*Bahan Kuliah ke-14*

**IF5054 Kriptografi**

# Sistem Kriptografi Kunci-Publik

**Disusun oleh:**

**Ir. Rinaldi Munir, M.T.**

**Departemen Teknik Informatika**
**Institut Teknologi Bandung**
**2004**

# 14. Sistem Kriptografi Kunci-Publik

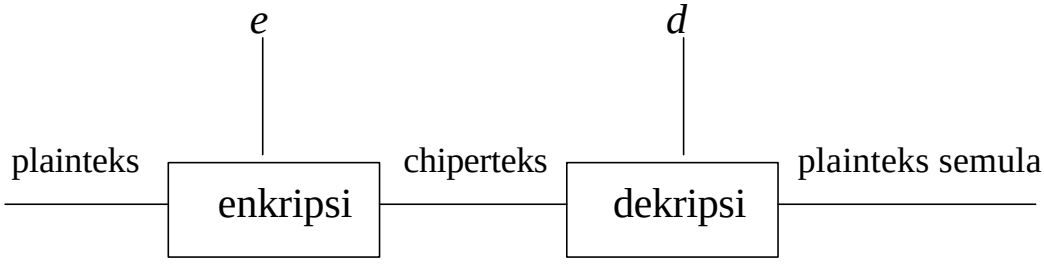
## 14.1 Pendahuluan

- Sampai akhir tahun 1970, hanya ada sistem kriptografi simetri. Karena sistem kriptografi simetri menggunakan kunci yang sama untuk enkripsi dan dekripsi, maka hal ini mengimplikasikan dua pihak yang berkomunikasi saling mempercayai. Kedua pihak harus menjaga kerahasiaan kunci (sehingga, junci enkripsi/dekripsi disebut juga *secret key*)

- Pada sistem kriptografi kunci-publik, kunci kriptografi dibuat sepasang, satu kunci untuk enkripsi dan satu kunci untuk dekripsi (Gambar 14.1);
  - Kunci untuk enkripsi diumumkan kepada publik – oleh karena itu tidak rahasia – sehingga dinamakan **kunci publik** (*public-key*), disimbolkan dengan *e*.

  - Kunci untuk dekripsi bersifat rahasia – sehingga dinamakan **kunci privat** (*private key*), disimbolkan dengan *d*.

  Karena ada kunci enkripsi ≠ kunci dekripsi, maka sistem kriptografi kunci-publik kadang-kadang disebut juga sistem **kriptografi asimetri**.




**Gambar 1.** Sistem kriptografi kunci-publik.
Ket: *e = public key, d = private key*

- Sistem kriptografi kunci-publik didasarkan pada fakta:
  1. Komputasi untuk enkripsi/dekripsi pesan mudah dilakukan.
  2. Secara komputasi hampir tidak mungkin (*infeasible*) menurunkan kunci privat, *d*, bila diketahui kunci publik, *e*, pasangannya.

  Kedua fakta di atas analog dengan:
  - Perkalian *vs* pemfaktoran
    Mengalikan dua buah bilangan prima, $a \times b = n$, mudah, tetapi memfaktorkan *n* menjadi faktor-faktor primanya sulit.
    Contoh: $31 \times 47 = 1457$ (perkalian)
    $1457 = ? \times ?$ (pemfaktoran)

  - Perpangkatan *vs* logaritmik
    Melakukan perpangkatan, $y = a^x$, mudah, tetapi menghitung $x = {}^a\log y$ sulit jika *a* tidak diketahui.
    Contoh: $12^5 = 248832$ (perpangkatan)
    $x = {}^a\log 248832 = ?$ (logaritmik)

## 14.2 Konsep Kriptografi Kunci-Publik

- Konsep kriptografi kunci-publik sederhana dan elegan, tetapi mempunyai konsekuensi penggunaan yang hebat.

- Misalkan *E* adalah fungsi enkripsi dan *D* adalah fungsi dekripsi. Misalkan (*e*, *d*) adalah pasangan kunci untuk enkripsi dan dekripsi sedemikian sehingga

  $$E_d(m) = c \quad \text{dan } D_d(c) = m$$

  untuk suatu plainteks *m* dan cipherteks *c*.

Kedua persamaan ini menyiratkan bahwa dengan mengetahui *e* dan *c*, maka secara komputasi hampir tidak mungkin menemukan *m*. Asumsi lainnya, dengan mengetahui *e*, secara komputasi hampir tidak mungkin menurunkan *d*.

