Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 6, June 2015, pg.937 – 944

RESEARCH ARTICLE

Post Processing of Symmetric Key in Quantum Key Distribution with Limitations of Key Access

Mr. Hemanth Kumar A R[1], Mrs. Jaya R[2]

Student, M.Tech (Computer Science and Engineering) New Horizon College of Engineering, Bangalore, India[1]

Sr. Assistant professor, Computer Science and Engineering department, New Horizon college of Engineering, Bangalore, India[2]

hemanth.arrhp@gmail.com, jayamanojkumar@gmail.com

Abstract: In this paper we are concerned of limiting the key access to the receiver from sender by Private Query, and privacy between the two end users.

For example: consider Bob has a database with many items and Alice wants to buy an item in Bob's database, the aim of limiting key access to Alice is to ensure that Alice can get only one item from Bob's database .

Here we do the post processing of symmetric key, as a practical model, State Transfer. Which is used to distribute key from Bob to Alice with limited access, and then applies the symmetric key to achieve aim by using Private Query

We also do the error correction method for the symmetric key

Keywords: post processing, quantum key distribution, symmetric key, privacy, error correction

I. INTRODUCTION

Communication security is very important, we use communication security in our daily life knowingly or unknowingly.

We use communication security every day like

Paying by credit cards,

SMS, cell phone conversation

Email, chats, online calls

Secure browsing

Communication online

Cloud communication and storage between your devices

Software updates on the laptop and mobile phone

Online banking etc.

"Note: 'User A' (Bob): Sender/Data owner

'User B' (Alice): Receiver/One who fetch the data"

As we know security of most classical cryptosystem is based on the assumptions of computation complexity and they may be broken by the strong ability of some advanced algorithms like quantum computations [1], [2], fortunately this difficulty can be-overcome by quantum-cryptography [3], [4], where the security is assured by physical principles, owing to its higher level of security, quantum cryptography has attracted a great deal of attention now

In the existing system the chances of compromising privacies are there, and implementing in a large database may be difficult since the dimensions in the oracle is high

Giovannetti et al [6]-[8] has been presented the first quantum private query in 2008 in which the database will be indicated in unary operators, this quantum private query will be done in two query states.

Olejnik [9] proposed the improved quantum private query than previous in 2011 which needed only 1 query state.

Compared to classical symmetric private information retrieval the above 2 protocols may have the advantage in security like fundamental physical principal, but not much successful on assumptions on computational complexity, runtime computational complexity and communicational complexity.

II. PREVIOUS METHODS

Previous methods demands the following.

D1: 'User A' will know every bit in the key.

D2: 'User B' knows every bit with certain probability p (for example p = 0.34 for an honest 'User B')

D3: 'User A' doesn't know which key bits are known by 'User B'

A. KN→N Method

For our convenience, the kN-bit raw key O^R can be denoted as $O_1^R\ O_2^R\ \dots\dots O_{kN}^R$, and N-bit final key O^F can be denoted as $O_1^F\ O_2^F\ \dots\dots\dots O_N^F$.

Here every $O_i^R(1 \le i \le kN)\ or\ O_i^F(1 \le i \le N)$ represents a key bit, in this method the relation between O^R and O^F is as below

$$O_i^F = \bigoplus_{j=0}^{k-1} O_{i+jN,}^R 1 \leq i \leq N \qquad \text{…………… (a)}$$

Where ⊕ is the addition modulo 2.

For example: the raw keys for 'User A' is

0110, 0100, 0111

0011, 0101, 1001

While for 'User B' it will be as below

?1??, 0???, ?1??

0???, ?1??, ?0??

i.e. 'User B' will be knowing only 2nd, 5th, 10th, 13th, 18th, and 22nd bit in the key.

Then after the dilution, the final key will be as below for 'User A'

0101, 0001, 1110

But for 'User B' it will be as below

????, ????, ?0??

For 'user B' it's easy to see that the number of known bits for 'User B' is 1 i.e. known bits is reduced from 6 to 1.

B. The N→N Method

Same as above, the N-bit raw key O^R can be denoted as $O_1^R\ O_2^R\ \dots \dots .\ O_{kN}^R$, and N-bit final key O^F can be denoted as $O_1^F\ O_2^F\ \dots \dots \dots .\ O_N^F$.

