INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 3 ISSUE 3 (2017) PAGES 2432 - 2439
RECEIVED : 25.04.2017 PUBLISHED : 29.05.2017
May 29, 2017

An Efficient Privacy-Preserving Outsourced Calculation Toolkits with Multiple Keys

P.Vinoth kumar [1], G.Kannabiran M.Tech [2]

Global institute of Engineering and Technology, Department of CSE Melvisharam, India.

ABSTRACT - This paper proposes a toolkit for efficient and privacy-preserving outsourced calculation under multiple encrypted keys, which is refer as EPOM. Using EPOM, a large scale of users can securely outsource their data to a cloud server for storage. Moreover, encrypted data belonging to multiple users can be processed without compromising on the security of the individual user's (original) data and the final computed results. To reduce the associated key management cost and private key exposure risk in EPOM, This paper presents a Distributed Two Trapdoor Public-Key Cryptosystem (DT-PKC), the core cryptographic primitive. This paper also presents the toolkits to ensure that the commonly used integer operations can be securely handled across different encrypted domains. This method also prove that the proposed EPOM achieves the goal of secure integer number processing without resulting in privacy leakage of data to unauthorized parties. Lastly, this demonstrates the utility and the efficiency of EPOM using simulations.

Index Terms - Privacy-Preserving, Homomorphic Encryption, Outsourced Computation, Antonym keys; multiple keys.

1. INTRODUCTION

Cloud computing due to its capability to support real time and massive storing and processing of data, is increasingly used in domains such as Internet of Things .A toolkit for efficient and privacy-preserving outsourced calculation under multiple encrypted keys (EPOM). Using EPOM, a large scale of users can securely outsource their data to a cloud server for storage. Despite the benefits afforded by the use of cloud computing, data security and privacy remain areas of ongoing focus. In attempts to conserve resources, reduce operational costs, and maintain efficiency, cloud service providers often store data belonging to multiple users on the same server (i.e. multi tenancy). Therefore, different users should be distributed with an individual key (i.e. multiple keys), to avoid multi-tenancy related attacks (e.g. a user's private data viewed by other unauthorized users). One application of the multi-key setting is e-healthcare cloud , here patients can transmit and store their health related information (e.g. patient's heart rate, blood pressure and glucose levels) on the hospital's cloud servers. This will facilitate diagnosis of the patients' physical condition based on the information. It is, however, important to ensure the security and privacy of patient's health and other personally identifiable information, such as health status. The privacy of decision making model used is also considered by the e-health service provider as a trade secret. One

way to achieve the security and privacy of the data is to issue all users (e.g. patients and service provider) different (unique) keys. In addition, an e-health service provider uses patients' health and PII (encrypted under different keys) in their training decision model.

2. SCOPE

The Scope is to achieve the privacy of divided private key which is guaranteed by Shamir secret sharing scheme which is information-theoretic secure. The strong private key SK is randomly split into two shares in a way that any less than two shares cannot recover the original SK Shamir secret sharing technique is used). It further implies that the adversary cannot cover the original plaintext with less than two shares of partial decrypted cipher texts.

3. RELATED WORK

With the constant evolution of cloud and related technologies, more users choose to encrypt before outsource their own data to cloud servers for storage. However, it is important to ensure the security and privacy of outsourced data. While Homomorphic encryption technique allows searching of encrypted data, it is not yet practical to do so. More specifically, Gentry [1] constructed the first fully Homomorphic encryption scheme based on lattice-based cryptography to support an arbitrary number of addition and multiplication operations. Since the seminal work of Gentry in 2009, a number of single-key fully Homomorphic encryption schemes (see [2], [3]) and multi-key fully Homomorphic encryption schemes (see [4], [5]) had been proposed. However, one of the biggest drawbacks of fully Homomorphic cryptosystems is complexity in both computation (including encryption and decryption) and storage (including both public/private key size and cipher text size). It is not yet practical to implement fully Homomorphic cryptosystem in the real-world . Partial Homomorphic encryptions (including additive and multiplicative Homomorphic encryption) are often considered the next best solution. However, partial Homomorphic encryptions can only handle one kind of Homomorphic operation with arbitrary times. Additive Homomorphic encryption scheme, such as Paillier cryptosystem and Benaloh cryptosystem, allows other parties to securely perform some additive Homomorphic calculations over the cipher text. Multiplicative Homomorphic encryption scheme, such as unpadded RSA cryptosystem and El-Gamal cryptosystem [7], allows some multiplication over the plaintext. In recent years, some cryptosystems attempt to provide for both additive and multiplicative operations. However, these systems generally achieve only limited numbers of Homomorphic operations. For example, the BGN cryptosystem can only support limited numbers of additive Homomorphic operations and only one multiplicative Homomorphic operation.

