Tutorial XI

14 December 2020 13:38

TWO PARTY CRYPTOGRAPHY (part 2)

→ Coin tossing

→ Rabin OT

↳ Rabin-OT ⟺ 1-outof-2-OT

→ Example of practical use of 1-out-of-n OT

Coin tossing (slide 9)

Element: two large primes p and q, $n = p \cdot q$

Alice keeps private p and q and reveals n

1.) Bob chooses $x_B \leq \frac{n}{2}$ and sends $y = x^2 \bmod n$ to Alice

2.) Alice calculates $x_i \in \{x_1, x_2, x_3, x_4 \mid x_i^2 = y \bmod n\}$

$$\underbrace{x_1 < x_2}_{\leq n/2} < x_3 < x_4 \qquad x \qquad n-x$$

3.) Alice chooses $x_A \in \{x_1, x_2\}$ at random and discloses to Bob the least significant bit in which

x_1 and x_2 differ. ←

4.) Alice and Bob reveal all their information $A = \{x_1, x_2\}$ $B = \{x_B\}$. They use it to verify whether Alice guessed x correctly.

5.) if Alice guessed correcty outcome is 1 (heads)
if Bob guessed correctly outcome is 0 (tails)

Assume in step 3 Alice discloses x_A in full.

→ $x_A = x_B$ Bob can't cheat
(he cannot calculate the other square root)

→ $x_A \neq x_B$ Bob can cheat

He can say she guessed correctly
and in step 4 reveal $x_B = x_A$

He can say she didn't guess correctly
and in step 4 reveal $x_B \neq x_A$

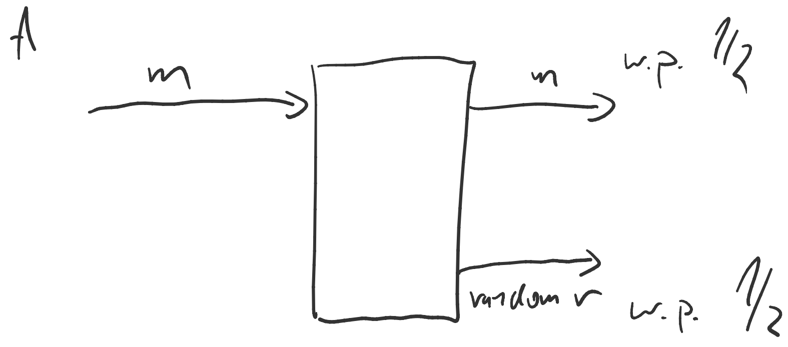
Revealing only LSB of Alice's choice can be used to verify her guess in step 4, without revealing x_A.

Security against cheating Alice – IT

Security against cheating Bob – computational

Security against cheating Bob - computational

Rabin-OT



→ Alice doesn't learn which option happened

→ Bob knows whether he got r or m.

1.) Alice chooses large primes p and q and sends $n = p \cdot q$ to Bob

2.) Bob chooses x and sends $y = x^2 \bmod n$ to Alice

3.) Alice calculates $\{x_1, x_2, x_3, x_4 \mid x_i^2 = y\}$, chooses one at random and sends it to Bob.

1.) Information Alice is sending is the factorization of n (p, q)

2.) After step 2 Bob knows two of four square roots of y $(x, n-x)$ and he doesn't know the remaining 2.

3.) After step 3 Bob know all the roots w.p. $1/2$ and thus factors p, q. And w.p. $1/2$ he doesn't learn anything new

4.) Alice doesn't know if she disclosed the factors

5.) With RSA : Alice calculates $m^e \bmod n$

6.) if Bob learns how to factor n, he can calculate $e^{-1} \bmod (p-1)(q-1)$ and recover m.

$\Leftarrow$

Rabin $\Leftrightarrow$ 1-out-of 2

Rabin $\Rightarrow$ 1-out-of-2

1.) Alice sends $\boxed{3n}$ randomly chosen bit messages $(x_1, \dots, x_{3n})$ to Bob using Rabin OT

$x_1\, x_2\, x_3\, x_4 \dots \quad x_3$

$\downarrow$

$x_1\, ?\, ?\, x_4$

2.) Bob chooses n indices of the messages he received I and n indices of the messages he did not receive J

