?OIS Evaluation 5

Rishabh Singhal
20171213

There are n channels to send k blocks of data between A and B.

So, we can use same way as in evaluation 2, by storing k blocks of data as coefficient of a polynomial and generating n points at random values ~~for~~ of this polynomial.

$$\text{So, } P(x) = \sum_{i=0}^{k-1} d_i x^i$$

and points $\Rightarrow$ $(1, P(1)), (2, P(2)), \ldots, (n, P(n))$

and we only need k points out of these n points to reconstruct the polynomial and have k data points.

Also, to check if the recieved point is "un-corrupted" rather than sending points as it is, ~~@~~ points with their digital signatures are sent.

If, after recieving, digital signatures are correct the points are verified and are not corrupted.

And minimum no. of non-corrupted blocks required are k to reconstruct polynomial

hence

$$e \geq n-k \quad \text{or} \quad \boxed{k \geq n-e}$$

Elgamal Encryption

A ⟷ B

1. Generation of public key & private key

a large prime "p" is generated
& a generator "g" for $\mathbb{Z}_p$

then, a random number [x] is chosen from $\mathbb{Z}_p^*$
which is the private key

& public key = $\{g^x \bmod p, g, p\}$

2. Enc Scheme

let m = message,

choose random $y \in \mathbb{Z}_p^*$

$z = g^y$ & $s = h^y$ & $c = (ms) \bmod p$

then ~~Signature~~ Encrypted value = $\{z, c\}$

3. Decryption] Let $s = z^x = g^{xy}$

then, $m = \boxed{c \times s^{-1}} = m \times s \times s^{-1} = (m)$

Oblivious Transfer Protocol

let A have index $i \in \{0, 1, \ldots, n-1\}$

and B is the one having $\{b_0, b_1, \ldots, b_{n-1}\}$

and B wants to send values to A, such that A recieves value of one of (b_i) while B not knowing what's the value of i that A is recieving.

Protocol :

- A generates random array $\Rightarrow R = \{r_0, r_1, \ldots, r_i, \ldots r_{n-1}\}$

then as it wants to acquire value of b_i at i^{th} index it encrypts it. so,

$$R = \{r_0, r_1, \ldots, r_n\ Enc(r_i), \ldots, r_{n-1}\}$$

& then $A \xrightarrow{R} B$

- B recieves R then decrypts all values of R

i.e. $X = \{dec(r_0), dec(r_1) \ldots, dec(Enc(r_i)), \ldots dec\ r_{n-1}\}$

$\Rightarrow X = \{dec(r_0), dec(r_1) \ldots, r_i, \ldots, dec\{r_{n-1}\}\}$

and takes XOR of i^{th} value with b_i

& $X' = \{ b_0 \oplus dec(r_0), b_1 \oplus dec(r_1), \ldots, b_i \oplus r_i, \ldots, b_{n-1} \oplus dec(r_{n-1}) \}$

and sends back to A.

☆ A then takes the i^{th} value i.e. $b_i \oplus r_i$ and takes XOR with r_i,

i.e. $\Rightarrow$ $b_i \oplus r_i \oplus r_i \Rightarrow b_i$

hence obtains (b_i) without B knowing the index (i).

Now for given problem :-

(1) The public keys of A & B are known to all

For this, it is same as using previous case with n points and k blocks.

& $\boxed{c \geq n-k}$ as proved earlier.

because public keys are known to all.

(2) Public key of B is known to A but not otherwise :-

As far as sending n block is concerned in this case,
first A sends Array B with Encrypting (ith index) with public key of B (which is known before hand) so, no problem here

But, when it recieves the array of n points, B needs to verify digital signatures and the points for that, B must know the public key of A to verify.

So for this standard coding theory is required to transfer k blocks in a fault tollerent way through n different channels

$$\text{so} \Rightarrow \boxed{n \geq k + 2e'} \Rightarrow e' \leq \frac{n-k}{2}$$

Also, let's say public key of A can be represented as 's' blocks, then to send it,

$$n \geq 2e' + s \Rightarrow \boxed{e' \leq \frac{n-s}{2}} \quad \text{---(*)}$$

And after sending public key, normal OT transfer can be used

which is $\boxed{e \leq n-k}$ --- (**)

from (*) & (**) $\boxed{e \leq \min\left((n-k), \frac{(n-s)}{2}\right)}$

(iii) Public key of B is known to A, not otherwise.

Here, again first public key of A is sent to B via normal coding theory protocol in a fault tolerent way, which gives ⇒ $e \leq \frac{n-s}{2}$

where s ≥ no. of block for public key of A

And also, after then normal DT can take place for which,

$$e \geq n-k$$

hence ⇒ $e \leq \min\left(n-k, \frac{n-s}{2}\right)$

(iv) Neither party knows the public-key of ~~each~~ other:

Here, all the public keys will be sent to the other party,

this will again take place as

$e \leq \left(\frac{n-s}{2}\right)$ where s = no. of blocks required to represent A's public key

and $e \leq \left(\frac{n-m}{2}\right)$ m = no. of blocks required to represent public key of B

also, after that normal OT transfer can take place for which

$$e \leq (n-k)$$

hence $\Rightarrow$ $e \leq \min\left[n-k, \frac{n-m}{2}, \frac{n-s}{2}\right]$

☆ considering $(m=s=1)$,

gives $\boxed{e \leq \min\left[n-k, \frac{n-1}{2}\right]}$ ☆