IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

CLIENT CENTRIC PROXY RE-ENCRYPTION FOR OUTSOURCED DATA IN THE CLOUD

R.Sathya[1],S.Dhana Priyanka[2],S.Nithiya Shree[3],N.Madhumidha[4],S.Sowmiya[5]
[1]Assistant professor,Department of computer science and engineering
[2,3,4,5] Final year students
SRM TRP Engineering college
Tiruchirapalli,Tamil Nadu,India.

ABSTRACT

Data sharing in cloud computing is depriving user's direct control over the outsourced data, which inevitably raises security concerns and challenges. Extra computation cost and communication overhead have been introduced to the data owner. Normally in Public Key Encryption the data owner downloads and decrypt the requested data, and further re-encrypts it under the target user's public key which introduces extra computation cost and communication overhead which is a contradiction of the concept cloud computing. Another way is to allow data owners to define access policies so that the sharing data can be encrypted with the attribute-based encryption using the access policies which allows only authenticated users whose attributes matching their policies to decrypt the cipher text. However, here also data owner needs to download, decrypt and re-encrypt the requested data in case data access policies change dynamically and frequently. To overcome these drawbacks Client Centric Proxy Re-Encryption scheme gives a concrete solution for secure data sharing in cloud computing which deprives user's direct control over the outsourced data. By making cloud server responsible for re-encryption this application reduces communication overhead, extra computational cost which have been introduced to data owners.

Keywords--client Centric Proxy Re-Encryption, Computation Cost, Communication Overhead.

A.INTRODUCTION

When one is too busy to deal with all his encrypted files, he may wish to delegate his decryption rights to someone he trusts. This delegation of the power to decrypt the cipher text can be easily done if the delegator is online – simply decrypts the cipher text and re-encrypts the plaintext with the public key of whom he trusts. However, this is not always practical, for the delegator may not be online all the time and, it is undesirable to just disclose the secret key to some un-trusted server to do the transformation of the cipher text. To solve the above mentioned problems, at Eurocrypt'98, Blaze, Bleumer and Strauss firstly proposed the concept of proxy re-encryption (PRE). In a PRE scheme, a semi-trusted proxy with some additional information (re-encryption key, which is computed by the delegator in advance) can convert a cipher text computed under Alice's (delegator's) public-key into one intended to Bob (delegate) with the same plaintext. The fundamental property of proxy re-encryption schemes is that the proxy is not fully trusted, i.e., the proxy should not know the secret keys of Alice or Bob, and should not learn the plaintext during the conversion. Blaze et al. gave two methods to classify PRE schemes. One is according to the allowed times of transformation. If the cipher text can be transformed more than one time, i.e., from Alice to Bob, then from Bob to Carol, and so on, we call the PRE scheme multi hop; otherwise, it is single-hop. The other classification is according to the allowed direction of the transformation. If the re-encryption key can be used to transform the cipher text from Alice to Bob, and vice versa, we call the PRE scheme bidirectional; otherwise, it is unidirection.

B. LITERATURE SURVEY

1. Type and Identity Based Proxy Re-encryption Scheme

The type-and-identity-based proxy re-encryption scheme is based on the Boneh Franklin Identity Based Encryption scheme enabling implementation of different access control policies for cipher-texts against multiple receivers. The messages are categorized into different types according to the decryption rights of the intended receivers.

Demerits

This proposed scheme works only for the cipher-texts generated by the sender.

2. Conditional Proxy Re-encryption Scheme

In situation where fine-grained delegation is required requiring fulfilment of a predetermined condition, the notion of conditional proxy re-encryption (or CPRE) was introduced, whereby only cipher-text satisfying one condition set by Sender is allowed to be transformed and then decrypted by receiver. The scheme is Chosen Cipher text Attack-secure.

Demerits

They can be a set of pre-defined integers, the sending or receiving conditions of the parties, the physical location of the sender or the receiver.