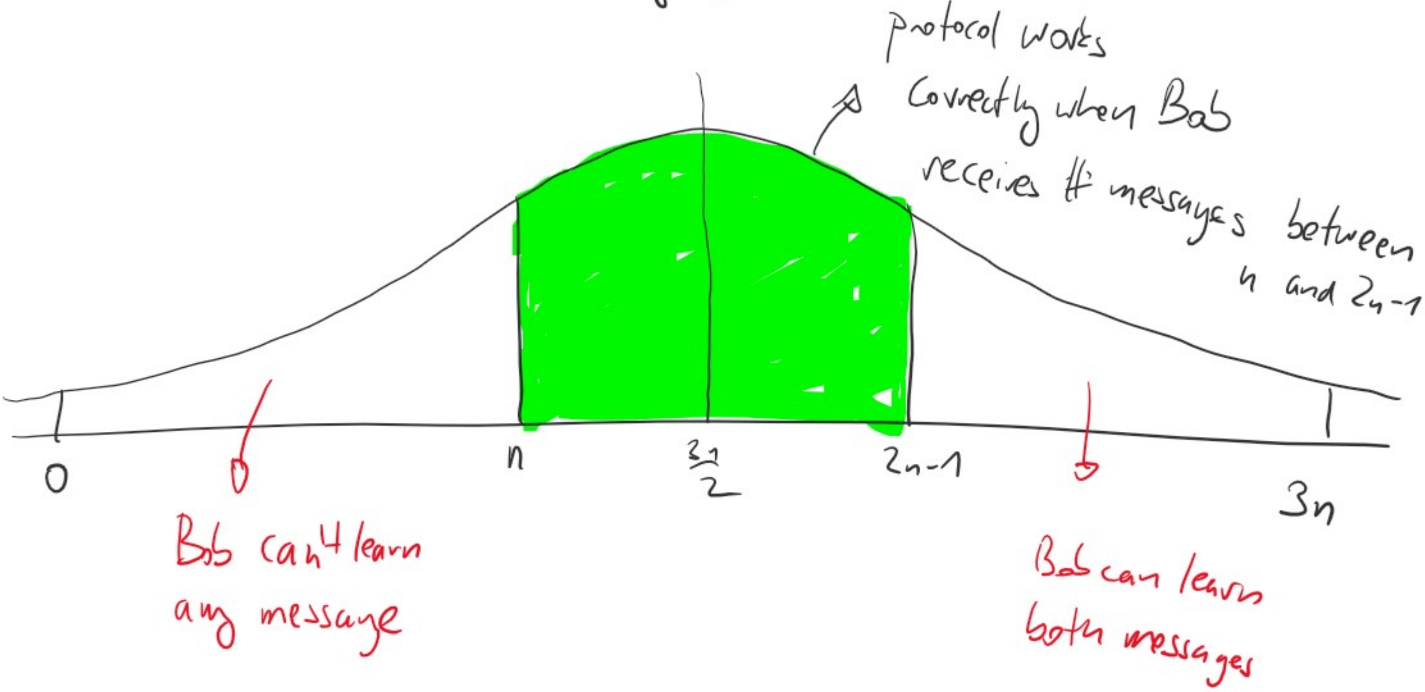
$\downarrow$

3.) Bob sends (I, J) if he wants to learn m_1

(J, I) if he wants to learn m_2

4.) Alice receives (S_1, S_2) and sends

$m_1 \bigoplus_{i \oplus S_1} x_i$ and $m_2 \bigoplus_{j \in S_2} x_j$


protocol works correctly when Bob receives # messages between n and 2n-1
0
n
$\frac{3n}{2}$
2n-1
3n
Bob can't learn any message
Bob can learn both messages

Pr that Bob receives $< n$ or $> 2n-1$ decreases exponentially with security parameter n (Chernoff tail inequalities)

Example of interesting use of 1-out-of n OT

Scenario: Alice has an online shop in customers can buy vouchers which can be used to pay for services

Requirements: 1.) hard to forge

2.) Anonimity (Alice cannot match voucher to a person

2.) Anonimity (Alice cannot match voucher to a person she sold it to)

1.) Alice creates a message

x = " Voucher for 100 Kč "

And the voucher is (x, s) where s is Alice's signature.

PRoblem: Vouchers can be copied!

2.) Alice creates a voucher (id is a counter)

x_i = " Voucher for 100 Kč, id: i "

Voucher (x_i, s_i) s_i is a signature of x_i

Problem: id can be to it's buyer.

3.) 1-of-n OT

Alice creates a large database of vouchers $(x_1, s_1), \dots, (x_n, s_n)$

If someone buys a voucher Alice sends it via 1-out-of-n OT

Later if the voucher is used, it is removed from the database.

PROBLEM: Alice can sell the same voucher twice.