#26

Syahrul Imardi, MT

STMIK Amik Riau

MATAKULIAH

KEAMANAN PERANGKAT LUNAK

Digital Signatures

P26

STMIK Amik Riau

MATAKULIAH
KEAMANAN PERANGKAT LUNAK

Syahrul Imardi, MT

P26: **Digital Signature**

Tanda-tangan Digital

Review materi awal

- Aspek keamanan yang disediakan oleh kriptografi:
  1. Kerahasiaan pesan (*confidentiality/secrecy*)
  2. Otentikasi (*authentication*).
  3. Keaslian pesan (*data integrity*).
  4. Anti-penyangkalan (*nonrepudiation*).

- Aspek 1 diselesaikan dengan enkripsi/dekripsi
- Aspek 2 s/d 4 diselesaikan dengan tanda-tangan digital (*digital signature*).

Tanda-tangan

- Sejak zaman dahulu, tanda-tangan sudah digunakan untuk otentikasi dokumen cetak.

- 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.

Hari: Senin, Tanggal 12 September 2016

Waktu: 14.00 Wib

Tempat: Aula

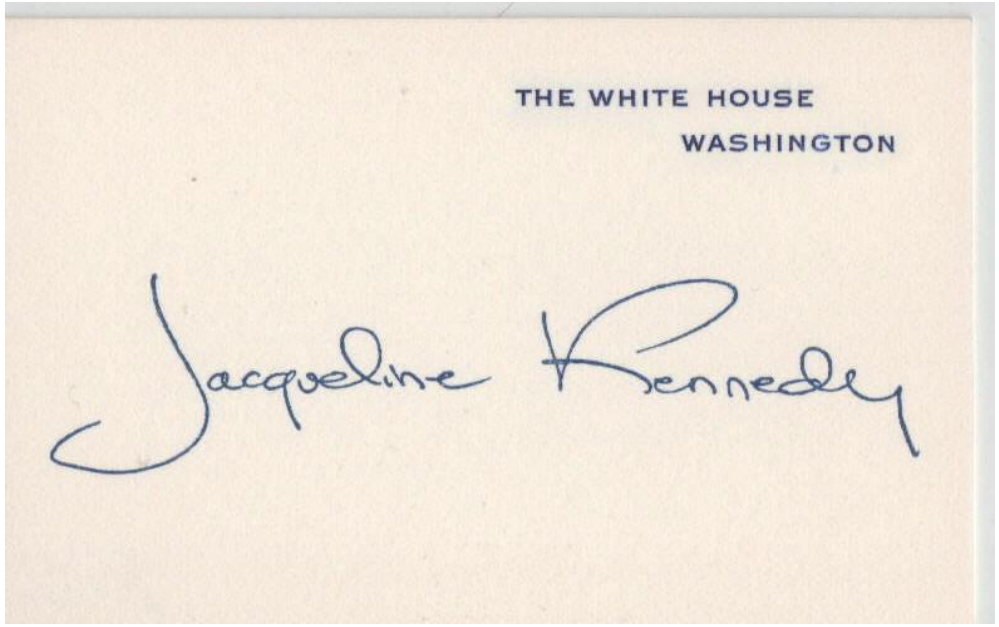
Demikian kami buat surat ini dengan sebenarnya. Harap semua peserta untuk hadir sesuai jadwal yang telah disebutkan di atas. Terima kasih.

Tertanda tangan

Mr Mukidi

Master of Ceremony
Doktor Mukidi, Mpd

- Fungsi tanda tangan pada dokumen kertas juga diterapkan untuk otentikasi pada data digital (pesan, dokumen elektronik).
- Tanda-tangan untuk data digital dinamakan **tanda-tangan digital** (*digital signature*).
- Tanda-tangan digital bukanlah tulisan tanda-tangan yang di-digitisasi (*digitized signature*) dengan cara dipindai atau difoto.



- Tanda-tangan digital adalah nilai kriptografis yang bergantung pada isi pesan dan kunci.

- Tanda-tangan pada dokumen cetak selalu sama, apa pun isi dokumennya.

- Tanda-tangan digital selalu berbeda-beda antara satu isi dokumen dengan dokumen lain.