Here in N→N Method the relation between O^R and O^F is as below

$$O_i^F = \bigoplus_{j=1}^{i+k-1} O_{j,}^R 1 \leq i \leq N \qquad \text{…………… (b)}$$

Where ⊕ is the addition modulo 2.

For example: N=12, k=2,

The raw key for 'User A' is

0110, 0100, 0111

But for 'User B' it will be

???0, 0?0?, ????

i.e. 'User B' knows only 4th, 5th, and 7th key bit only.

Then after the dilution, the final key for 'User A' will be as below

1010, 1100, 1001

But for 'User B' it will be as below

???0, ????, ????

Now it's easy to see that number of known key bits is reduced from 3 to 1.

C. The rM→N Method

Here in this method, rM- bit raw key O^R can be divided into r sub-key with the same length of M,

Then the sub-key extensions and the shift addition can be done as shown below

$$O_j^F = \bigoplus_{i=1}^{r} O_{j+_{si},}^{\sim R_i} 1 \leq i \leq N \qquad \ldots\ldots\ldots\ldots.(c)$$

The sub-key extensions tries to reuse every bit to the most degree so that M reaches lowest value for each sub-key in the communication complexity.

The shift addition is used to reduce 'User B' knowledge in the final key.

III. DISADVANTAGE IN PREVIOUS METHODS

In all previous methods, the parity of raw key bits is final key bit

There are chances that 'User B' can obtain whole database by many queries, at most N, by identifying almost known sets.

In all previous methods, we have only seen how number of known key bits are affected for 'User B', i.e. 'User A' privacy

But here none of previous methods are concerned about its influence on 'User A' data, about the position of 'User B' i.e. 'User B' privacy.

As we have seen in the demand of previous methods, 'User A' won't be knowing which bits are known by 'User B'

A. Disadvantage of N→N Method

Here in this method, 'User B' know every bit with probability p=0.34, there will be no problem only if 'User B' is honest. Else if the 'User B' is dishonest the value of probability may be changed. For example: 'User B' may do a quantum memory and execute an unambiguous state to attack,

If in case almost known sets of 'User A' are known to 'User B' then there are chances of 'User B' exploit 'User A' data.

'User B' can easily obtain at least part of the data in the database, even if not the complete database,

If the 'User B' intention is to exploit part of data from 'User A' database then 'User A' privacy will be compromised, 'User A' data can be misused.

B. Disadvantage of rM→N Method

Here in this rM→N Method, 'User A' and 'User B' generate an N bit final key by using rM raw key bits, But in this method ‘User B’ can obtain almost all data in the ‘User A’ database. By doing at most rM queries.

For Example: ‘User B’ can honestly execute first query, then ‘User B’ may calculate the basis of final key of ‘User A’ and ‘User B’ can do many queries less than rM, and there are chances of getting whole database is high.

IV. PROPOSED SYSTEM

The proposed post processing of symmetric key in quantum key distribution with limited key access, targets the privacy in the users in the communication between them.

Here in the proposed system focused on the data privacy of the data owner, and limit the access of data by database owner to others who try to fetch his data,

For example: if ‘User A’ is data owner and ‘User B’ need to fetch a data form ‘User A’,

‘User A’ need to limit the access of data from ‘User B’, which assures ‘User A’ privacy

‘User A’ should not know which data was fetched by ‘User B’, which assures ‘User B’ privacy.

Proposed system also focus on the error correction method

So the main objective are, ensure both users privacy in a communication. ‘User B’ should not be able to get more than one item. Symmetric key in quantum key distribution become more practical, Error correction scheme is effective.

We can obtain this using following stages:

S1: State Transfer (Quantum): here we do distribution of quantum key with the limitation of key access. Where ‘User A’ will know every bit in the key, but ‘User B’ knows only limited keys with certain probability.

S2; Private Query: if ‘User B’ knows the j^{th} bit in the key shared by ‘User A’ and need to fetch the i^{th} item in the ‘User A’ database, ‘User B’ declares a shift value s=j-i so that the ‘User A’ shift his final key by s. and at the end ‘User A’ encrypts his/hers database by shifted final key and send the entire database to ‘User B’, generally it is assumed that the content of each item in the database is a bit, and one key bit can encrypt one item by one-time pad. Thus ‘User B’ can correctly decrypt the item he/she needed by known key bit, and more importantly both users privacy are successfully protected.