3. BASIC CRYPTO–DISTRIBUTED TWO TRAPDOORS PUBLIC-KEY CRYPTOSYSTEM (DT-PKC)

In order to realize EPOM, the public-key cryptosystem with a double trapdoor decryption cryptosystem introduced by Bresson et al. [8] could be a suitable solution for key management in the multi-key setting at first glance. However, the strong trapdoor leakage is a risk to the system, since encrypted data in Bresson et al.'s cryptosystem can be decrypted by the strong trapdoor. Therefore, we design a new cryptosystem – Distributed Two Trapdoors Public-Key Cryptosystem (DTPKC) – to split a strong private key into different shares. In addition, the weak decryption algorithm should support distributed decryption to solve the authorization problem in the multi-key environment (see Section V-I). Our DT-PKC is based on Bresson et al.'s cryptosystem [8], follows the idea in [9], and works as follows:

Step 1: Key generation

Step 2: Encryption

Step 3: Decryption with weak private key

Step 4: Decryption with strong private key

Step 5: Strong private key splitting

Step 6: Partial decryption with partial strong private key Step One

Step 7: Partial decryption with partial strong private key step two

Step 8: Partial decryption with partial weak private key step one

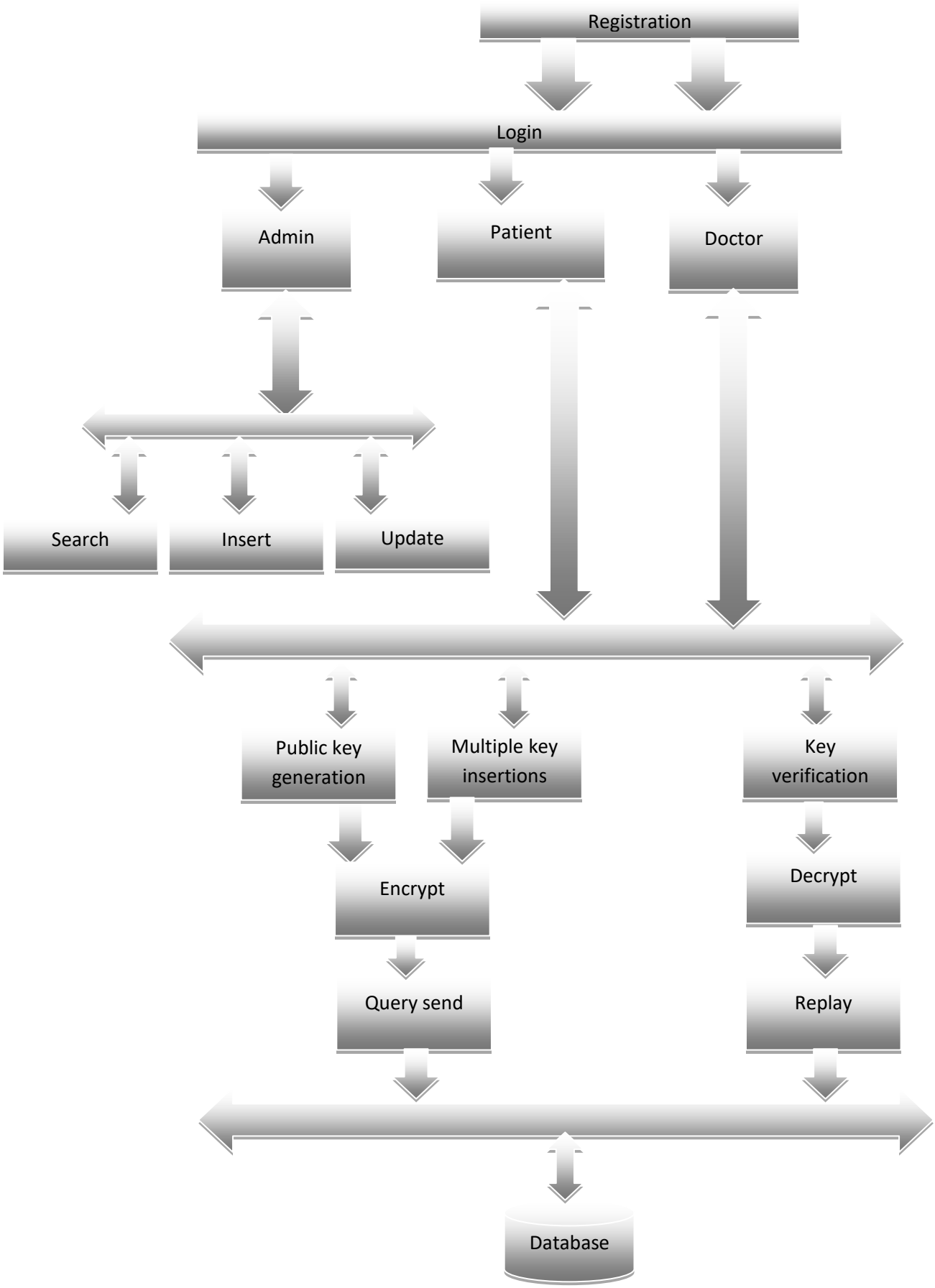
Step 9: Partial decryption with partial weak private key step one

