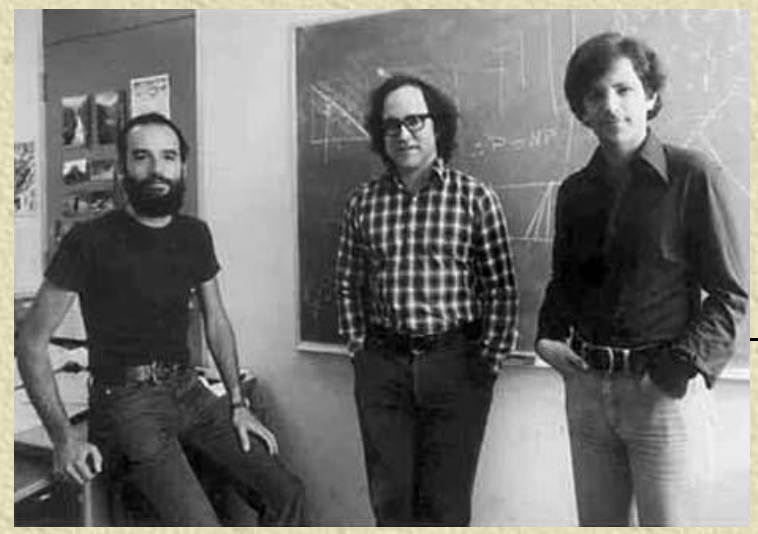
# Algoritma RSA

Bahan Kuliah

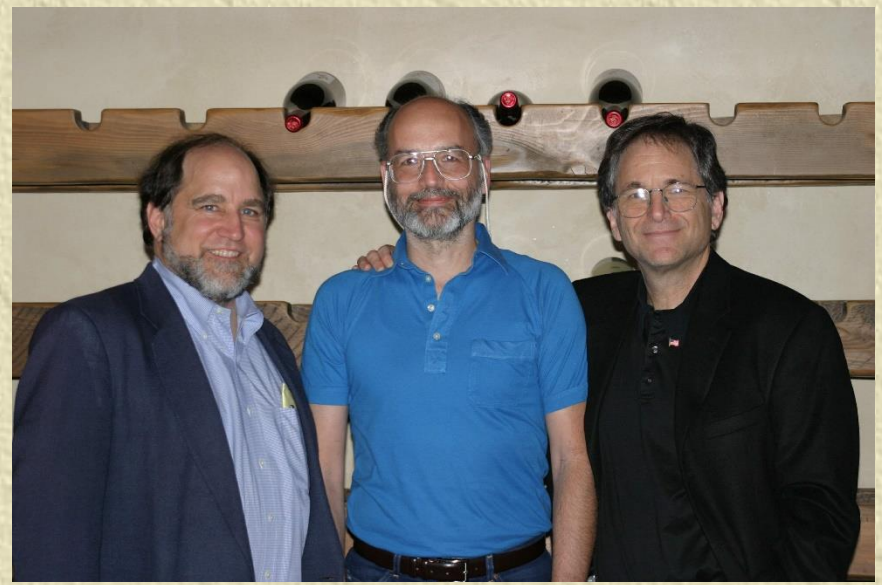
IF4020 Kriptografi

# Pendahuluan

- Algoritma kunci-publik yang paling terkenal dan paling banyak aplikasinya.
- Ditemukan oleh tiga peneliti dari *MIT* (*Massachussets Institute of Technology*), yaitu Ron Rivest, Adi Shamir, dan Len Adleman, pada tahun 1976.
- Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima.



dahulu



sekarang

The authors of RSA: Rivest, Shamir and Adleman

# Properti Algoritma RSA

1. $p$ dan $q$ bilangan prima (rahasia)
2. $n = p \cdot q$ (tidak rahasia)
3. $\phi(n) = (p - 1)(q - 1)$ (rahasia)
4. $e$ (kunci enkripsi) (tidak rahasia)

   Syarat: PBB($e$, $\phi(n)$) = 1
5. $d$ (kunci dekripsi) (rahasia)

   $d$ dihitung dari $d \equiv e^{-1} \bmod (\phi(n))$
6. $m$ (plainteks) (rahasia)
7. $c$ (cipherteks) (tidak rahasia)

# Penurunan Rumus RSA

- Prinsip: Teorema Euler $a^{\phi(n)} \equiv 1 \pmod{n}$
- Syarat:
  1. $a$ harus relatif prima terhadap $n$
  2. $\phi(n)$ = Toitent Euler = fungsi yang menentukan berapa banyak dari bilangan-bilangan 1, 2, 3, …, $n$ yang relatif prima terhadap $n$.