S3: post processing: post processing converts the initial/raw key into the final key, and also ensures the privacy to both users, ensures privacy to ‘User A’ by greatly reducing eavesdropping information owing to the compression of the key. Ensures privacy to ‘User B’ by not allowing the ‘User A’ to know which item was fetched from ‘User A’

V. COMMUNICATION BETWEEN TWO USERS

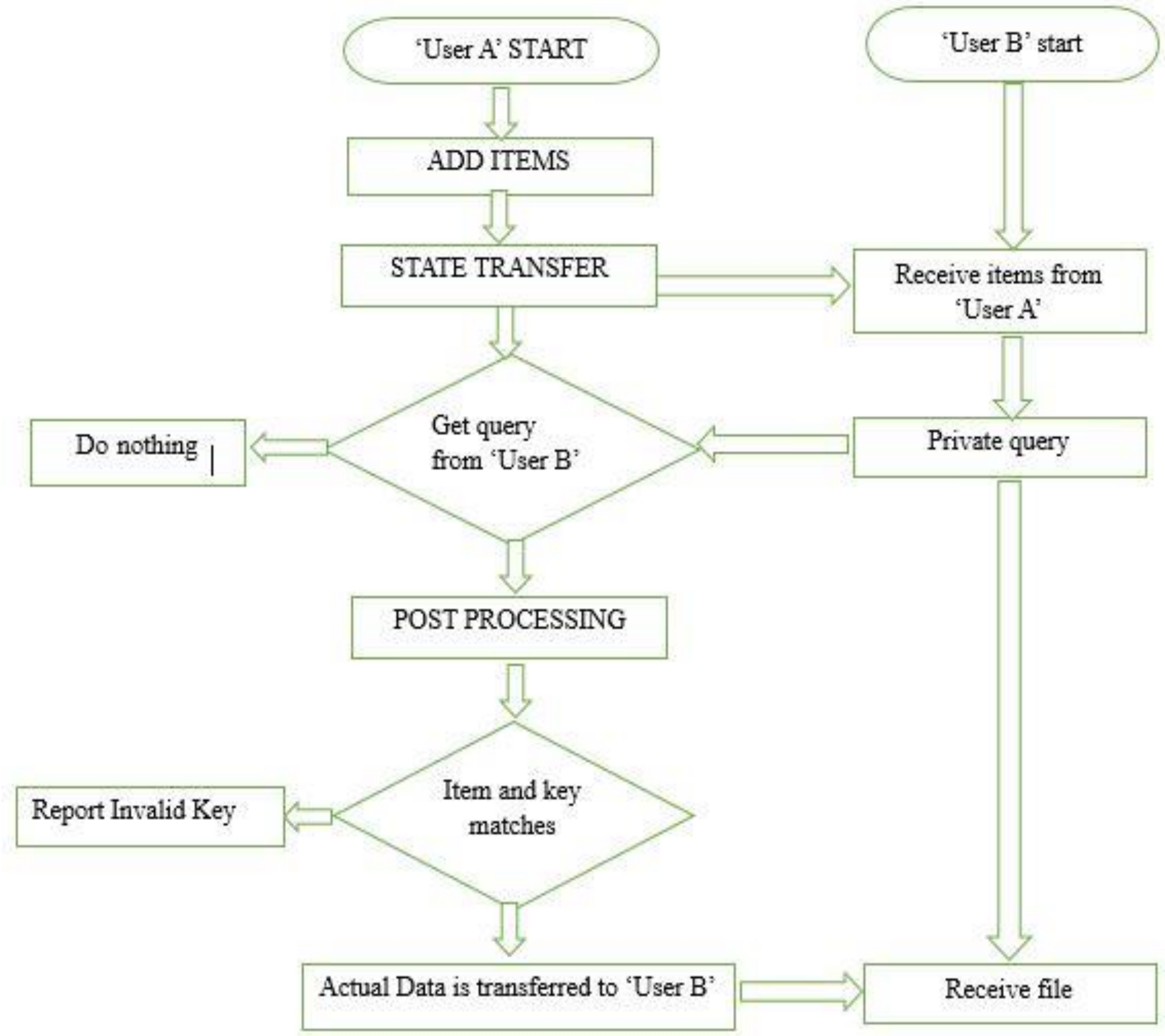

Fig 1. UML for communication between 'User A' and 'User B'

As we have considered 'User A' as the data owner, who has his/her database and have sensitive data in his/her data.