3. Attribute Based Proxy Encryption Scheme

The Attribute based proxy re-encryption schemes solves the issues that occur while impersonating a user. In this scheme, various user attributes like city, country, street number are predefined while encryption. The decryption of a message is possible only if the user possesses the aforementioned attributes.

Demerits

The whole decryption fails since a single attribute cannot meet the threshold.

4. Key Private Proxy Re-encryption Scheme

Key Private Proxy Re-Encryption is also called as Anonymous Proxy Re-Encryption. The keys are kept private in such a way that even the proxy that performs the transformation of message cannot identify the users who are involved. No other PRE schemes introduced earlier provides key security.

Demerits

Though it is CPA-secure it is still working on providing security to CCA.

5. Cipher text-Policy Attribute based Proxy Re-encryption:

Cipher text-Policy ABPRE is a joint construction of attribute-based encryption and traditional proxy re-encryption scheme. This provides security against Chosen Plaintext Attacks. It is a type of ABE where the key is associated with an access structure namely a group of attributes defining the type of user that should be given access and decryption rights. This solves the issue of multiple users and key distribution over a large audience.

Demerits

Key management creates an overhead in such situations

6. Time/Clock Based Proxy Re-encryption Scheme

In a time based re-encryption scheme, each cloud server is allowed to independently re-encrypt data automatically in contrast to the previous methods where the data was encrypted only after receiving a command from the sender. Instead of using manual commands, an automatic re-encryption based on the internal time of cloud servers happens.

Demerits

Therefore a user satisfying the access structure i.e. the attribute set can decrypt the data if the time hasn't expired yet.

7. Threshold Proxy Re-encryption Scheme

The proposed system is constructed around the proposed scheme named Threshold Proxy Re-encryption. The cloud storage system stores the details of the users in the database. The user needs to get registered in the database, by entering his data like user_name, user_gender, user_location, user_password, user_birthdate, and user_e-mail address. The user then logs into the system using his credentials that were initially registered. The file is forwarded contained in a folder along with the user and recipients name, a security question for decryption access, the file containing the key for decryption and the status of the message. The file is transferred using the receiver's email and public key. After the file is received by the receiver, the selected file is downloaded. But before downloading the file, he has to download the key file that was sent in the same folder.

Demerits

Processed based decryption

Slows the process

Waiting for decryption

C. EXISTING SYSTEM

Type Based PRE provides cipher-text privacy control but it cannot do encoding operations over encrypted messages which **T** limits its widespread use.

Key-Private PRE provides security against Chosen cipher-text Attack but has privacy proof more difficult than Chosen plaintext attack. Identity-based PRE provides security against an adaptive CCA but it is difficult to find an algorithm that is multi-use, efficient and CCA secured.

Cipher text Policy Attribute-Based PRE Though it provides access control over data by limiting the decryption rights it has average efficiency and flexibility compared to the other schemes.

Conditional PRE schemes provide a very efficient mechanism against CCA.

Time based PRE provides a scalable user revocation and reduces the data owners' workload. The disadvantage is that it requires the effective time period to be same for all attributes associated with the user.

Threshold PRE Though it enables data forwarding efficiently it requires very high access control which is difficult to provide.

D. PROPOSED SYSTEM

Client Centric Proxy Re – Encryption

Client centric or User Centric proxy re-encryption can step in to offer an edge or user. The platform creates a re-encryption token off of the public key of the entity with whom its customers want to share data. That token can then be uploaded to the cloud where the third party can access it — in turn enabling them to decrypt and access the data. Ensuring compliance with regulations around the processing of sensitive data — data such as a bank or healthcare company might hold.

Merits

Cloud enablement

Secure sharing of sensitive encrypted data" — with multiple third parties, be it a customer, partner, supplier or even a regulator.

No Computational Overhead

CC-proxy re-encryption technology enables it to give customers the ability to manage access controls without needing to provide full access to the data — which means it can remove any single point of failure (i.e. via an admin who has to have full access control to all of the data).