Step 10: Cipher text Refresh

4. PRIVACY PRESERVING INTEGER CALCULATION TOOLKITS FOR MULTIPLE KEYS

After introducing the underlying algorithms in DT-PKC, This paper now present the secure sub-protocols as the toolkits for processing integers, namely: Secure Addition Protocol across Domains (SAD), Secure Multiplication Protocol across Domains (SMD), Secure Sign Bit Acquisition Protocol (SSBA) Secure Less Than Protocol (SLT), Secure Maximum and Minimum Sorting Protocol (SMMS), Secure Equivalent Testing Protocol (SEQ), Secure Division Protocol (SDIV) and Secure Greatest Common Divisor Protocol (SGCD). This paper assumes that both (Cloud Provider) CP and (Service Cloud Provider)CSP will be involved in the sub-protocol, as the CP holds a partial strong private key SK(1), and the CSP has the remaining partial strong private key SK(2) and public key.

INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 3 ISSUE 3 (2017) PAGES 2432 - 2439
RECEIVED : 25.04.2017 PUBLISHED : 29.05.2017
May 29, 2017



5. DESIGN MODEL

The users or nodes involved in our projects are Sender, Intermediate and Receiver. In order to send file, the sender has to find out the list of nodes which are connected with the sender. From that available list he can choose receiver. Then the sender has to analyze the performance of each and every node which is connected with the sender. The performance analysis list will return the priority based result so that sender can choose the intermediate to send the file. The Intermediate will receive the file from sender then it will analyze the performance so that it can send data to another intermediate or receiver. In the receiver side, the receiver has to select the file path to receive the file from sender or intermediate. Then the receiver can view the file received file.

6. IMPLEMENTATION

ADMIN

Authentication

The user has to give exact username and password which was provided at the time of registration, if login success means it will take up to main page else it will remain in the login page itself.

Patient Search

In this schema admin search the patient details using the patient names. And find out the patient by using patient ID.

Patient details

In this schema admin select the patient ID and insert the medical report or update the medical report

PATIENT

Authentication:

Login

The user has to give exact username and password which was provided at the time of registration, if login success means it will take up to main page else it will remain in the login page itself.