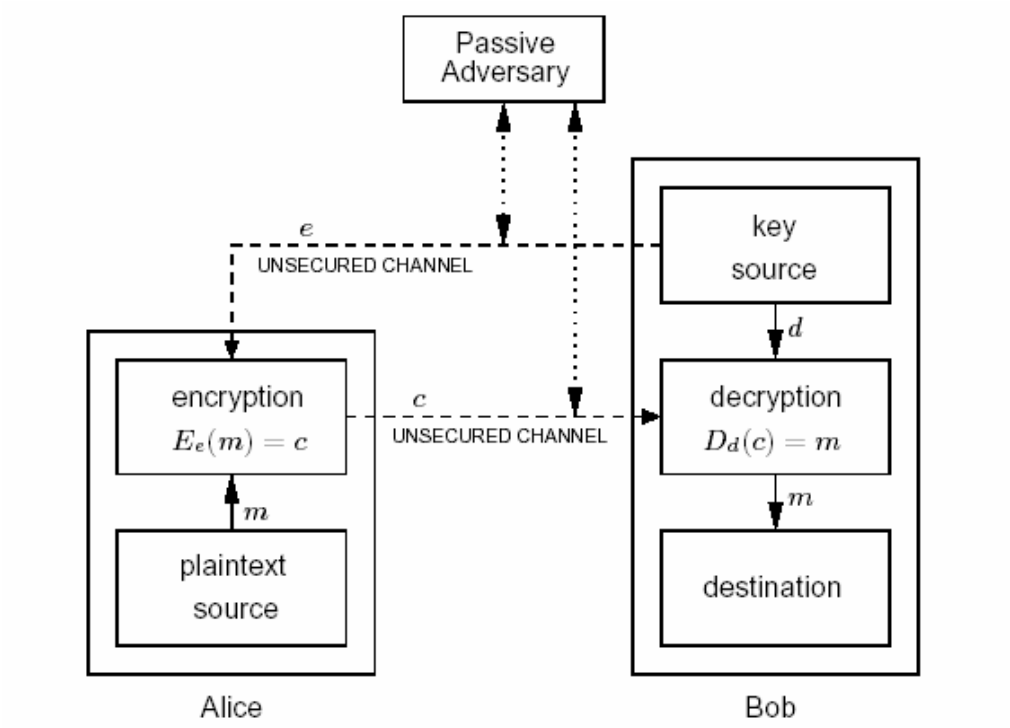
$E_e$ digambarkan sebagai fungsi pintu-kolong (*trapdoor*) satu-arah dengan *d* adalah informasi *trapdoor* yang diperlukan untuk menghitung fungsi inversinya, *D,* yang dalam hal ini membuat proses dekripsi dapat dilakukan.

- Konsep di atas menjadi penting bila kriptografi kunci-publik digunakan untuk mengamankan pertukaran pesan dari dua entitas yang berkomunikasi.

  Misalkan Alice berkomunikasi dengan Bob. Bob memilih pasangan kunci (*e*, *d*). Bob mengirimkan kunci enkripsi *e* (kunci publik) kepada Alice melalui sembarang saluran tetapi tetap menjaga kerahasiaan kunci dekripsinya, *d* (kunci privat).

  Kemudian, Alice ingin mengirim pesan *m* kepada Bob. Alice mengenkripsikan pesan *m* dengan menggunakan kunci publik Bob, untuk mendapatkan $c = E_e(m)$, lalu mengirimkan *c* melalui saluran komunikasi (yang tidak perlu aman). Bob mendekripsi cipherteks *c* dengan menggunakan kunci privatnya untuk memperoleh $m = D_d(c)$,

  Perhatikan skema komunikasi dengan kriptografi kunci-publik pada Gambar 14.2. Gambar ini memperlihatkan perbedaan mendasar sistem asimetri dengan sistem simetri. Di sini kunci enkripsi dikirim kepada Alice melalui saluran yang tidak perlu aman (*unsecure channel*). Saluran yang tidak perlu aman ini mungkin sama dengan saluran yang digunakan untuk mengirim cipherteks.




**Gambar 14.2** Enkripsi/dekripsi dengan kriptorafi kunci-publik.

- Sistem kriptografi kunci-publik juga cocok untuk kelompok pengguna di lingkungan jaringan komputer (LAN/WAN). Setiap pengguna jaringan mempunyai pasangan kunci publik dan kunci privat yang bersuaian. Kunci publik, karena tidak rahasia, biasanya disimpan di dalam basisdata kunci yang dapat diakses oleh pengguna lain. Jika ada pengguna yang hendak berkirim pesan ke pengguna lainnya, maka ia ia perlu mengetahui kunci publik penerima pesan melalui basisdata kunci ini lalu menggunakannya untuk mengenkripsi pesan. Hanya penerima pesan yang berhak yang dapat mendekripsi pesan karena ia mempunyai kunci privat.