No internet needed for encrypt or decrypt

E. METHODOLOGY

Proxy re-encryption is a set of algorithms which allows an un trusted proxy to transform cipher text from being encrypted under one key to another, without learning anything about the underlying plaintext. Proxy re-encryption algorithms usually work as public-key encryption, in which a public-private key-pair is used to encrypt and decrypt the data, respectively. As a class, proxy re-encryption is well-suited for use cases in which you want to share encrypted data with multiple parties. Rather than naively sharing your private key with recipients (insecure) or encrypting the entire message N times for each recipient, proxy re-encryption allows you to encrypt the data once and then delegate access to it based on the recipients' public keys. The requirement for the data owner to be online is removed which facilitates revocation of access.

The basic usage of Proxy Re-Encryption is shown. A more general model for PRE operations is assumed where a Policy Authority operates as a proxy for Alice. It generates Alice's public key and generates re-encryption keys to control who can decrypt information that are encrypted by Alice. The high-level operational flow of this key management infrastructure is as follows:

1. The public and secret key pairs are generated by policy authority. These keys are designated as pkA and skA, respectively. This key generation occurs before deployment, or when publishers need to send information to a PRE server.

2. Prior to deployment, the public key pkA is sent by policy authority to the publisher Alice. The publisher (and possibly multiple publishers) uses this public key to encrypt cipher texts cA = Enc(m, pkA) they send to the PRE server. The policy authority holds the secret key skA so that it can access information encrypted by the publisher.

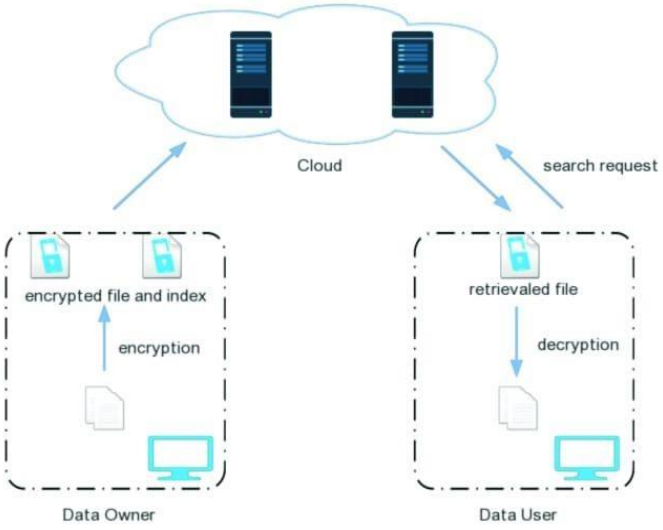


Figure: Proxy Re-Encryption Functional Key Management and Interaction Workflow

3. The subscriber Bob sends his public key (pkB) to the policy authority whenever a subscriber needs to receive information from the PRE server.

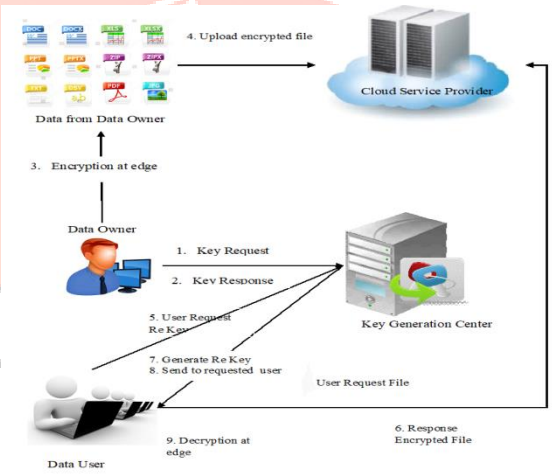
4. The publisher secret key (skA) and the subscriber public key (pkB) are used to generate a re-encryption key (rkAB) by the policy authority. This generation of re-key could occur prior to deployment or when a subscriber needs to receive information.