Paris, 31 Desember 2018

Halo Alice
Sudah lama kita tidak berjumpa sejak lulus SMA. Saya sekarang tinggal di Paris sejak tahun 2016. Saya bekerja di sebuah perusahaan IT yang bernama Solution Express. Perusahaan ini memberikan layanan keamanan informasi berbasis cloud computing. Saya menjadi menejer Quality Control. Klien kami umumnya adalah bank-bank yang membutuhkan keamanan data nasabah.

Oh ya, saya belum menanyakan bagaimana keadaanmu sekarang. Di mana kamu bekerja atau malah melanjutkan studi S2 di mana? Saya ingat kamu dulu jago sekali pelajaran kimia. Apakah kamu masih menekuni bidang kimia saat ini?

Oke deh, jika kamu jalan-jalan ke Eropa jangan lupa mampir ke kota Paris. Nanti saya akan ajak kamu mengunjungi Menara Eiffel. Bisa naik sampai ke atas lho.

Salam dari temanmu di Paris
Bob

-- BEGIN SIGNATURE—
13706B6D42442620B2FD1098BD4D54ADFA9F7DC27576954ADCE5E5FC901
-- END SIGNATURE--

Dua cara menandatangani pesan:

1. Mengenkripsi pesan

2. Menggunakan kombinasi fungsi *hash* (*hash function*) dan kriptografi kunci-publik

Penandatangan dengan Cara Mengenkripsi Pesan

a. Menggunakan kriptografi simetri

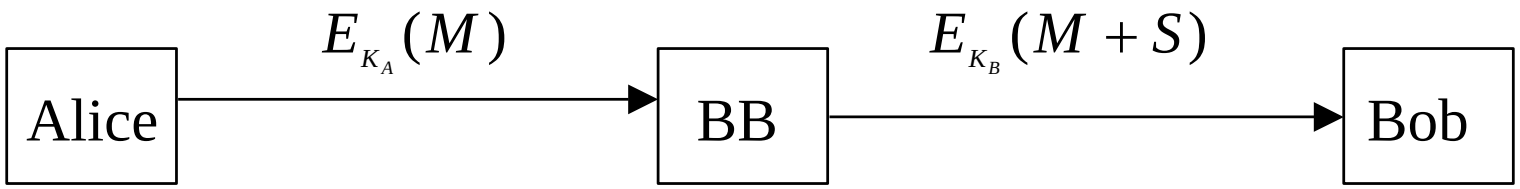
- Pesan yang dienkripsi dengan algoritma simetri sudah memberikan solusi untuk otentikasi pengirim karena kunci simetri hanya diketahui oleh pengirim dan penerima.

- Namun cara ini tidak menyediakan mekanisme untuk anti-penyangkalan.

- 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 M 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.



- 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 terebut.

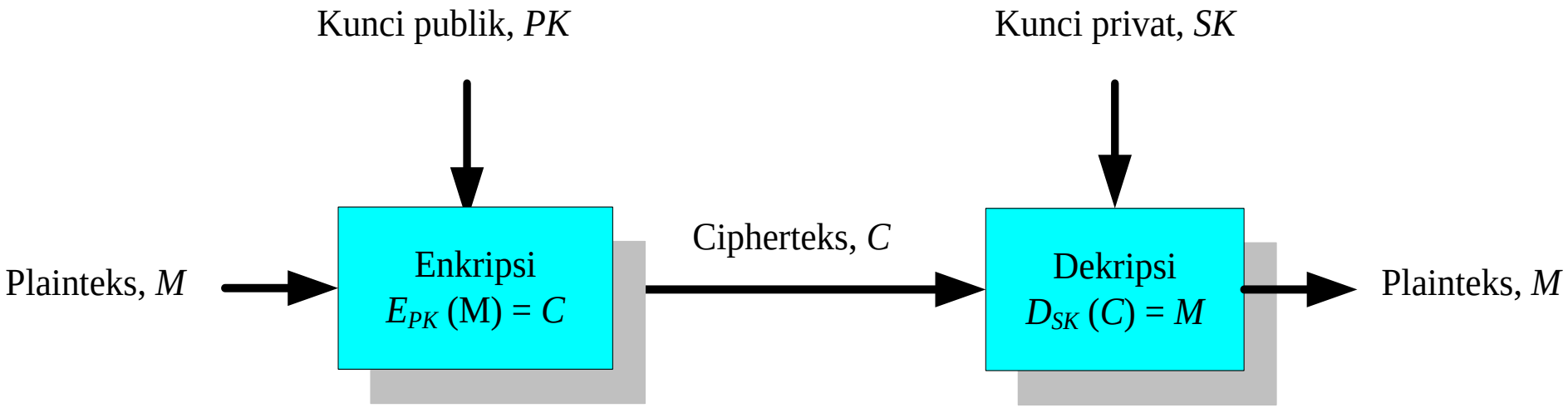
b. Mengunakan kriptografi kunci-publik

Enkripsi biasa dengan kriptografi kunci-public:

- pesan dienkripsi dengan kunci publik penerima.