     Contoh: $\phi(20) = 8$, sebab terdapat 8 buah yang relatif prima dengan 20, yaitu 1, 3, 7, 9, 11, 13, 17, 19.

     Jika $n = pq$ adalah bilangan komposit dengan $p$ dan $q$ prima, maka $\phi(n) = \phi(p)\,\phi(q) = (p-1)(q-1)$.

$a^{\phi(n)} \equiv 1 \pmod{n}$

$\downarrow$ (pangkatkan kedua ruas dengan k)

$a^{k\phi(n)} \equiv 1^k \pmod{n}$

$\downarrow$

$a^{k\phi(n)} \equiv 1 \pmod{n}$

$\downarrow$ (ganti $a$ dengan $m$)

$m^{k\phi(n)} \equiv 1 \pmod{n}$

$\downarrow$ (kalikan kedua ruas dengan $m$)

$m^{k\phi(n)+1} \equiv m \pmod{n}$

- Misalkan $e$ dan $d$ dipilih sedemikian sehingga

  $$e \cdot d \equiv 1 \pmod{\phi(n)}$$

  atau

  $$e \cdot d = k\phi(n) + 1$$

  Maka

  $$m^{k\phi(n)+1} \equiv m \pmod{n}$$

  $$\downarrow$$

  $$m^{e \cdot d} \equiv m \pmod{n} \rightarrow (m^{e})^{d} \equiv m \pmod{n}$$

- Enkripsi: $E_e(m) = c \equiv m^e \bmod n$
- Dekripsi: $D_d(c) = m \equiv c^d \bmod n$

# Pembangkitan Sepasang Kunci

1. Pilih dua bilangan prima, $p$ dan $q$ (rahasia)
2. Hitung $n = pq$.
3. Hitung $\phi(n) = (p - 1)(q - 1)$.
4. Pilih sebuah bilangan bulat $e$ untuk kunci publik, sebut, $e$ relatif prima terhadap $\phi(n)$ .
5. Hitung kunci dekripsi, $d$, dengan persamaaan

   $ed \equiv 1$ (**mod** $\phi(n)$) atau $d \equiv e^{-1} \bmod (\phi(n))$

Hasil dari algoritma di atas:

- Kunci publik adalah pasangan $(e, n)$
- Kunci privat adalah pasangan $(d, n)$

# Enkripsi

1. Nyatakan pesan menjadi blok-blok plainteks: $m_1$, $m_2$, $m_3$, … ( syarat: $0 < m_i < n - 1$)

2. Hitung blok cipherteks $c_i$ untuk blok plainteks $p_i$ dengan persamaan

$$c_i = m_i^e \textbf{ mod } n$$

yang dalam hal ini, $e$ adalah kunci publik.

# Dekripsi

Proses dekripsi dilakukan dengan menggunakan persamaan

$$m_i = c_i^d \textbf{ mod } n,$$

yang dalam hal ini, $d$ adalah kunci privat.

# Contoh:

- Misalkan dipilih $p = 47$ dan $q = 71$ (keduanya prima), maka dapat dihitung:

  $$n = p \times q = 3337$$

  $$\phi(n) = (p - 1) \times (q - 1) = 3220.$$

- Pilih kunci publik $e = 79$ (yang relatif prima dengan 3220 karena pembagi bersama terbesarnya adalah 1).

- Nilai $e$ dan $n$ dapat dipublikasikan ke umum.

- Selanjutnya akan dihitung kunci privat $d$ dengan kekongruenan:

$$e \times d \equiv 1 \pmod{\phi(n)}$$

$d$ adalah invers $e$ dalam modulus $\phi(n)$

$d$ dapat dihitung dengan algoritma Euclidean atau dengan perhitungan coba-coba sebagai berikut:

$$d = \frac{1 + (k \times 3220)}{79}$$

Dengan mencoba nilai-nilai $k$ = 1, 2, 3, …, diperoleh nilai $d$ yang bulat adalah 1019. Ini adalah kunci privat (untuk dekripsi).

- Misalkan plainteks $M$ = ‘HARI INI’
  atau dalam ASCII: 726582733273873

  Pecah $M$ menjadi blok yang 3 digit:

  $m_1 = 726$ $m_4 = 273$
  $m_2 = 582$ $m_5 = 787$
  $m_3 = 733$ $m_6 = 003$

(Perhatikan, $m_i$ masih terletak antara 0 sampai $n - 1 = 3337$)

- *Enkripsi setiap blok:*

  $c_1 = 726^{79} \bmod 3337 = 215$

  $c_2 = 582^{79} \bmod 3337 = 776$

  dst

  Hasil: $C$ = 215 776 1743 933 1731 158.