5. The re-encryption key is sent by the policy authority to the PRE server.

6. The cipher text is re-encrypted by PRE server so Bob can decrypt it.

7. Bob decrypts the cipher text using its secret key skB received by him. An important aspect of this key management infrastructure is that PRE pushes trust from the publisher to the policy authority and computational effort and bandwidth requirements to the PRE server. The policy authority determines who can share information and the PRE server uses information access policies to determine what subset of information from the publisher should be sent to the subscriber. The publisher and subscriber, who generally have the lowest computational capability in mobile applications, require the lowest computational effort and only need to maintain single keys, thus simplifying mobile deployments

F. IMPLEMENTATION



Proxy cryptography is a useful concept that can be applied in many contexts. An example is when a data owner store his/her data in a server in the Internet so that the data can be accessed from anywhere at any time. Although this model has many advantages, the data owner should consider the security of his/her confidential data, because without additional security mechanisms an attacker (who breaks the security of the server) and also the database administrator can do anything to the data: accessing, updating or removing the data. The data owner can implement an access control mechanism to secure their data. The access control mechanism is normally implemented by a trusted reference monitor in the server that intercepts access to the data. In an un-trusted server, this solution is not convenient and cost-heavy because the data owner needs to install a trusted hardware in the server securely. The access control can also be implemented by using cryptography (encryption mechanism). This method is convenient because the data owner does not need to interact with the data service provider (to set up a trusted hardware). The data

owner encrypts the sensitive data and stores the encrypted data in the server. The data owner can share the data (granting access) to other users accessing the database by providing the encryption keys. However, there are some inflexibilities with the encryption solution and access granting by providing the encryption keys. A user who has the encryption key can always decrypt the data– although the user's access right has been revoked. To completely revoke the access of the user, the data owner needs to re-encrypt the data with a new key. Using a naive solution, re-encryption is a costly computation, because basically the data owner needs to download the data, decrypt the data and encrypt with the new key and then upload the data to the database. A promising solution is by securely delegating re-encryption mechanism to a proxy in the database server. Before re-encryption, the owner only needs to send a re-encryption key to the proxy. The proxy re-encrypts the data without the need to decrypt any parts of the data. The proxy should be a semi-trusted party: it should execute the protocol correctly even though we do not trust the proxy accessing unencrypted data. To implement the proxy re- encryption, we need to find a function that translates the cipher text from one key to another key without having to access the plaintext. In a symmetric encryption setting, a solution is by encrypting the data using a two layers symmetric encryption. Re-encryption is performed only in the second layer encryption, however, to re-encrypt the data, the proxy needs to execute two costly computations: decrypt and then encrypt. There are many solutions for proxy re-encryption in the public-key world. The problem with these schemes is the schemes cannot be applied directly to the database because, for performance reason, the database is normally encrypted using symmetric key encryption. The public-key proxy re-encryption can be efficiently used to re-encrypt the private key that is used to encrypt the data with the symmetric ciphers.

G. CONCLUSION

In this paper, we introduce a new cryptographic primitive, called autonomous path proxy re-encryption, which is motivated by the demands in several potential applications. Not only do we first put forward the concept of delegator autonomous path proxy re-encryption, but also we give a concrete construction of an IND-CPA secure scheme under this concept. We note that such scheme combines the advantage of a single-hop PRE and a multi-hop PRE, in other words, CCPRE provides much better fine-grained access control to the delegation path than the traditional multi-hop PRE. In the new proposed AP-PRE scheme, the delegator has the ability to fully control the selection of the delegates as in a single-hop PRE, as well as the convenience and flexibility of a multi-hop PRE. In fact, an AP-PRE must be a multi-hop proxy re-encryption. Moreover, in our scheme, any delegate can designate a new delegation path, while any re encrypted cipher text from other path cannot be branched into his new path. As discussed, CCPRE schemes are desirable in many interesting applications

H. REFERENCES

[1] Giuseppe Ateniese, Karyn Benson, and Susan Hohenberger. Key-private proxy re-encryption. In Marc Fischlin, editor, CT-RSA, volume 5473 of Lecture Notes in Computer Science, pages 279–294. Springer, 2009.