First the 'User A' need to have data in his database, to do that he uses ADD ITEMS

STATE TRASFER is used to send the complete data to 'User B' which will be discussed in implementation.

Since 'User B' job is to just fetch at most 1 data from the 'User A',

'User B' task is to do private query from the information he/she has and fetch the data from 'User A'.

For example: now let us consider 'User A' has some sensitive data in his database,

To provide privacy to 'User A', he/she do STATE TRANSFER, which is used to send the complete encrypted data of 'User A' to 'User B' with the limitation of key access to 'User B'

After receiving the key and items from 'User A', 'User B' do the private query which ensures that exact data needed by 'User B' and by which 'User A' can shift his final key.

POST PROCESSING is done by 'User A' where the key distribution information greatly decreases and owing to the compressing of the key. And raw or initial key into the final key

On POST PROCESSING done by 'User A' if the private query done by is correct, i.e. item matches the key shared then the data can be received from the 'User B',

If the private query is not proper i.e. if item does not match with the key shared between both users, we should have something to report the 'User A' so that error correction should be maintained.

There are chances that if 'user B' pays money for the data what she retrieve to the 'User A' but after private query she may not get the actual data, may get wrong data or may not get any data at all. This will lead to a bad reputation of 'User A' a database owner.

Practically till now in previous methods as indicated in previous published papers there can be errors in the shared key between the two users.

Errors correction is very much necessary to avoid the bad reputation of 'User A' and to do the perfect transaction of the data.

To avoid this we have done an error correction method in post processing of symmetric key in quantum key distribution, by reporting the invalid key to 'User A' so that he/she will come to know that transaction is not successful, but here again 'User A' won't be knowing which data 'User B' tried to fetch, this is to maintain privacy to 'User B'

By this we provide privacy to both users and error correction method will be effective.

VI. CONCLUSION

Quantum cryptography has drawn much attention of the scholars in the world, because of the success in the high security in key distribution, every user's hope the security of various kinds of protocols can be overall upgraded by quantum manner, to this aim different kinds of quantum protocols have been proposed, however post processing of symmetric key in quantum key distribution with limitation of key access will ensures the following

- Ensure both users privacy in a communication. '
- User B' will not be able to get more than one item.
- Symmetric key in quantum key distribution become more practical,
- Error correction scheme is effective.

REFERENCES

[1] P. W. Shor, "Algorithms for quantum computation: Discrete algorithms and factoring," in Proc. 35th Annu. Symp. Foundations Comput. Sci., Santa Fe, New Mexico, Nov. 1994, pp. 124–134.

[2] L. K. Grover, "A fast quantum mechanical algorithm for database search," in Proc. 28th Annu. ACM Symp. Theory Comput., New York, NY, USA, May. 1996, pp. 212–219.

[3] C. H. Bennett and G. Brassard, “Quantum cryptography: Public key distribution and coin tossing,” in Proc. IEEE Int. Conf. Comput., Syst. Signal, Bangalore, India, Dec. 1984, pp. 175–179.
[4] N.Gisin,G. Ribordy,W. Tittel, andH. Zbinden, “Quantum cryptography,” Rev. Mod. Phys., vol. 74, pp. 145–195, 2002.
[5] H.-K. Lo, “Insecurity of quantum secure computations,” Phys. Rev. A, vol. 56, pp. 1154–1162, 1997.
[6] V. Giovannetti, S. Lloyd, and L. Maccone, “Quantum private queries,” Phys. Rev. Lett., vol. 100, p. 230502, 2008.
[7] V. Giovannetti, S. Lloyd, and L. Maccone, “Quantum private queries: Security analysis,” IEEE T. Inform. Theory, vol. 56, no. 7, pp. 3465–3477, Jul. 2010.
[8] F. D. Martini, V. Giovannetti, S. Lloyd, L. Maccone, E. Nagali, L. Sansoni, and F. Sciarrino, “Experimental quantum private queries with linear optics,” Phys. Rev. A, vol. 80, p. 010302, 2009.
[9] L. Olejnik, “Secure quantum private information retrieval using phase encoded queries,” Phys. Rev. A, vol. 84, p. 022313, 2011.