Bahan Kuliah ke-18

IF5054 Kriptografi

Otentikasi dan Tandatangan Digital
(*Authentication and Digital Signature*)

Disusun oleh:

Ir. Rinaldi Munir, M.T.

Departemen Teknik Informatika
Institut Teknologi Bandung
2004

18. Otentikasi dan Tandatangan Digital
(*Authentication and Digital Signature*)

18.1 Pendahuluan

- Untuk mengingatkan kembali mengenai kegunaan kriptografi, di bawah ini kita tuliskan kembali masalah-masalah keamana yang dapat diselesaikan dengan kriptografi:

  1. Kerahasiaan pesan (*confidentiality/secrecy*)
     Kriptografi menjaga kerahasiaan pesan dengan cara mengenkripsinya ke dalam bentuk yang tidak mempunyai makna.

  2. Keabsahan pengirim (*user authentication*).
     Hal ini berkaitan dengan kebenaran identitas pengirim. Dengan kata lain, masalah ini dapat diungkapkan sebagai pertanyaan: "Apakah pesan yang diterima benar-benar berasal dari pengirim yang sesungguhnya?"

  3. Keaslian pesan (*message integrity*).
     Hal ini berkaitan dengan keutuhan (*integrity*) pesan. Dengan kata lain, masalah ini dapat diungkapkan sebagai pertanyaan: "Apakah pesan yang diterima tidak mengalami perubahan (modifikasi)?"

  4. Anti-penyangkalan (*nonrepudiation*).
     Pengirim tidak dapat menyangkal (berbohong) tentang isi pesan yang ia kirim.

- Tiga masalah yang terakhir dapat diselesaikan dengan teknik otentikasi pesan (*message authentication*) atau disingkat otentikasi saja.

- Teknik otentikasi (dalam komunikasi data) adalah prosedur yang digunakan untuk membuktikan:
  1. Keaslian pesan (*message integrity*)
  2. Keaslian identitas pengirim (*user authentication*).
  3. Pengirim tidak dapat menyangkal isi pesan (*non-repudiation*)

- Dua alternatif cara yang digunakan untuk otentikasi:
  1. Menandatangani pesan (*message signature*)
     Pesan ditandangani oleh pengirim. Pemberian tanda tangan adalah secara digital. Pesan yang sudah ditandatangani menunjukkan bahwa pesan tersebut otentik (baik otentik isi maupun otentik pengirim).

  2. Menggunakan *MAC* (*Message Authentication Code*).
     MAC adalah kode yang ditambahkan (*append*) pada pesan. Kode tersebut dibangkitkan oleh suatu algoritma, dan bergantung pada pesan dan kunci rahasia.

17.2 Menandatangani Pesan

Tanda-tangan Digital (Digital Signature)

- Sejak berabad-abad lamanya, tanda tangan digunakan untuk membuktikan otentikasi dokumen kertas (misalnya surat, piagam, ijazah, buku, karya seni, dan sebagainya).

- Tanda-tangan mempunyai karakteristik sebagai berikut:
  1. Tanda-tangan adalah bukti yang otentik.
  2. Tanda tangan tidak dapat dilupakan.
  3. Tanda-tangan tidak dapat dipindah untuk digunakan ulang.
  4. Dokumen yang telah ditandatangani tidak dapat diubah.
  5. Tanda-tangan tidak dapat disangkal (*repudiation*).

- Fungsi tanda tangan pada dokumen kertas juga diterapkan untuk otentikasi pada data digital seperti pesan yang dikirim melalui saluran komunikasi dan dokumen elektronis yang disimpan di dalam memori komputer.

- Tanda tangan pada data digital ini disebut **tanda-tangan digital** (*digital signature*). Yang dimaksud dengan tanda-tangan digital bukanlah tanda tangan yang di-digitisasi dengan alat *scanner*, tetapi suatu nilai kriptografis yang bergantung pada pesan dan pengirim pesan (Hal ini kontras dengan tanda tangan pada dokumen kertas yang bergantung hanya pada pengirim dan selalu sama untuk semua dokumen).

- Dengan tanda-tangan digital, maka integritas data dapat dijamin, disamping itu ia juga digunakan untuk membuktikan asal pesan (keabsahan pengirim), dan anti-penyangkalan.