- *Dekripsi (menggunakan kunci privat d = 1019)*

  $m_1 = 215^{1019} \bmod 3337 = 726$

  $m_2 = 776^{1019} \bmod 3337 = 582$

  dst untuk sisi blok lainnya

  Plainteks $M$ = 726582733273 7873

  yang dalam ASCII adalah ‘HARI INI’.

# *Kekuatan dan Keamanan RSA*

- Kekuatan algoritma *RSA* terletak pada tingkat kesulitan dalam memfaktorkan bilangan menjadi faktor-faktor prima, yang dalam hal ini $n = a \times b$.

- Sekali $n$ berhasil difaktorkan menjadi $a$ dan $b$, maka $\phi(n) = (a - 1)\times(b - 1)$ dapat dihitung. Selanjutnya, karena kunci enkripsi $e$ diumumkan (tidak rahasia), maka kunci dekripsi $d$ dapat dihitung dari persamaan $ed \equiv 1 \pmod{\phi(n)}$.

- Penemu algoritma *RSA* menyarankan nilai *a* dan *b* panjangnya lebih dari 100 digit. Dengan demikian hasil kali $n = a \times b$ akan berukuran lebih dari 200 digit.

- Usaha untuk mencari faktor bilangan 200 digit membutuhkan waktu komputasi selama 4 milyar tahun! (dengan asumsi bahwa algoritma pemfaktoran yang digunakan adalah algoritma yang tercepat saat ini dan komputer yang dipakai mempunyai kecepatan 1 milidetik).

# Contoh RSA 512 bit

(dikutip dari Sarwono Sutikno, EL)

- Modulus *n* = 81 5a d0 b9 0a ac 9f 4c da cc 57 6e ca a7 6a c3 46 92 a7 81 68 ec 08 ec 77 dd 40 c2 ec 97 52 cb 3b 34 2c b6 a6 e2 76 3a ed 42 84 fa 55 ac 0d 6c 10 39 a2 7e a3 09 be 40 35 38 04 7d 06 43 1f 6f
- *e* = 29 40 70 02 50 db 19 6b b1 f4 8a a7 b4 59 6c 4b 66 b5 94 f6 15 ae e4 69 44 95 23 f3 d0 fc ea 84 19 7c 55 e0 27 40 2d 19 18 15 08 05 51 ac f5 98 91 f0 98 5f c4 17 05 eb 3b e8 a3 04 32 d4 20 2f
- *d* = 59 f1 2f 29 73 d0 bc 8e 13 6e 2a 21 53 2c b7 4d 69 82 c9 54 92 6c 64 43 0d 69 15 83 e9 44 a6 de 5e 30 e9 ae 48 f9 c8 84 a4 16 44 4d df 50 f2 0e 96 3e 24 df a4 f4 ec 3d c6 db 61 a7 e6 dc ea cf

- Secara umum dapat disimpulkan bahwa RSA hanya aman jika *n* cukup besar.
- Jika panjang *n* hanya 256 bit atau kurang, ia dapat difaktorkan dalam beberapa jam saja dengan sebuah komputer *PC* dan program yang tersedia secara bebas.
- Jika panjang *n* 512 bit atau kurang, ia dapat difaktorkan dengan beberapa ratus komputer [WIK06]

- Tahun 1977, 3 orang penemu *RSA* membuat sayembara untuk memecahkan cipherteks dengan menggunakan RSA di majalah *Scientific American*.
- Hadiahnya: $100
- Tahun 1994, kelompok yang bekerja dengan kolaborasi internet berhasil memecahkan cipherteks hanya dalam waktu 8 bulan.

## Kelemahan RSA

- *RSA* lebih lambat daripada algoritma kriptografi kunci-simetri seperti *DES* dan *AES*
- Dalam praktek, *RSA* tidak digunakan untuk mengenkripsi pesan, tetapi mengenkripsi kunci simetri (kunci sesi) dengan kunci publik penerima pesan.
- Pesan dienkripsi dengan algoritma simetri seperti DES atau AES.
- Pesan dan kunci rahasia dikirim bersamaan.
- Penerima mendekripsi kunci simetri dengan kunci privatnya, lalu mendekripsi pesan dengan kunci simetri tersebut.