*Bahan Kuliah ke-24*

**IF5054 Kriptografi**

# *Manajemen Kunci*

**Disusun oleh:**

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

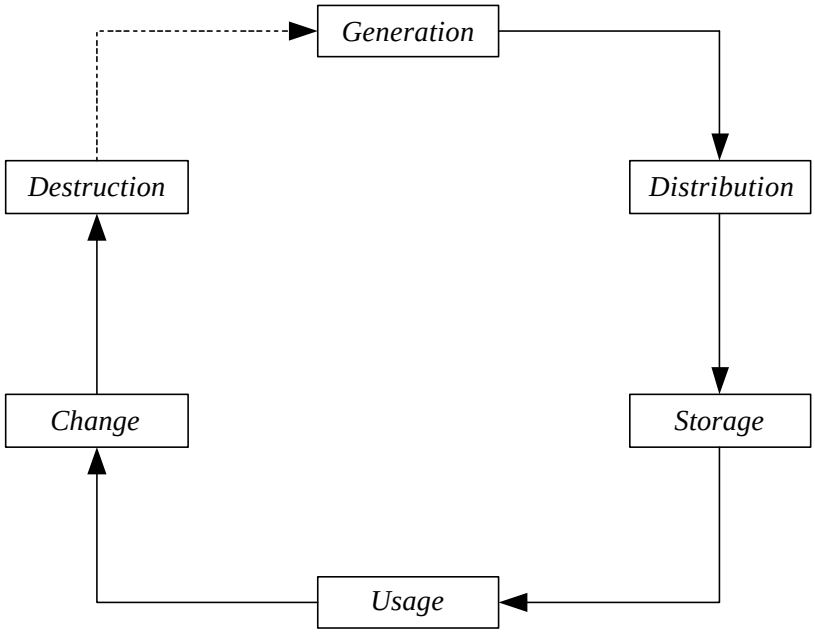
**Departemen Teknik Informatika**
**Institut Teknologi Bandung**
**2004**

# 24. Manajemen Kunci

## 24.1 Pendahuluan

- Kekuatan sistem kriptografi secara total bergantung pada keamanan kunci. Kunci perlu dilindungi selama fase daur hidupnya.

- Daur hidup kunci dimulai dari pembangkitan kunci (*generation*) sampai kunci tidak diperlukan lagi untuk kemudian dihancurkan (*destruction*). Secara garis besar, daur hidup kunci digambarkan pada Gambar 24.1 sbb:




**Gambar 24.1.** Daur hidup kunci

- Tujuan manajemen kunci adalah menjaga keamanan dan integritas kunci pada semua fase di dalam daur hidupnya. Pada umumnya setiap kunci akhirnya diganti dengan kunci lain. Jadi, keseluruhan fase membentuk siklus (lingkaran) karena penghancuran kunci biasanya diikuti dengan penggantiannya dengan kunci baru (garis putus-putus).

- Manajemen kunci yang dibahas difokuskan pada algoritma kriptografi simetri karena manajemen kunci untuk algoritma kunci-publik sangat berbeda dengan algoritma simetri.

## 24.2 Pembangkitan Kunci (*Key Generation*)

- Pembangkitan kunci pada algoritma simetri jauh lebih mudah daripada pembangkitan kunci pada algoritma kunci-publik. Karena kunci simetri umumnya rangkaian bii atau rangkaian karakter, maka setiap pengguna dapat membangkitkan kuncinya sendiri.

- Masalah utama yang muncul pada pembangkitan kunci adalah bagaimana membuat kunci yang tidak dapat diprediksi. Metode yang dapat digunakan untuk menjawab hal ini adalah dengan teknik manual (misalnya pelemparan koin/dadu), pembangkitan dari data pribadi (misalnya PIN), atau menggunakan pembangkit bilangan acak.

- Pada algoritma kunci-publik, pembangkitan kunci merupakan masalah tersendiri, karena pembangkitan kunci membutuhkan perhitungan matematis yang rumit. Selain itu, pembangkitan bilangan prima yang besar juga dibutuhkan untuk membentuk kunci.

- Oleh karena itu, pada algoritma kunci-publik dibutuhkan program khusus untuk membangkitkan kunci. Masalah yang timbul di sini adalah kepercayaan pengguna terhadap program tersebut. Pada RSA misalnya, bila program hanya dapat membangkitkan bilangan prima yang terbatas, maka pihak lawan dapat membangkitkan sendiri bilangan-bilangan prima yang terbatas itu dan menggunakannya sebagai faktor dari salah satu parameter *RSA*.

## 24.3 Penyebaran Kunci (*Key Distribution*)

- Jika pengguna menggunakan kunci untuk melindungi informasi yang disimpan di dalam *storage*, maka tidak ada kebutuhan untuk menyebarkan kunci.

- Tetapi, untuk kebutuhan komunikasi secara aman, maka diperlukan kebutuhan untuk mengirimkan kunci.