- Dengan sistem kriptografi kunci-publik, tidak diperlukan pengiriman kunci privat melalui saluran komunikasi khusus sebagaimana pada sistem kriptografi simetri.

- Meskipun kunci publik diumumkan ke setiap orang di dalam kelompok, namun kunci publik perlu dilindungi agar otentikasinya terjamin (misalnya tidak diubah oleh orang lain).

## 14.3 Kriptografi Simetri *vs* Kriptografi Asimetri

- Baik kriptografi simetri maupun kriptografi asimetri (kunci-publik), keduanya mempunyai kelebihan dan kelemahan.

- **Kelebihan kriptografi simetri:**
  1. Algoritma kriptografi simetri dirancang sehingga proses enkripsi/dekripsi membutuhkan waktu yang singkat.
  2. Ukuran kunci simetri relatif pendek.
  3. Algoritma kriptografi simetri dapat digunakan untuk membangkitkan bilangan acak.
  4. Algorima kriptografi simetri dapat disusun untuk menghasilkan *cipher* yang lebih kuat.
  5. Otentikasi pengirim pesan langsung diketahui dari cipherteks yang diterima, karena kunci hanya diketahui oleh pengirim dan penerima pesan saja.

- **Kelemahan kriptografi simetri:**
  1. Kunci simetri harus dikirim melalui saluran yang aman. Kedua entitas yang berkomunikasi harus menjaga kerahasisan kunci ini.
  2. Kunci harus sering diubah, mungkin pada setiap sesi komunikasi.

- **Kelebihan kriptografi kunci-publik (asimetri):**
  1. Hanya kunci privat yang perlu dijaga kerahasiaannya oleh seiap entitas yang berkomuniaksi (tetapi, otentikasi kunci publik tetap harus terjamin). Tidak ada kebutuhan mengirim kunci kunci privat sebagaimana pada sistem simetri.
  2. Pasangan kunci publik/kunci privat tidak perlu diubah, bahkan dalam periode waktu yang panjang.
  3. Dapat digunakan untuk mengamankan pengiriman kunci simetri.
  4. Beberapa algoritma kunci-publik dapat digunakan untuk memberi tanda tangan digital pada pesan (akan dijelaskan pada materi kuliah selanjutnya)

- **Kelemahan kriptografi kunci-publik (asimetri):**
  1. Enkripsi dan dekripsi data umumnya lebih lambat daripada sistem simetri, karena enkripsi dan dekripsi menggunakan bilangan yang besar dan melibatkan operasi perpangkatan yang besar.
  2. Ukuran cipherteks lebih besar daripada plainteks (bisa dua sampai empat kali ukuran plainteks).
  3. Ukuran kunci relatif lebih besar daripada ukuran kunci simetri.
  4. Karena kunci publik diketahui secara luas dan dapat digunakan setiap orang, maka cipherteks tidak memberikan informasi mengenai otentikasi pengirim.
  5. Tidak ada algoritma kunci-publik yang terbukti aman (sama seperti *block cipher*). Kebanyakan aalgoriam mendasakan keamanannya pada sulitnya memecahkan persoalan-persoalan aritmetik (pemfaktoran, logaritmik, dsb) yang menjadi dasar pembangkitan kunci.

## 14.4 Aplikasi Kriptografi Knci-Publik

- Aplikasi kriptogafi kunci-publik dapat dibagi menjadi 3 kategori:
  1. Enkripsi/dekripsi
     Seperti pada algoritma kriptografi simetri, algoritma kunci-publik dapat digunakan untuk menjaga kerahasiaan pesan (*provide confidentiality/secrecy*).
     Contoh algoritma: *RSA, Knapsack, Rabin, ElGamal*

  2. *Digital signatures*
     Algoritma kriptografi kunci-publik dapat digunakan untuk membuktikan otentikasi pesan maupun otentikasi pengirim (*provide authentication*)
     Contoh algoritma: *RSA, DSA, ElGamal, GOST*

  3. Pertukaran kunci (*key exchange*)
     Algoritma kriptografi kunci-publik dapat digunakan untuk pengiriman kunci simetri (*session keys*)
     Contoh algoritma: *RSA, Diffie-Hellman*

- Beberapa algoritma kriptografi kunci-publik cocok digunakan untuk ketiga macam kategori aplikasi (misalnya *RSA*), beberapa algoritma hanya ditujukan untuk aplikasi spesifik (misalya *DSA* untuk *digital signature*).