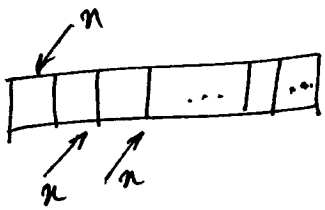
lecture 14

Probabilistic public-key Encryption

Rand. Alg. → you have "x" & want to see if "x" exists in an array

(A) size & this array is "n"

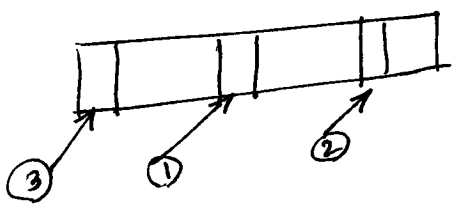


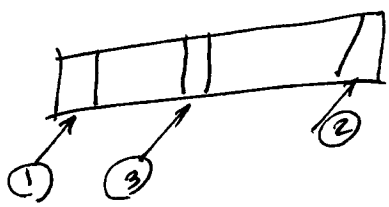
$O(n)$

(A') → optimize by binary search $O(\log n)$ (sorted array)

(B) las vegas Alg.

Repeate
  randomly select an elemet from array
Until "x" is found

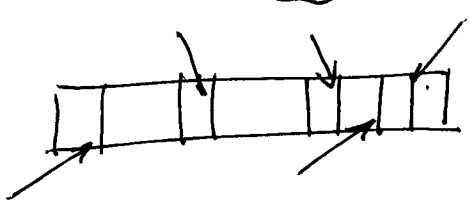




(C) Monte Carlo Alg.

i = 1
repeate
  randomly select an elemet from array
  i = i + 1
until i = k

k = 5



Basic def security: a minimal security requirement of an Enc scheme is that it must be difficult, essentially in all cases, for a passive adv. to recover plaintext from the corresponding ciphertext.

Deterministic Enc schemes → RSA, Rabin

under a fixed public-key, a paricular plaintext "m" is always encrypted to the same ciphertext "c".

→ It's easy to detect when the same message is sent twice.

→ this problem is going to be resolved in pr.b. encryption schemes as they utilize randomness ...

Note quadratic residue mod "n":

$n=10$ $(1\sim9)$ $\mathbb{Z}_{10}$

$1^2 \overset{10}{\equiv} 1$ $\quad 2^2 \overset{10}{\equiv} 4$ $\quad 3^2 \overset{10}{\equiv} 9$ $\quad 4^2 \overset{10}{\equiv} 6$ $\quad 5^2 \overset{10}{\equiv} 5$

$6^2 \overset{10}{\equiv} 6$ $\quad 7^2 \overset{10}{\equiv} 9$ $\quad 8^2 \overset{10}{\equiv} 4$ $\quad 9^2 \overset{10}{\equiv} 1$

1, 4, 5, 6, 9 → are (quadratic residue mod 10)

Blum - Goldwasser probabilistic Enc Scheme

Key gen

1. Select two large random (& distinct) primes p & q such that they congruent to 3 modulo 4.

$$p \equiv 3 \pmod 4 \qquad q \equiv 3 \pmod 4$$

2. Compute $n = pq$ ← factoring problem

3. use EEAlg to compute "a" & "b" such that

$$ap + bq = 1$$

4. public key is "n" & private key is (p, q, a, b)

this is not a randomized behaviour

Enc: (a) obtain A's Authentic public key "n"

(b) let $k = \lfloor \log_2 n \rfloor$ and $h = \lfloor \log_2 k \rfloor$. Represent "m" as a string $m_1 m_2 m_3 \cdots m_t$ of length "t", where each m_i is a binary string of size "h"

n=16
$\log_2 16 = 4$
0 ~ 15

(c) select seed x_0 (random quadratic residue mod "n")

$$r \in \mathbb{Z}_n^* \longrightarrow x_0 = r^2 \pmod n$$

randomized behaviour

(d) $1 \rightsquigarrow t$: d-1 $x_i = x_{i-1}^2 \pmod n$ d-2 p_i be the "h" least significant bits of x_i

d-3 $c_i = p_i \underset{\text{XOR}}{\oplus} m_i$

Dec

(e) $x_{t+1} = x_t^2 \pmod n$

(f) ciphertext : $c = (c_1, c_2, \ldots, c_t, x_{t+1})$

(a) $d_1 = \left(\frac{p+1}{4}\right)^{t+1} \bmod (p-1)$ (b) $d_2 = \left(\frac{q+1}{4}\right)^{t+1} \bmod (q-1)$

(c) $u = x_{t+1}^{d_1} \pmod p$ (d) $v = x_{t+1}^{d_2} \pmod q$

(e) $x_0 = vap + ubq \pmod n$

(f) $1 \rightsquigarrow t$

f-1 Compute $x_i = x_{i-1}^2 \pmod n$

f-2 p_i to be "h" least significant bits of x_i

f-3 Compute $m_i = p_i \oplus c_i$ ← XOR

Example of Blum-Goldwasser

Key gen

$p = 499 \equiv 3 \pmod 4$

$q = 547 \equiv 3 \pmod 4$

$n = p \times q = 272953$

EE algo $\longrightarrow$ $(-57)\underbrace{499}_{p} + \underbrace{(52)}_{b}\underbrace{547}_{q} = 1$ ($a = -57$)

$k = \lfloor \log_2^{n=272953} \rfloor = 18 \longrightarrow h = \lfloor \log_2^{k=18} \rfloor = 4$

$2^4 = 16$ (18) $2^5 = 32$

Enc

20 bit

$m_1\ m_2\ m_3\ m_4\ m_5$ $\xrightarrow{t=5}$ $5 \times 4 = 20$ bits

$m_1 = 1001$ $m_2 = 1100$ $m_3 = 0001$ $m_4 = 0000$ $m_5 = 1100$

Randomized selection $\longrightarrow$ $x_0 = 399^2 \pmod n = 159201$

i	$x_i = x_{i-1}^2 \pmod n$	p_i	$c_i = p_i \oplus m_i$
1	180539	1011	0010
2	193932	1100	0000
3	245613	1101	1100
4	130286	1110	1110
5	40632	1000	0100

$x_6 = x_5^2 \bmod n = 139680 \longrightarrow c = (0010, 0000, 1100, 1110, 0100, x_6)$

Dec

x^b

$d_1 = ((p+1)/4)^6 \pmod{p-1} \equiv 463$

$d_2 = ((q+1)/4)^6 \pmod{q-1} \equiv 337$

$u = x_6^{463} \bmod p = 20$ $\quad v = x_6^{337} \bmod q \equiv 24$

$x_0 = vap + ubq \pmod n = 159201$

$x_i, p_i \longrightarrow m_i = p_i \oplus c_i$ (XOR)