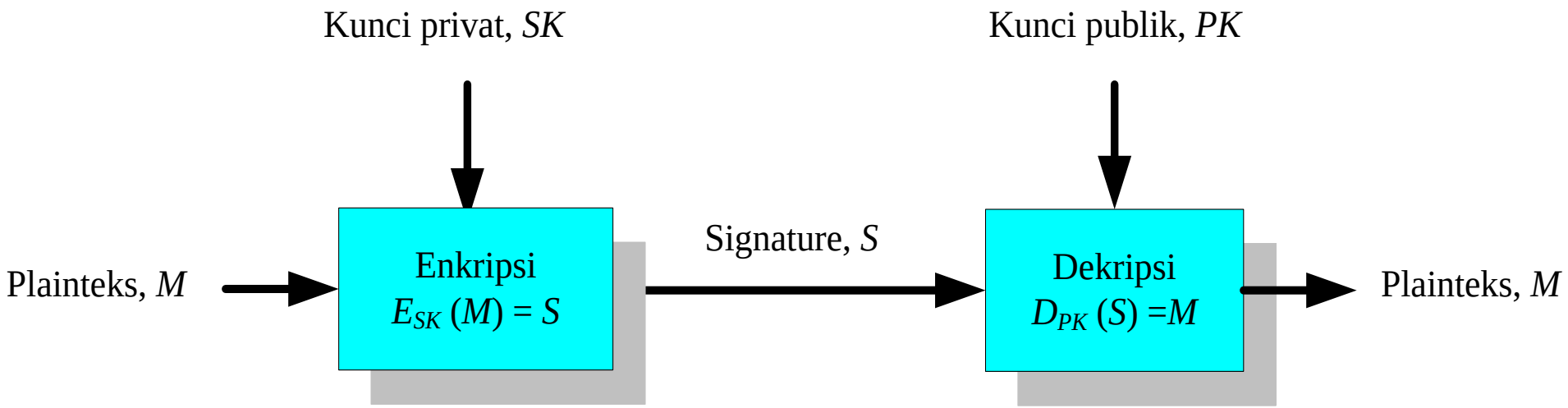
- pesan didekripsi dengan kunci privat penerima.

