CRYPTOGRAPHY

PEMBANGKIT BILANGAN ACAK & FUNGSI HASH

CAPAIAN PEMBELAJARAN

- Mahasiswa memahami metode-metode pembangkit bilangan acak
- Mahasiswa memahami konsep hash dalam menyederhanakan ukuran enkripsi pesan

Agenda.

- Pembangkit Bilangan Acak
- Fungsi Hash

4

PEMBANGKIT BILANGAN ACAK

- **Bilangan acak:** Sebuah bilangan yang kemunculannya untuk kali kedua dan seterusnya tidak pernah sama. Contoh pin OTP, captcha code dll.
- Bilangan acak terdiri dari:
  - **Bilangan acak sejati (*true random number*):** Bilangan acak yang kemunculannya untuk kali kedua tidak akan bisa diprediksi.
  - **Bilangan semi-acak (*pseudorandom number*):**
    - Bilangan acak yang kemunculannya dapat diprediksi
    - Merupakan hasil sebuah prosedur (algoritma) komputasi pembangkitan dari satu atau lebih parameter yang disebut seed (umpan) dna berlaku sebagai *key*.
    - Bilangan ini menggunakan sebuah kunci untuk dapat menentukan hasil bilangan acak.

5

PEMBANGKIT BILANGAN ACAK

Linier Congruential Generator (LCG)

- Merupakan salah satu pembangkit bilangan acak konvensional
- LCG didefinisikan dalam relasi:

$$X_{i+1} = (aXi + b) \bmod m$$

Dimana:

X_{i+1} = bilangan acak ke-i
X_i = bilangan acak sebelumnya
m = modulus (> 0)
a = faktor pengali $(0 \leq a \quad m)$
b = increment $(0 \leq b \quad m)$

} *konstanta*

6

PEMBANGKIT BILANGAN ACAK

- Xo dibutuhkan untuk mengawali pembangkitan bilangan acak yang disebut sebagai umpan (*seed*)

  Contoh:

  Misalkan a = 7, b = 11, dan m = 17, maka LCG sebagai berikut:

$$X_{i+1} = (aXi + b) \bmod m$$

  Hasilnya adalah sebagai berikut.

$X_{i+1} = (7.0 + 11) \bmod 17$, $X_0 = 0$;

7

PEMBANGKIT BILANGAN ACAK

Selengkapnya.

i	X_i	i	X_i	i	X_i	i	X_i	i	X_i
0	0	**5**	7	**10**	16	**15**	13	**20**	14
1	11	**6**	9	**11**	4	**16**	0	**21**	7
2	3	**7**	6	**12**	5	**17**	11	**22**	9
3	15	**8**	2	**13**	12	**18**	3	**23**	6
4	14	**9**	8	**14**	10	**19**	15	**24**	2

8

PEMBANGKIT BILANGAN ACAK

- Keunggulan LCG terletak pada kecepatannya karena hanya membutuhkan sedikit operasi aritmatika
- Kekurangannya adalah LCG tidak aman, karena bilangan acaknya dapat diprediksi urutan kemunculannya.
- Tidak digunakan dalam sistem kriptografi.
- LCG lebih digunakan untuk simulasi seperti memperlihatkan sifat statistic dan sangat tepat untuk uji empirik.

9

PEMBANGKIT BILANGAN ACAK KRIPTOGRAFI

- Aman dan tidak dapat diprediksi kemunculannya
- Pembangkit bilangan acak yang menghasilkan bilangan yang tidak dapat diprediksi disebut dengan *cryptographically secure pseudorandom generator* (CSPRING).
- Sebuah CSPRING harus memenuhi 2 syarat.
  1. Lolos uji keacakan secara hitung statistik
  2. Tahan terhadap serangan (attack) yang dimaksudkan untuk memprediksi bilangan acak yang dihasilkan.

10

PERANCANGAN CSPRING

CSPRING Berbasis Teori Bilangan.

Blum-Blum-Shub

- Merupakan CSPRING yang paling sederhana dan barhasil guna
- Dikembangkan pada tahun 1986 oleh Lenore **Blum**, Manuel **Blum**, dan Michael **Shub**.

- Algoritma **Blum-Blum-Shub**.
  1. Pilih 2 buah bilangan prima (rahasia), p dan q yang besar.
  2. Kalikan keduanya menjadi $n = p.q$ (**bilangan bulat blum**)
  3. Pilih sebuah bilangan acak, s, sebagai *seed* dengan aturan:
     - $2 \leq s < n)$
     - s dan n relatif prima, kemudian hitung $x_o = s^2 \, mod \, n$.

11

PERANCANGAN CSPRING

4. Barisan bit acak dihasilkan dengan melakukan iterasi berikut sepanjang yang diinginkan:
   - Hitung $x_i = xi\_1^2 \, mod \, n$
   - z_i = bit LSB dari x_i

Contoh: Misalkan p = 11, q = 23, $n = p.q$ = 11 . 23

12

PERANCANGAN CSPRING

Algoritma RSA

Algoritma.

1. Pilih 2 buah bilangan prima (rahasia), p dan q yang besar.
2. Kalikan keduanya menjadi $n = p.q$
3. Pilih sebuah bilangan acak, s, sebagai *seed* dengan aturan:
   - $2 \leq s \leq n)$
   - s dan n relatif prima, kemudian hitung $x_o = s^2 \, mod \, n$.
     Relatif prima adalah s dan n memiliki FPB yang sama.
4. Barisan bit acak dihasilkan dengan melakukan iterasi berikut sepanjang yang diinginkan:
   - Hitung $x_i = x_{i\_1}^2 \, mod \, n$ dengan $x_o = s$
   - z_i = bit LSB dari x_i

13

PEMBANGKIT BILANGAN ACAK & FUNGSI HASH