[2] Giuseppe Ateniese, Kevin Fu, Matthew Green, and Susan Hohenberger. Improved proxy re-encryption schemes with applications to secure distributed storage. In NDSS. The Internet Society, 2005.

[3] Giuseppe Ateniese, Kevin Fu, Matthew Green, and Susan Hohenberger. Improved proxy re-encryption schemes with applications to secure distributed storage. ACM Trans. Inf. Syst. Secur., 9(1):1–30, 2006.

[4] Mihir Bellare and Amit Sahai. Non-malleable encryption: Equivalence between two notions, and an indistinguishability-based characterization. In Michael J. Wiener, editor, CRYPTO, volume 1666 of Lecture Notes in Computer Science, pages 519–536. Springer, 1999.

[5] Matt Blaze, Gerrit Bleumer, and Martin Strauss. Divertible protocols and atomic proxy cryptography. In EUROCRYPT, pages 127–144, 1998.

[6] Ran Canetti and Susan Hohenberger. Chosen-ciphertext secure proxy reencryption. In Peng Ning, Sabrina De Capitani di Vimercati, and Paul F. Syverson, editors, ACM Conference on Computer and Communications Security, pages 185–194. ACM, 2007.

[7] Cheng Kang Chu, Sherman S. M. Chow, Wen Guey Tzeng, Jianying Zhou, and Robert H. Deng. Key-aggregate cryptosystem for scalable data sharing in cloud storage. IEEE Transactions on Parallel & Distributed Systems, 25(2):468–477, 2014.

[8] Cheng Kang Chu, Jian Weng, Sherman S. M. Chow, Jianying Zhou, and Robert H. Deng. Conditional proxy broadcast re-encryption. In Australasian Conference on Information Security and Privacy, pages 327–342, 2009.

[9] Robert H. Deng, Jian Weng, Shengli Liu, and Kefei Chen. Chosenciphertext secure proxy re-encryption without pairings. In Matthew K. Franklin, Lucas Chi Kwong Hui, and Duncan S. Wong, editors, CANS, volume 5339 of Lecture Notes in Computer Science, pages 1–17.Springer, 2008.

[10] Li Ming Fang, Jian Dong Wang, Chun Peng Ge, and Yong Jun Ren. Fuzzy conditional proxy re-encryption. Science China Information Sciences, 56(5):1–13, 2013.

[11] Matthew Green and Giuseppe Ateniese. Identity-based proxy reencryption. In Jonathan Katz and Moti Yung, editors, ACNS, volume 4521 of Lecture Notes in Computer Science, pages 288–306. Springer, 2007.

[13] Susan Hohenberger, Guy N. Rothblum, Abhi Shelat, and Vinod Vaikuntanathan. Securely obfuscating re-encryption. In Salil P. Vadhan, editor, TCC, volume 4392 of Lecture Notes in Computer Science, pages 233– 252. Springer, 2007.

[12] Toshiyuki Isshiki, Manh Ha Nguyen, and Keisuke Tanaka. Proxy reencryption in a stronger security model extended from ct-rsa2012. In Ed Dawson, editor, CT-RSA, volume 7779 of Lecture Notes in Computer Science, pages 277–292. Springer, 2013.

[14] Anca-Andreea Ivan and Yevgeniy Dodis. Proxy cryptography revisited. In NDSS. The Internet Society, 2003.

[15] Aggelos Kiayias, Murat Osmanoglu, and Qiang Tang. Graded Encryption, or How to Play Who Wants To Be.