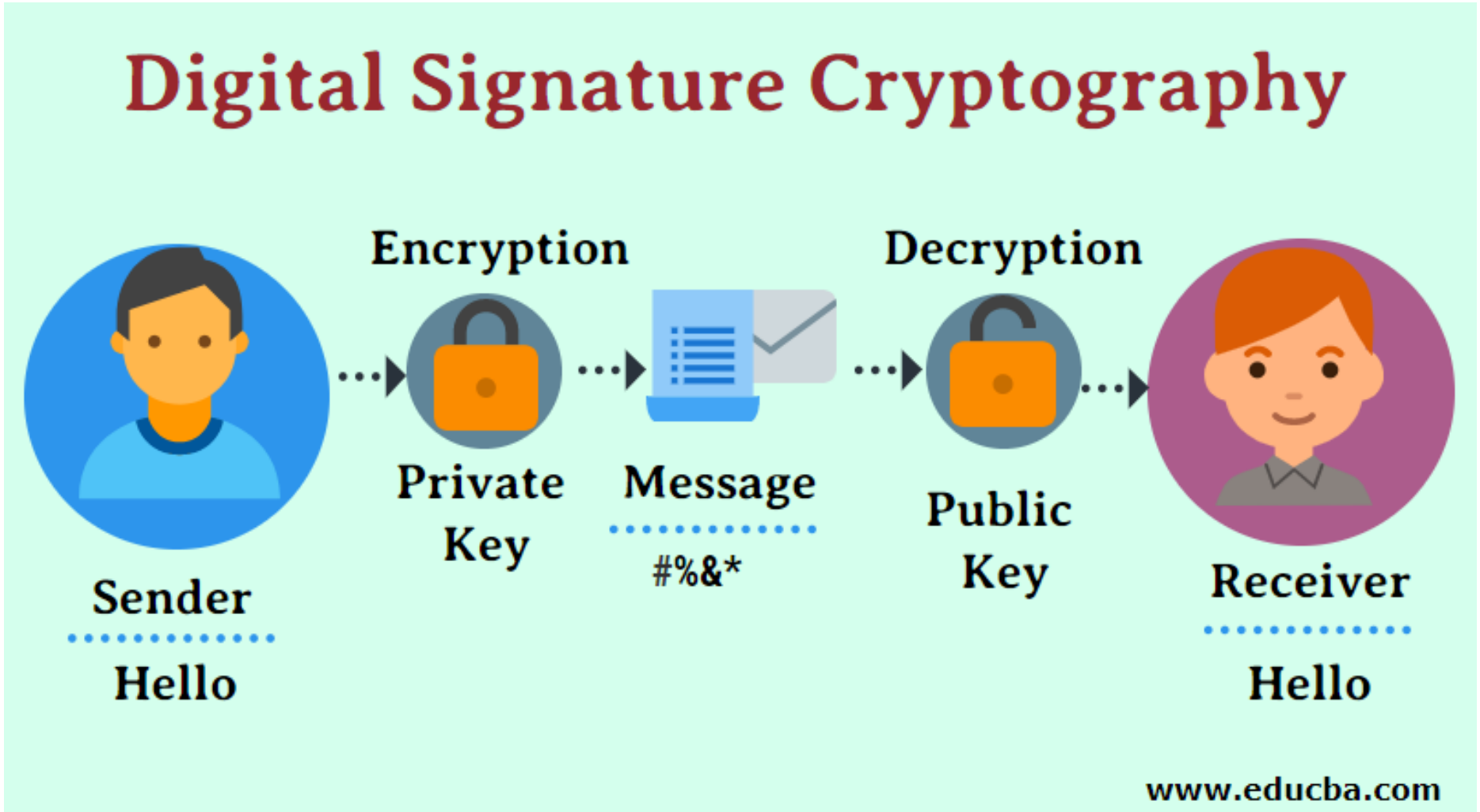
- cara ini tidak memberikan bukti otentikasi karena kunci publik diketahui oleh banyak orang



Enkripsi sebagai tanda-tangan:

- pesan dienkripsi kunci privat pengirim.
- pesan didekripsi pesan dengan kunci publik pengirim.
- dengan cara ini, maka kerahasiaan pesan dan otentikasi keduanya dicapai sekaligus.
- ide ini ditemukan oleh Diffie dan Hellman.





Sumber: https://www.educba.com/digital-signature-cryptography/

- Proses menandatantangani pesan (oleh pengirim):

  $S = E_{SK}(M)$

- Proses membuktikan otentikasi pesan (oleh penerima):

  $M = D_{PK}(S)$

  Keterangan:

  SK = *secret key* = kunci privat pengirim

  PK = *public key* = kunci publik pengirim

  E = fungsi enkripsi *D* = fungsi dekripsi

  M = pesan semula

  S = *signature* = hasil enkripsi pesan

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

- Beberapa agoritma kunci-publik dapat digunakan untuk menandatangani pesan dengan cara mengenkripsinya, asalkan algoritma tersebut memenuhi sifat:

$$D_{SK}(E_{PK}(M)) = M \text{ dan } D_{PK}(E_{SK}(M)) = M ,$$

Keterangan:

PK = kunci publik ; *SK* = kunci privat (*secret key*).