Registration

If you are the new user going to login into the application then you have to register first by providing necessary details. After successful completion of sign up process, the user has to login into the application by providing username and exact password.

View Medical Report

In this schema Patient can view their medical report if the report given by admin. Hence patient can know the status of their health if any doubt they can send query to their doctors along with their medical report

Public key generation

In this schema patient insert the message and generate the public key to encrypt the message

Multiple key Insertions

In this schema patient enter the message and Generate the public key to encrypt message. And insert multiple keys manually to decrypt the message

Message Sending

In this schema patient can send the message to doctor. That message should be in encrypted format by using multiple keys**.**

Verify private key

In this schema patient will get the replay from doctors but they have to verify the private key that assigned by doctor.

Decrypt replayed message

In this schema view the encrypted replay message from doctor and verify the private key assigned by doctor. Each time it will change. And decrypt the message using private key

DOCTOR

Authentication:

Login

The user has to give exact username and password which was provided at the time of registration, if login success means it will take up to main page else it will remain in the login page itself.

Registration

If you are the new user going to login into the application then you have to register first by providing necessary details. After successful completion of sign up process, the user has to login into the application by providing username and exact password.

View encrypted patient queries

In this schema view the patient queries in the encrypted format. Then select one of the Patients to view their queries.

Public key verification

In this schema to read the message from patients the doctor has to verify the public key. Public key is generated by the machine from the patient side. Hence no unauthorized person can access this message

Access private key

In this schema doctor should have to verify the public key. Then he only he can verify the private key that assigned to him. Private key assigned by the machine while the verification time of public key

Decrypt message

In this schema the doctor can decrypt the message using the private key. Then only he can read the message as well as view the medical report of the patient

7. CONCLUSION AND FUTURE WORK

In this paper, a new efficient and privacy preserving outsourced calculation framework with multiple keys is proposed. Here we are confirmed that the outsource data encrypted with multiple keys. Individually one key assigned to receiver to access that they need to verify the public key. If they are not authorized they cannot access the private key. To achieve secure data in cloud a distributed Two Trapdoors Public- Key Cryptosystem (DT-PKC) is used. This demonstrates that frameworks (and the underlying building blocks) are sufficiently efficient for a real-world deployment.

This can be further enhanced in terms of computation and storage In the near future, if an efficient multi-key fully holomorphic cryptosystem exists, we can remove the CSP from the system which will also result in a more elegant system both single key and multiple keys fully holomorphic cryptosystem in the existing scheme are rather inefficient.

7. REFERENCES

[1] C. Gentry, "Fully homomorphic encryption using ideal lattices," in Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009, 2009, pp. 169–178

[2] C. Gentry and S. Halevi, "Fully homomorphic encryption without squashing using depth-3 arithmetic circuits," in IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011, 2011, pp. 107–109.

[3] Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," SIAM J. Comput., vol. 43, no. 2, pp. 831–871, 2014.

[4] M. Clear and C. McGoldrick, "Multi-identity and multi-key leveled FHE from learning with errors," in Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part II, 2015, pp. 630–656.

[5] P. Mukherjee and D. Wichs, "Two round MPC from LWE via multi-key FHE," IACR Cryptology ePrint Archive, vol. 2015, p. 345, 2015. [Online]

[6] L. Morris, "Analysis of partially and fully homomorphic encryption," http://www.liammorris.com/crypto2 Homomorphic%20Encryption % 20\Paper.pdf, 2013.

[7] T. E. Gamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," vol. 31, no. 4, 1985, pp. 469

[8] E. Bresson, D. Catalano, and D. Pointcheval, "A simple public-key cryptosystem with a double trapdoor decryption mechanism and its applications," in Advances in Cryptology - ASIACRYPT 2003, 9th International Conference on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, November 30 - December 4, 2003, Proceedings, 2003, pp. 37–54.

[9] L. J. Hoffman, K. Lawson-Jenkins, and J. J. Blum, "Trust beyond security: an expanded trust model," Commun. ACM, vol. 49, no. 7, pp. 94–101, 2006.