- Protokol pertukaran kunci dengan menggunakan algoritma kunci-publik (lihat pembahasan Protokol Kriptografi) dapat digunakan untuk mendistribusikan kunci.

## 24.4 Penyimpanan Kunci (*Key Storage*)

- Kunci disimpan di tempat yang aman yang tidak memungkinkan pihak lawan mengaksesnya. Oleh karena itu, penyimpanan kunci mungkin memerlukan perlindungan secara fisik (misalnya disimpan di dalam lemari besi).

- Alternatif lain, kunci dapat disimpan di dalam *smart card* yang hanya dapat dibaca dengan menggunakan kode rahasia.

- Kunci sebaiknya disimpan tidak dalam bentuk jelas. Ada dua solusi alternatif untuk masalah ini.
  1. kunci disimpan dengan mengenkripsinya dengan menggunakan kunci lain. Konsep ini mengarah pada konsep *key hierarchy*, yang dalam hal ini setiap kunci di dalam hirarkhi digunakan untuk melindungi kunci di bawahnya.

  2. kunci dipecah menjadi beberapa komponen, setiap komponen disimpan di tempat terpisah. Jika kunci akan digunakan, maka setiap komponen direkonstruksi kembali.

- Misalkan kunci $K$ dibagi menjadi dua komponen, $K_1$ dan $K_2$. Membagi dua langsung $K$ sedemikian sehingga setengah bagian pertama menjadi $K_1$ dan setengah bagian sisanya menjadi $K_2$ tidak dianjurkan, karena dapat memungkinkan pihak lawan menemukan $K$ jika ia hanya mengetahui salah satu dari $K_1$ dan $K_2$. Misalkan $K$ panjangnya 64 bit, dan lawan mengetahui $K_1$, maka $K$ dapat ditentukan dengan hanya $2^{32}$ percobaaan untuk menemukan $K_2$ secara *exhaustive search* (lebih sedikit dibandingkan $2^{64}$ percobaan).

- Solusi pemecahan yang lebih baik adalah membentuk kunci $K$ dari $K_1$ dan $K_2$ sedemikian sehingga $K = K_1 \oplus K_2$. Dalam hal ini, ukuran $K_1$ dan $K_2$ sama dengan ukuran $K$, sehingga jika salah satu dari komponen $K_1$ atau $K_2$ diketahui, maka $K$ relatif lebih sukar ditentukan.

## 24.5 Penggunaan Kunci (*Key Usage*)

- Setiap kunci digunakan sesuai tujuannya. Misalnya ada kunci yang digunakan untuk mengenkripsi pesan, dan ada kunci yang digunakan untuk mengenkripsi kunci lainnya.

- Supaya setiap kunci mempunyai penggunaan yang unik, maka kita perlu membeli label pada setiap kunci, yang dalam hal ini label menspesifikasikan penggunaan kunci. Misalnya, label tersebut menspesifikasikan 'kunci untuk mengenkripsi data', 'kunci untuk mengenkripsi kunci', 'kunci untuk pembangkitan bilangan acak', dan sebagainya.

- Untuk algoritma kunci-publik, pengguna perlu memberi label untuk dua pasang kunci yang setiap pasang terdiri dari kunci publik dan kunci rahasia. Satu pasang kunci untuk enkripsi dan satu pasang lagi untuk sidik dijital.

## 24.6 Perubahan Kunci (*Key Change*)

- Kunci sebaiknya diubah secara periodik dan teratur. Sistem kriptografi harus mempunyai kemampuan untuk mengubah kunci.

- Kunci diubah secara teratur untuk membatasi lama keberadaanya dan mengurangi nilainya dimata penyerang.

- Pada sistem EFTPOS (*Electronic Funds Transfer at Point of Sale*), kunci diubah setiap kali setelah selesai satu transaksi.

- Tidak ada aturan seberapa sering kunci seharusnya diubah. Tetapi cukup jelas dimengerti bahwa setiap kunci seharusnya

diubah jauh sebelum ia dapat ditemukan dengan cara *exhaustive search*.

## 24.7 Penghancuran Kunci (*Key Destruction*)

- Kunci yang tidak dibutuhkan lagi seharusnya dihancurkan dengan cara yang aman.

- Jika kunci dicatat pada media kertas, maka cara penghancurannya misalnya menggunakan alat pemotong kertas (crosscut), membakarnya, atau menguburnya.

- Jika kunci disimpan di dalam media elektronik (seperti CD), maka cara penghancurannya bisa dengan menghapusnya atau menimpanya (*overwritten*) sedemikian sehingga tidak meninggalkan jejak yang bisa dilacak oleh penyerang.

- Kunci yang yang disimpan pada material lain dihancurkan sedemikian rupa sehingga ia tidak mungkin ditemukan kembali secara fisik maupun secara elektronik.