- Menandatangani pesan dapat dilakukan dengan salah satu dari dua cara:

  1. Enkripsi pesan
     Mengenkripsi pesan dengan sendirinya juga menyediakan ukuran otentikasi. Pesan yang terenkripsi sudah menyatakan bahwa pesan tersebut telah ditandatangani.

  2. Tanda tangan digital dengan fungsi *hash* (*hash function*)
     Tanda-tangan digital dibangkitkan dari *hash* terhadap pesan. Nilai *hash* adalah kode ringkas dari pesan. Tanda tangan digital berlaku seperti tanda-tangan pada dokumen kertas. Tanda-tangan digital ditambahkan (*append*) pada pesan.

1. Penandatangan dengan Cara Mengenkripsi Pesan

(a) Menandatangani Pesan dengan Algoritma Simetri

- Pesan yang dienkripsi dengan algoritma simetri sudah memberikan solusi untuk otentikasi pengirim dan keaslian pesan, karena kunci simetri hanya diketahui oleh pengirim dan penerima. Jadi, jika *B* menerima pesan dari *A*, maka ia percaya pesan itu dari *A* dan isinya tidak mengalami perubahan, karena tidak ada orang lain yang mengetahui kunci selain mereka berdua.

- Namun, algoritma simetri tidak dapat menyediakan suatu mekanisme untuk mengatasi masalah penyangkalan, yaitu jika salah satu dari dua pihak, *A* dan *B*, membantah isi pesan atau telah mengirim pesan.

- Agar dapat mengatasi masalah penyangkalan, maka diperlukan pihak ketiga yang dipercaya oleh pengirim/penerima. Pihak ketiga ini disebut penengah (arbitrase).

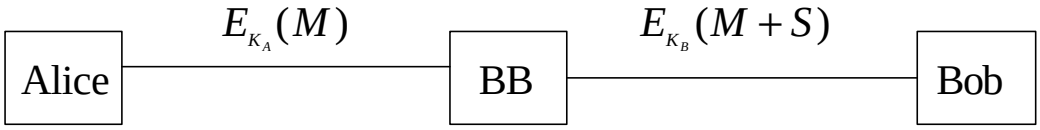
- Misalkan BB (*Big Brothers*) adalah otoritas arbitrase yang dipercaya oleh Alice dan Bob. BB memberikan kunci rahasia K_A kepada Alice dan kunci rahasia K_B kepada Bob.

  Hanya Alice dan BB yang mengetahui K_A, begitu juga hanya Bob dan BB yang mengetahui K_B.

- Jika Alice bekirim pesan *P* kepada Bob, maka langkah-langkahnya adalah sebagai berikut:
  1. Alice mengenkripsi pesan *M* untuk Bob dengan K_A, lalu mengirim cipherteksnya ke BB.

2. BB melihat bahwa pesan dari Alice, lalu mendekripsi pesan dari Alice dengan K_A.
3. BB membuat pernyataan *S* bahwa ia menerima pesan dari Alice, lalu menambahkan pernyataan tersebut pada plainteks dari Alice.
4. BB mengenkripsi bundel pesan (M + S) dengan K_B, lalu mengirimkannya kepada Bob.
5. Bob mendekripsi bundel pesan dengan K_B. Ia dapat membaca pesan dari Alice (M) dan pernyataan (S) dari BB bahwa Alice yang mengirim pesan tersebut.

Gambar 18.1 memperlihatkan skema penandatanganan yang dimaksudkan.



Gambar 17.1 Penandanganan pesan dengan bantuan arbitrase

- Jika Alice menyangkal telah mengirim pesan tersebut, maka pernyataan dari BB pada pesan yang diterima oleh Bob digunakan untuk menolak penyangkalan Alice.

- Bagaimana BB tahu bahwa pesan tersebut dari Alice dan bukan dari Charlie? Karena hanya BB dan Alice yang mengetahui kunci rahasia, maka hanya Alice yang dapat mengenkripsi pesan dengan kunci tesrebut.

(b) Menandatangani Pesan dengan Algoritma Kunci-Publik

- Jika algoritma kunci-publik digunakan, maka enkripsi pesan dengan kunci publik tidak dapat digunakan untuk otentikasi, karena setiap orang potensial mengetahui kunci-publik.

