenkripsi). Hasil kalkulasi inilah yang kemudian disebut sebagai tanda-tangan digital. Semua orang dapat melakukan verifikasi terhadap tanda-tangan digital tersebut dengan menggunakan kunci publik Alice (mekanisme verifikasi telah dijelaskan pada bagian sebelumnya). Jika verifikasi berhasil, berarti benar bahwa Alice yang telah menandatangani dokumen tersebut, karena hanya ia yang mengetahui kunci privat-nya.

Secara matematis, mekanisme diatas bekerja dengan baik. Tetapi tidak secara semantik. Contoh diatas tidak menjelaskan apapun tentang ‘menandatangani’. Kenyataannya, ‘tanda-tangan digital’ mungkin adalah kesalahan tata-nama terburuk dalam sejarah kriptografi.

Dalam hukum, tanda-tangan digunakan untuk mengindikasikan persetujuan, atau setidaknya pengakuan terhadap dokumen yang ditandatangani. Ketika dipersidangan misalnya, hakim/juri melihat dokumen kertas yang ditandatangani Alice, maka ia mengetahui bahwa Alice pernah ‘memegang’ dokumen itu sebelumnya, dan memiliki alasan untuk percaya bahwa Alice membaca dan menyetujui kata-kata pada dokumen tersebut. Seseorang tentu tidak dapat begitu saja membuat dokumen dengan tanda-tangan palsu dan mengatakan bahwa dokumen tersebut telah ditandatangani oleh Alice. Jika demikian, hal tersebut justru dapat membahayakan dirinya sendiri karena bisa saja ‘ahli tanda-tangan’ dihadirkan sebagai saksi. Untuk menghindari hal ini, maka untuk dokumen-dokumen yang sifatnya penting, digunakan tanda-tangan notaris.

Lalu bagaimana halnya dengan tanda-tangan digital ? Bagaimana membuktikan kepada hakim/juri bahwa benar Alice yang menandatangi dokumen tersebut ? Atau bahkan membuktikan bahwa Alice pernah melihat dokumen tersebut. Apakah verifikasi digital dengan menggunakan kunci publik Alice cukup ? Secara konsep seharusnya iya. Jika hasil verifikasi sesuai, berarti benar bahwa Alice yang telah menandatangani dokumen tersebut. Karena hanya dia yang ‘dianggap’ mengetahui kunci privat yang bersesuaian. Tetapi pada implementansinya, ternyata terdapat hal-hal yang menyebabkan

keabsahan tanda-tangan digital menjadi cacat.

Masalahnya adalah, tanda-tangan digital hanya memberikan otentikasi antara dokumen dengan komputer yang menandatangani dokumen tersebut, tetapi tidak memberikan otentikasi keterkaitan antara Alice dengan komputer. Perlu dicatat bahwa bukan Alice yang menghitung nilai tanda-tangan digital dari dokumen, melainkan komputer yang melakukannya untuk Alice. Hal ini berbeda dengan tanda-tangan pada dokumen kertas dimana Alice sendiri yang langsung menandatangani dokumen tersebut.

Sebagai contoh misalnya PGP (*Pretty Good Privacy*). Program ini memberikan tanda-tangan digital pada *e-mail*. Antarmuka penggunanya sederhana, saat seseorang ingin memberikan tanda-tangan digital pada *e-mail*, dia dapat memilih menu yang sesuai, memasukan *passphrase* pada kotak dialog, dan klik "OK". Program akan melakukan dekripsi dengan *passphrase* tadi untuk mendapatkan kunci privat. Dengan kunci privat ini kemudian dilakukan kalkulasi nilai kriptografi dari pesan tersebut, hasil kalkulasi (enkripsi) inilah yang kemudian disebut sebagai tanda-tangan digital dan kemudian disertakan (*embedded*) pada *e-mail*. Suka atau tidak, pengguna hanya dapat percaya bahwa PGP melakukan kalkulasi tanda-tangan digital dengan valid, bahwa PGP menandatangani *e-mail* sebagaimana yang diinginkannya, bahwa PGP tidak memberikan salinan (*copy*) kunci privat-nya ke pihak lain, yang kemudian dapat menandatangani apa saja yang ia inginkan dengan menggunakan tanda-tangannya. Seseorang dapat dengan mudah menulis versi 'nakal' dari PGP yang kemudian menggunakan tanda-tangan digital pengguna untuk menandatangani pesan lainnya. Seseorang juga dapat menulis *back orifice plug-in* yang meng-*capture* kunci privat pengguna dan menandatangani dokumen lain tanpa ijin atau sepengetahuan pemiliknya. Kita bahkan telah mengenal virus komputer yang mencoba mencuri kunci privat PGP, yaitu varian dari *nastier*. Intinya, banyak cara yang dapat digunakan untuk menandatangani dokumen dengan tanda-tangan digital milik orang lain.