E = fungsi enkripsi; *D* = fungsi dekripsi

M = pesan

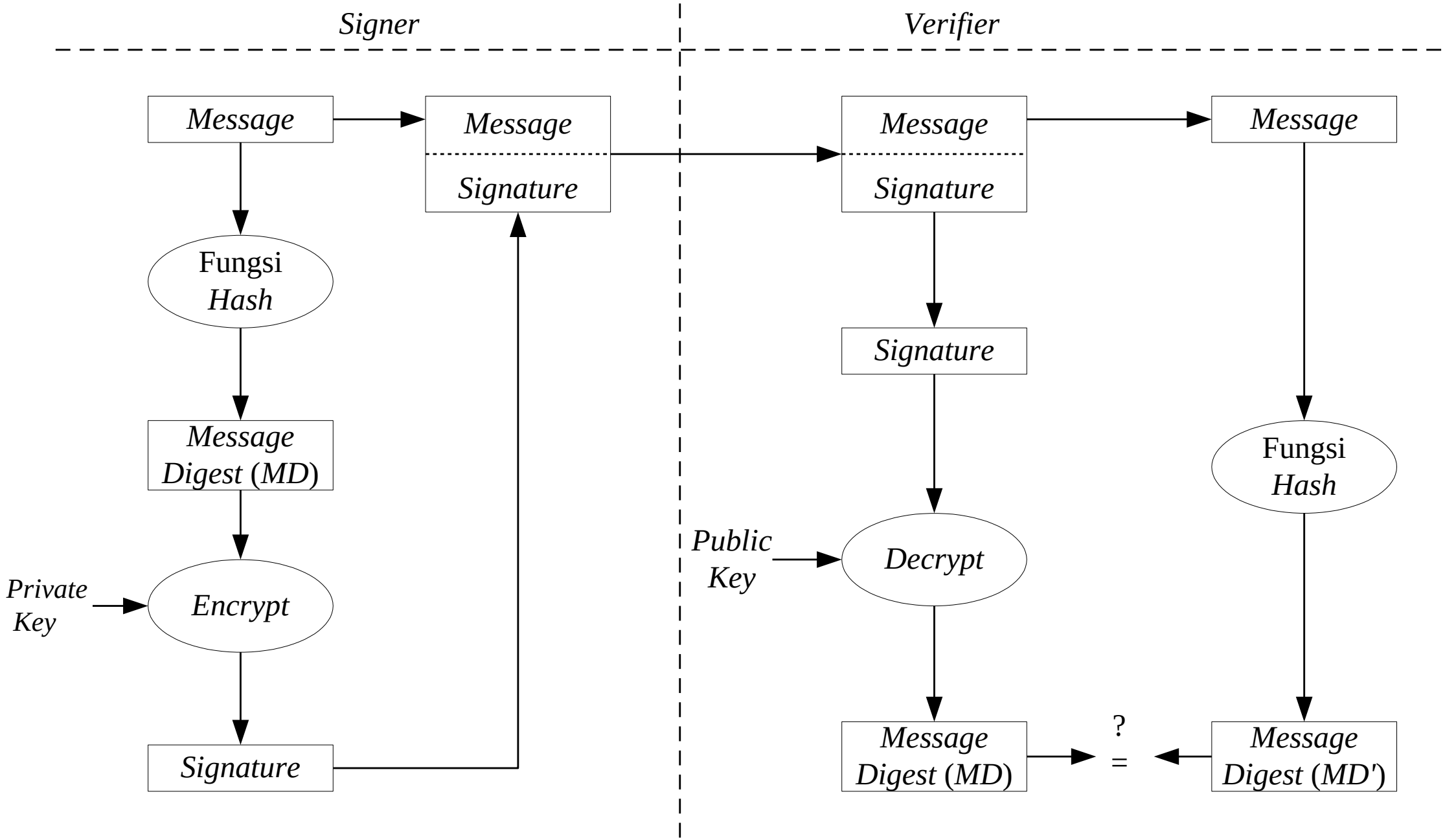
- Contoh algoritma yang memenuhi sifat ini adalah RSA

Penandatangan dengan Menggunakan Kriptografi kunci-publik dan 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 dienkripsi, sebab yang dibutuhkan hanya keotentikan pesan saja.

- Algoritma kunci-publik dan fungsi *hash* dapat digunakan untuk kasus seperti ini.


Signer
Verifier
Message
Message
Signature
Message
Signature
Message
Fungsi
Hash
Signature
Message
Digest (MD)
Fungsi
Hash
Public
Key
Private
Key
Encrypt
Decrypt
Signature
Message
Digest (MD)
?
=
Message
Digest (MD')

- 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

Tanda-tangan dengan algoritma RSA

Langkah-langkah pemberian tanda-tangan (*signing*)

1. Pengirim menghitung nilai *hash* dari pesan *M:*

   $h = H(M)$

2. Pengirim mengenkripsi *h* dengan kunci privatnya (SK) menggunakan persamaan enkripsi *RSA,* hasilnya adalah signature *S*:

   $S = h^{SK} \bmod n$ (*n* adalah modulus, $n = pq$).