- Tetapi, jika enkripsi pesan menggunakan kunci privat si pengirim dan dekripsi pesan juga menggunakan kunci-publik si pengirim, maka kerahasiaan pesan (*secrecy*) dan otentikasi keduanya dicapai sekaligus. Ide ini ditemukan oleh Diffie dan Hellman.

- Beberapa agoritma kunci-publik seperti RSA dapat digunakan untuk menandatangani pesan dengan cara mengenkripsinya, asalkan algoritma tersebut memenuhi sifat: $D_{SeK}(E_{PK}(M)) = M$ dan $D_{PK}(E_{SK}(M)) = M$, dengan PK = kunci publik dan SK = kunci privat (*secret key*).

  Sebagai contoh, pada algoritma RSA, kunci publik atau kunci privat dapat digunakan untuk untuk enkripsi (lihat lagi penurunan fungsi enkripsi/dekripsi RSA).

- Misalkan M adalah pesan yang akan dikirim. Pesan M ditandatangani menjadi pesan terenkripsi S dengan menggunakan kunci privat (SK) si pengirim,

  $$S = E_{SK}(M) \qquad (18.1)$$

  yang dalam hal ini, E adalah fungsi enkripsi dari algoritma kunci-publik. Selanjutnya, S dikirim melalui saluran komunikasi.

- Di tempat penerima, pesan dibuktikan otentikasinya dengan menggunakan kunci publik (*PK*) pengirim,

$$M = D_{PK}(S) \qquad (18.2)$$

  yang dalam hal ini, *D* adalah fungsi enkripsi dari algoritma kunci-publik. *S* dikatakan absah apabila pesan *M* yang dihasilkan merupakan pesan yang mempunyai makna.

- Dengan algoritma kunci-publik, penandatanganan pesan tidak membutuhkan lagi pihak penengah (arbitrase).

2. Tanda-tangan dengan Menggunakan Fungsi Hash

- Penandanganan pesan dengan cara mengenkripsinya selalu memberikan dua fungsi berbeda: kerahasiaan pesan dan otentikasi pesan.

- Pada beberapa kasus, seringkali otentikasi yang diperlukan, tetapi kerahasiaan pesan tidak. Maksudnya, pesan tidak perlu dienkripsikan, sebab yang dibutuhkan hanya keotentikan pesan saja.

- Hanya sistem kriptografi kunci-publik yang cocok dan alami untuk pemberian tanda-tangan digital dengan menggunakan fungsi *hash*. Hal ini disebabkan karena skema tanda-tangan digital berbasis sistem kunci-publik dapat menyelesaikan masalah *non-repudiation* (baik penerima dan pengirim pesan mempunyai pasangan kunci masing-masing).

Proses Pemberian Tanda-tangan Digital (Signing)

- Pesan yang hendak dikirim diubah terlebih dahulu menjadi bentuk yang ringkas yang disebut *message digest. Message digest* (*MD*) diperoleh dengan mentransformasikan pesan *M* dengan menggunakan fungsi *hash* satu-arah (*one-way*) *H*,

$$MD = H(M) \qquad (18.3)$$

  Pesan yang sudah diubah menjadi *message digest* oleh fungsi *hash* tidak dapat dikembalikan lagi menjadi bentuk semula walaupun digunakan algoritma dan kunci yang sama (itulah sebabnya dinamakan fungsi *hash* satu-arah).

  Sembarang pesan yang berukuran apapun diubah oleh fungsi *hash* menjadi *message digest* yang berukuran tetap.

  Message digest disebut juga nilai *hash* (*hash value*) dari fungsi *hash*, *H*.

- Selanjutnya, *message digest MD* dienkripsikan dengan algoritma kunci-publik menggunakan kunci privat (SK) pengirim menjadi tanda-tangan digital *S*,

$$S = E_{SK}(MD) \qquad (18.4)$$

- Pesan *M* disambung (*append*) dengan tanda-tangan digital *S*, lalu keduanya dikirim melalui saluran komunikasi. Dalam hal ini, kita katakan bahwa pesan *M* sudah ditandatangani oleh pengirim dengan tanda-tangan digital *S*.