Ini menunjukan, betapa pun kuatnya nilai matematis kriptografi, tetap tidak dapat menjembatani jurang pemisah antara pengguna dengan komputer. Jadi tanda-tangan digital hanya memberikan otentikasi bahwa dokumen telah ditandatangani secara digital oleh komputer, tetapi tidak menjamin bahwa dokumen tersebut benar-benar ditandatangani oleh orang yang berhak. Hal ini dikarenakan pada dunia komputasi sebuah program mudah sekali dimanipulasi.

Bayangkan situasi pengadilan dimana Alice ditanya perihal dokumen yang ditandatanganinya secara digital. Alice kemudian menjawab bahwa ia bahkan tidak pernah melihat dokumen tersebut. Dia mengakui bahwa secara matematis memang terbukti bahwa kunci privatnya yang menandatangani dokumen tersebut, tetapi dia bersikeras bukan dia yang menandatangani dokumen tersebut. Bahkan, Alice mengaku tidak pernah melihatnya. Saat seorang pakar dihadirkan pada sidang tersebut, ia akan menjelaskan bahwa mungkin saja Alice tidak pernah melihat dokumen tersebut, bahwa dapat ditulis sebuah program untuk menandatangani dokumen tanpa sepengetahuan Alice, dan bahwa apa yang disebut dengan 'tanda-tangan digital' sama sekali tidak membuktikan bahwa Alice yang menandatangani dokumen tersebut. Lalu

dimana kekuatan ‘tanda-tangan digital’. Jika demikian, apakah fungsi tanda-tangan digital untuk otentikasi pesan, otentikasi pengirim, dan anti-penyangkalan masih valid ?

Konsep awal dari tanda-tangan digital ini memang sangat baik. Tapi ada satu hal yang terlupa, bahwa *gap* antara pengguna dan komputer cukup besar. Artinya, kita tidak benar-benar bisa memastikan apakah suatu komputer benar-benar digunakan oleh Alice untuk memberikan tanda-tangan digital, atau sebenarnya ada orang lain yang dengan kemampuannya mampu menyabotase komputer tersebut sehingga seolah-olah Alice yang memberi tanda-tangan digital. Berbeda dengan tanda-tangan pada dokumen kertas, sehebat-hebatnya seseorang menirukan tanda-tangan Alice, tetap tidak akan pernah persis sama karena setiap orang mempunyai pola ‘goresan tulisan’ yang unik. Hal ini dapat dibuktikan dengan menghadirkan ‘ahli tanda-tangan’.

Tanda-tangan digital membuktikan secara matematis bahwa nilai rahasia yang dikenal sebagai kunci privat memang dimasukkan ke komputer. Tetapi sama sekali tidak membuktikan bahwa kunci privat tersebut dimasukan oleh orang yang berhak.

4. Kesimpulan

Keabsahan tanda-tangan digital tidak dapat disamakan dengan keabsahan tanda-tangan pada dokumen kertas. Tanda-tangan pada dokumen kertas menunjukkan otentikasi dokumen dan tidak dapat disangkal. Tidak ada satu orang pun yang dapat menirukan secara persis tanda-tangan orang lain. Tanda-tangan pada dokumen kertas mengindikasikan bahwa pemilik tanda-tangan menyetujui atau mengetahui kata-kata yang tertera pada dokumen tersebut.

Tanda-tangan digital keabsahannya dapat dipertanyakan, bukan karena lemahnya kekuatan matematis algoritma kriptografi yang digunakan, tetapi karena celah-celah keamanan saat seseorang menggunakan komputer. Banyak cara yang dapat dilakukan untuk memberi tanda-tangan digital tanpa sepengetahuan orang yang berhak (yang memiliki kunci privat). Ini disebabkan pada dunia maya sabotase terhadap komputer orang lain bukan lah hal yang sulit dilakukan. Tanda-tangan digital hanya memberikan otentikasi antara dokumen dengan komputer, tetapi tidak memberikan otentikasi keterkaitan antara pemilik kunci privat yang sah dengan komputer.

[1] Ir. Rinaldi Munir, M.T., *Digital Signature Standard (DSS)*, Informatika ITB, 2004.
[2] Bruce Schneier, *Why Digital Signatures Are Not Signatures*, http://www.schneier.com/crypto-gram-0011.html, diakses tanggal 31 Desember 2004 pukul 15:20
[3] Bruce Schneier, *Security Pitfalls in Cryptography*, http://www.schneier.com/essay-028.html, diakses tanggal 31 Desember 2004 pukul 15:20