3. Pengirim mentransmisikan *M* +*S* ke penerima

Langkah-langkah verifikasi tanda-tangan (*verifying*)

1. Penerima menghitung nilai *hash* dari pesan *M* yang diterima:

   $h = H(M)$

2. Penerima melakukan dekripsi terhadap tanda-tangan *S* dengan kunci publik si pengirim (PK) menggunakan persamaan dekripsi *RSA*:

   $h' = S^{PK} \bmod n$

3. Penerima membandingkan *h* dengan *h'*. Jika *h* = *h'* maka tanda-tangan digital adalah otentik. Jika tidak sama, maka tanda-tangan tidak otentik sehingga pesan dianggap tidak asli lagi atau pengirimnya

Alice

M =

Pada wisuda sarjana baru ITB, ternyata ada seorang wisudawan yang paling muda. Umurnya baru 19 tahun. Ini berarti dia masuk ITB pada umur 15 tahun. Zaman sekarang banyak sarjana masih berusia muda belia. Mungkin masuk sekolah pada usia dini dan mengikuti kelas akselerasi pada tingkatan SD, SMP, dan SMA. Masuk SD umur 6 tahun dan ikut kelas aksel sehingga selesai dalam waktu lima tahun pada umur 11. SMP diselesaikan dalam waktu dua tahun dan SMA dalam waktu dua tahun, sehingga lulus SMA pada umur 15 tahun. Kuliah di ITB selama empat tahun sehingga wajar saja menjadi sarjana pada umur 19 tahun.

$h = H(M)$ = A4C05176E1440FC879C06C72FA603A24 (heksadesimal)

= 2189919645993823712285540132954717701 48 (desimal

$S = h^{PrivK} \bmod n$ (n = 223427, $PrivK$ = 171635)

$= (2189919645993823712285540132954717701 48)^{171635} \bmod (223427) = 46489$

$M + S$ =

Pada wisuda sarjana baru ITB, ternyata ada seorang wisudawan yang paling muda. Umurnya baru 19 tahun. Ini berarti dia masuk ITB pada umur 15 tahun. Zaman sekarang banyak sarjana masih berusia muda belia. Mungkin masuk sekolah pada usia dini dan mengikuti kelas akselerasi pada tingkatan SD, SMP, dan SMA. Masuk SD umur 6 tahun dan ikut kelas aksel sehingga selesai dalam waktu lima tahun pada umur 11. SMP diselesaikan dalam waktu dua tahun dan SMA dalam waktu dua tahun, sehingga lulus SMA pada umur 15 tahun. Kuliah di ITB selama empat tahun sehingga wajar saja menjadi sarjana pada umur 19 tahun.

46489

Bob

Pada wisuda sarjana baru ITB, ternyata ada seorang wisudawan yang paling muda. Umurnya baru 19 tahun. Ini berarti dia masuk ITB pada umur 15 tahun. Zaman sekarang banyak sarjana masih berusia muda belia. Mungkin masuk sekolah pada usia dini dan mengikuti kelas akselerasi pada tingkatan SD, SMP, dan SMA. Masuk SD umur 6 tahun dan ikut kelas aksel sehingga selesai dalam waktu lima tahun pada umur 11. SMP diselesaikan dalam waktu dua tahun dan SMA dalam waktu dua tahun, sehingga lulus SMA pada umur 15 tahun. Kuliah di ITB selama empat tahun sehingga wajar saja menjadi sarjana pada umur 19 tahun.
46489

$h = H(M) = \text{A4C05176E1440FC879C06C72FA603A24}$ (heksadesimal)

$= 218991964599382371228554013295471770148$ (decimal)

$\equiv 125468 \pmod{223427}$

$h' = S^{PubK} \bmod n \quad (PubK = 731)$

$= (46489)^{731} \bmod (223427)$

$= 125468$

sama

Tandda tangan valid!

Referensi utama :

>> Michael Felderer , Riccardo Scandariato (editor) - Exploring Security in Software Architecture and Design, 2018.

>> Nancy R. Mead, Carol Woody - Cyber Security Engineering_ A Practical Approach for Systems and Software Assurance-Addison-Wesley Professional (2016)

>> James Helfrich - Security for Software Engineers-CRC Press (2019)

>> Pete Loshin - Simple Steps to Data Encryption_ A Practical Guide to Secure Computing-Syngress (2013)

>> Tevfik Bultan,Fang Yu,Muath Alkhalaf,Abdulbaki Aydin (auth.) - String Analysis for Software Verification and Security (2017


Ada pertanyaan?