- Di tempat penerima, tanda-tangan diverifikasi untuk dibuktikan keotentikannya dengan cara berikut:

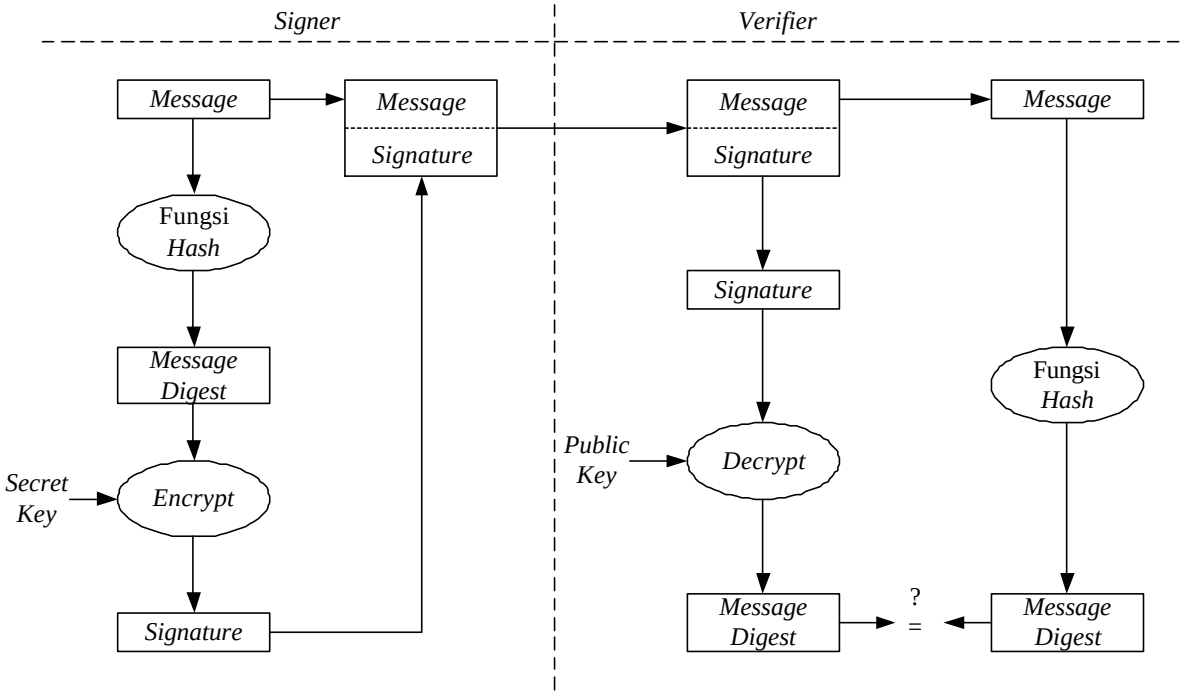
1. Tanda-tangan digital *S* didekripsi dengan menggunakan kunci publik (*PK*) pengirim pesan, menghasilkan *message digest* semula, *MD*, sebagai berikut:

$$MD = D_{PK}(S) \qquad (18.5)$$

2. Pengirim kemudian mengubah pesan *M* menjadi *message digest MD'* menggunakan fungsi *hash* satu-arah yang sama dengan fungsi *hash* yang digunakan oleh pengirim.

4. Jika *MD' = MD*, berarti tanda-tangan yang diterima otentik dan berasal dari pengirim yang benar.

Skema otentikasi dengan Tanda-tangan digital ditunjukkan pada Gambar 18.2.



Gambar 18.2. Otentikasi dengan tanda-tangan-digital yang menggunakan fungsi *hash* satu-arah

- Keotentikan ini dijelaskan sebagai berikut:
  a. Apabila pesan *M* yang diterima sudah berubah, maka *MD'* yang dihasilkan dari fungsi *hash* berbeda dengan *MD* semula. Ini berarti pesan tidak asli lagi.
  b. 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 privat pengirim).
  c. Bila *MD = MD'*, ini berarti pesan yang diterima adalah pesan yang asli (*message authentication*) dan orang yang mengirim adalah orang yang sebenarnya (*user authentication*).

- Dua algoritma *signature* yang digunakan secara luas adalah *RSA* dan *ElGamal*. Pada *RSA*, algoritma enkripsi dan dekripsi identik, sehingga proses *signature* dan verifikasi juga identik.

- Selain *RSA*, terdapat algoritma yang dikhususkan untuk tanda-tangan digital, yaitu *Digital Signature Algorithm* (DSA), yang merupakan bakuan (*standard*) untuk *Digital Dignature Standard* (DSS). Pada *DSA*, algoritma *signature* dan verifikasi berbeda